รายงานสหกิจศึกษาฉบับสมบูรณ์

ระบบบริหารจัดการอุบัติเหตุในโรงพยาบาล
Occurrence Management System in Hospital

นางสาววยุรี ทองเจิม

ภาควิชาวิศวกรรมคอมพิวเตอร์ สาขาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

รายงานสหกิจศึกษาฉบับสมบูรณ์

ระบบบริหารจัดการอุบัติเหตุในโรงพยาบาล

Occurrence Management System in Hospital

นางสาววยุรี ทองเจิม

ภาควิชาวิศวกรรมคอมพิวเตอร์ สาขาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

ชื่อโครงงานสหกิจศึกษา ระบบบริหารจัดการอุบัติเหตุในโรงพยาบาล

ชื่อ-สกุล นักศึกษา นางสาววยุรี ทองเจิม

คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมคอมพิวเตอร์ **สาขา** วิศวกรรมสารสนเทศ

ชื่อ-สกุล อาจารย์นิเทศ รศ.ดร.อรรถสิทธิ์ หล่าสกุล

ชื่อ-สกุล ผู้นิเทศงาน นายเอกชัย รัมสิน

ชื่อสถานประกอบการ บริษัท อัฟวาแลนท์ จำกัด

บทคัดย่อ

ระบบบริหารจัดการอุบัติเหตุในโรงพยาบาลนี้ จัดทำขึ้นเพื่อรับการแจ้งเหตุการณ์ต่าง ๆ ที่เกิดขึ้นในโรงพยาบาล ไม่ว่าจะเป็นอุบัติเหตุเกี่ยวกับคนไข้หรือบุคลากรในโรงพยาบาล รวมไปถึงอุปกรณ์ทางการแพทย์ หรืออุปกรณ์อำนวยความสะดวกทั้งหลาย ฯลฯ และนำเหตุการณ์นั้น ๆ มาวิเคราะห์โดยผู้เชี่ยวชาญที่รับผิดชอบ เพื่อหาทางแก้ปัญหาและวิธีป้องกันในอนาคต และพัฒนางานบริการของโรงพยาบาลให้ดีขึ้น ด้วยการใช้ซอฟต์แวร์ ONEWEB 4.0 ในการพัฒนาหน้าจอของระบบในส่วน Front-end หรือส่วนติดต่อผู้ใช้ และการปรับแต่งเพิ่มเติมโดยใช้ภาษา HTML CSS JavaScript และ jQuery ในการปรับแต่งหน้าตา รูปแบบการแสดงผล และภาษา Java ในการปรับแต่งการเข้าถึงฐานข้อมูลที่ซับซ้อนขึ้น รวมถึงการใช้คำสั่ง SQL เพื่อเรียกข้อมูลจากฐานข้อมูลออกมาแสดงผล เพิ่มข้อมูล และแก้ไขฐานข้อมูลให้มีการทำงานถูกต้องตามที่ได้ออกแบบไว้

คำสำคัญ : การแจ้งเหตุการณ์ อุบัติเหตุ แก้ปัญหาและป้องกัน ONEWEB 4.0 Front-end

Co-operative Title: Occurrence Management System in Hospital
Student Intern Name: Miss Wayuree Thongjerm
Faculty: Engineering
Department: Computer Engineering **Program:** Information Engineering
Advisor Name: Assoc.Prof.Dr. Attasit Lasakul
Mentor Name: Mr. Ekachai Rumsin
Company: Avalant Co., Ltd

ABSTRACT

The Occurrence Management System in Hospital is developed to receive events occurring in the hospital. Such as accidents involving patients, staff, medical equipment or medical equipment etc. The event was analyzed by the experts responsible to find out how to solve the problem, how to prevent it in the future and improve hospital services. Using ONEWEB 4.0 software to develop system screens in front-end and more customization using HTML, CSS, JavaScript and jQuery languages to customize display format and using Java language to customize complex database access. Including using SQL statements to retrieve data from the database to display, add data and modify the database. To work properly as designed.

Keywords: Occurrence, Incident, Solve Problems & Protecting, ONEWEB 4.0 and Front-end

กิตติกรรมประกาศ

รายงานฉบับนี้สามารถสำเร็จลุล่วงไปได้ด้วยดีจากความช่วยเหลือและความร่วมมือสนับสนุนของหลายฝ่าย โดยได้รับโอกาสจากบริษัท อัฟวาแลนท์ จำกัด ที่เปิดโอกาสให้นักศึกษาเข้ามามีส่วนร่วมในการทำงานกับองค์กร รศ.ดร.อรรถสิทธิ์ หล่าสกุล อาจารย์นิเทศ ที่คอยให้คำปรึกษาการมานิเทศงานที่บริษัท และให้การช่วยเหลือต่าง ๆ ตั้งแต่ต้นจนกระทั่งจบโครงการสหกิจศึกษา คณะวิศวกรรมศาสตร์และสาขาวิชาวิศวกรรมสารสนเทศ ที่ให้ความรู้ ผลักดัน และสนับสนุนมาโดยตลอด คุณเอกชัย รัมสิน (Senior Software Engineer) และคุณจันทิมา ลิ่มอักษรกุล พนักงานที่ปรึกษา ที่คอยช่วยเหลือดูแล ให้ความรู้ และสนับสนุนการทำงานอย่างเต็มที่ระยะเวลาโครงการ

สุดท้ายนี้ ขอขอบพระคุณผู้มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูล เป็นที่ปรึกษาในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ ตลอดจนให้การดูแล และให้ความเข้าใจเกี่ยวกับชีวิตของการทำงานจริง ข้าพเจ้าขอขอบพระคุณไว้ ณ ที่นี้

นางสาววยุรี ทองเจิม

วิศวกรรมสารสนเทศ

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VII

สารบัญภาพ VIII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญ 1

1.2 วัตถุประสงค์ของโครงการ 2

1.3 ขอบเขตของโครงการ 2

1.4 วิธีดำเนินการโครงการ 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 3

บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 4

2.1 ทฤษฎีที่เกี่ยวข้องในด้านภาษาคอมพิวเตอร์ 4

2.1.1 ภาษาเอชทีเอ็มแอล (HTML) 4

2.1.2 ภาษาซีเอสเอส (CSS) 5

2.1.3 ภาษาจาวาสคริปต์ (JavaScript) 6

2.1.4 ภาษาจาวา (Java) 8

2.1.5 เจคิวรี่ (jQuery) 10

2.1.6 ภาษาเอสคิวแอล (SQL) 11

2.1.6.1 ภาษาสำหรับนิยามข้อมูล (Data Definition Language: DDL) 11

2.1.6.2 ภาษาสำหรับจัดการข้อมูล (Data Manipulation Language: DML) 13

2.1.6.3 ภาษาสำหรับควบคุมข้อมูล (Data Control Language: DCL) 15

2.2 ทฤษฎีที่เกี่ยวข้องในด้านซอฟต์แวร์คอมพิวเตอร์ 15

2.2.1 ONEWEB 4.0 15

2.2.2 ระบบจัดการฐานข้อมูล (Database Management System: DBMS) 21

2.2.3 โปรแกรมระบบจัดการฐานข้อมูล 22

สารบัญ (ต่อ)

หน้า

2.2.4 โปรแกรม Eclipse 23
2.2.5 GitLab 24
2.2.6 Chrome Developer Tools 24
2.2.7 โปรแกรม SourceTree 25
2.2.8 โปรแกรม Hoo WinTail 26
บทที่ 3 วิธีดำเนินงาน 27
3.1 ศึกษาและเรียนรู้ข้อมูลที่จำเป็นต่อการทำงาน 27
3.2 เตรียมเครื่องมือหรือโปรแกรมที่ใช้ในการทำงาน 27
3.3 พัฒนาและปรับปรุงระบบให้เป็นไปตามกำหนด 33
บทที่ 4 ผลการดำเนินงาน 35
4.1 หน้าจอสำหรับข้อมูลมาสเตอร์ ส่วนผู้ดูแลระบบ (Master Setup - Admin) 35
4.1.1 หน้าจอ Setup BU 35
4.1.2 หน้าจอ Mapping BU Level 36
4.1.3 หน้าจอ Setup Occurrence Status 37
4.1.4 หน้าจอ Setup Department 38
4.1.5 หน้าจอ Setup Email Template 39
4.1.6 หน้าจอ Setup Listbox 40
4.1.7 หน้าจอ Setup Committee 41
4.1.8 หน้าจอ Setup Role 42
4.1.9 หน้าจอ Setup User 43
4.1.10 หน้าจอ Setup Change Holder 44
4.1.11 หน้าจอ Setup BU Group 45
4.2 หน้าจอสำหรับใช้งานระบบ ส่วนผู้ใช้งาน (Occurrence - User) 46
4.2.1 หน้าจอ New Occurrence 46
4.2.2 หน้าจอ New Occurrence RM 48
4.2.3 หน้าจอ Department Inbox 50

สารบัญ (ต่อ)

หน้า

4.3 ส่วนขยายเพิ่มเติมในหน้าจอ 52
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ 54
5.1 บทสรุป 54
5.2 ปัญหาที่พบในระหว่างการดำเนินงาน 54
5.3 แนวทางแก้ปัญหา 54
5.4 แนวทางพัฒนาต่อ และการนำไปใช้ 55
เอกสารอ้างอิง 56

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงคำสั่งต่าง ๆ ของภาษา DDL 11
ตารางที่ 2.2 แสดงคำสั่งต่าง ๆ ของภาษา DML 13

สารบัญภาพ

หน้า

ภาพที่ 2.1 ตัวอย่างส่วนประกอบภาษา HTML 4
ภาพที่ 2.2 ตัวอย่างคำสั่ง CSS 6
ภาพที่ 2.3 ตัวอย่างการเขียนคำสั่ง JavaScript 7
ภาพที่ 2.4 ตัวอย่างการเขียนและเรียกใช้ Function 7
ภาพที่ 2.5 โลโก้ของภาษา Java 9
ภาพที่ 2.6 ตัวอย่างคำสั่ง Java 9
ภาพที่ 2.7 jQuery 10
ภาพที่ 2.8 AppDesigner 16
ภาพที่ 2.9 Search Entity 17
ภาพที่ 2.10 Entities Management 17
ภาพที่ 2.11 Tab 18
ภาพที่ 2.12 Module 18
ภาพที่ 2.13 Button 19
ภาพที่ 2.14 Action 19
ภาพที่ 2.15 Field 20
ภาพที่ 2.16 แสดงโครงสร้างของระบบจัดการฐานข้อมูลและระบบปฏิบัติการ 21
ภาพที่ 2.17 โปรแกรม DbVisualizer Pro 22
ภาพที่ 2.18 Eclipse neon.3 23
ภาพที่ 2.19 GitLab 24
ภาพที่ 2.20 Chrome Developer Tools 25
ภาพที่ 2.21 SourceTree 25
ภาพที่ 2.22 Hoo WinTail 26
ภาพที่ 3.1 หน้าโปรแกรม SourceTree สำหรับ Clone Workspace 28
ภาพที่ 3.2 หน้าต่าง Eclipse สำหรับเลือก Workspace 28
ภาพที่ 3.3 ขั้นตอนการ Setting Server ที่ 1 29
ภาพที่ 3.4 ขั้นตอนการ Setting Server ที่ 2 29

สารบัญภาพ (ต่อ)

หน้า

ภาพที่ 3.5 ขั้นตอนการ Setting Server ที่ 3 30
ภาพที่ 3.6 ขั้นตอนการ Setting Server ที่ 4 31
ภาพที่ 3.7 การ Start Server 32
ภาพที่ 3.8 หน้าจอล็อกอินเข้าสู่ระบบ ONEWEB 4.0 32
ภาพที่ 3.9 หน้าจอ Create Database Connection 33
ภาพที่ 3.10 ลำดับขั้นตอนการทำงาน 34
ภาพที่ 4.1 หน้า Search Setup BU 35
ภาพที่ 4.2 หน้าจอ Search Mapping BU Level 36
ภาพที่ 4.3 หน้าจอ Insert/Update Mapping BU Level 36
ภาพที่ 4.4 หน้าจอ Search Setup Occurrence Status 37
ภาพที่ 4.5 หน้าจอ Insert/Update Setup Occurrence Status 37
ภาพที่ 4.6 หน้าจอ Search Setup Department 38
ภาพที่ 4.7 หน้าจอ Insert/Update Setup Department 38
ภาพที่ 4.8 หน้าจอ Search Setup Email Template 39
ภาพที่ 4.9 หน้าจอ Insert/Update Setup Email Template 39
ภาพที่ 4.10 หน้าจอ Search Setup Listbox 40
ภาพที่ 4.11 หน้าจอ Insert/Update Setup Listbox 40
ภาพที่ 4.12 หน้าจอ Search Setup Committee 41
ภาพที่ 4.13 หน้าจอ Insert/Update Setup Committee 41
ภาพที่ 4.14 หน้าจอ Search Setup Role 42
ภาพที่ 4.15 หน้าจอ Insert/Update Setup Role 42
ภาพที่ 4.16 หน้าจอ Search Setup User 43
ภาพที่ 4.17 หน้าจอ Insert/Update Setup User 43
ภาพที่ 4.18 หน้าจอ Search Setup Change Holder 44
ภาพที่ 4.19 หน้าจอ Insert/Update Setup Change Holder 44
ภาพที่ 4.20 หน้าจอ Search Setup BU Group 45

สารบัญภาพ (ต่อ)

หน้า

ภาพที่ 4.21 หน้าจอ Insert/Update Setup BU Group 45
ภาพที่ 4.22 หน้าจอ Search Occurrence 46
ภาพที่ 4.23 หน้าจอ Insert/Update New Occurrence 47
ภาพที่ 4.24 หน้าจอ Search RM Occurrence 48
ภาพที่ 4.25 หน้าจอ Update RM Occurrence 49
ภาพที่ 2.26 หน้าจอ Search Department Inbox 50
ภาพที่ 4.27 หน้าจอ Update Department Inbox 51
ภาพที่ 4.28 Pop up แจ้งเตือนเมื่อกดปุ่ม Back 52
ภาพที่ 4.29 Pop up แจ้งเตือนให้ระบุข้อมูล HN 52
ภาพที่ 4.30 Pop up แจ้งเตือนให้ระบุข้อมูล Cancel Reason 53
ภาพที่ 4.31 ไอคอนสำหรับค้นหาข้อมูล Physician 53
ภาพที่ 4.32 หน้าจอล็อกอินเข้าใช้งานระบบ 53

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

เนื่องมาจากบริษัท อัฟวาแลนท์ จำกัด เป็นผู้ประกอบการด้านเทคโนโลยีสารสนเทศที่ให้คำปรึกษาและพัฒนาระบบซอฟต์แวร์ แอปพลิเคชันให้กับหน่วยงานราชการและบริษัทเอกชนต่าง ๆ และเป็นผู้นำในการพัฒนาซอฟต์แวร์ แพลตฟอร์ม (Software Platform) โดยบริษัทได้พัฒนาแพลตฟอร์ม ONEWEB ขึ้นมา เป็นหนึ่งในผลิตภัณฑ์ที่ทางบริษัทได้ทำการผลิตเพื่อเสนอขายให้กับองค์กรต่าง ๆ ที่เกิดจากการริเริ่มสร้างโปรแกรมสำหรับสร้างหน้าเว็บเพจด้วยวิธีการลากวาง (Drag & Drop) ทดแทนวิธีการโค้ดดิ้ง (Coding) เพื่อทำให้กระบวนการสร้างเว็บเพจมีความสะดวกมากขึ้นและลดเวลาในการสร้างเว็บเพจให้น้อยลง โดยในการสร้างหน้าเว็บเพจหน้าหนึ่งจากเดิมที่เคยใช้เวลา 4 - 5 วัน กลับใช้เวลาในการสร้างเพียง 1 - 2 วัน เท่านั้นซึ่งส่งผลให้ใช้จำนวนบุคลกรในการดำเนินงานน้อยลงไปด้วยเช่นกัน

นั่นจึงทำให้ ONEWEB เป็นแอปพลิเคชันที่ตอบโจทย์ผู้ใช้งานทุกระดับ และรองรับความต้องการของธุรกิจได้อย่างครอบคลุม มั่นคง รวดเร็วและยั่งยืน โดยเป็นนวัตกรรมเพื่อช่วยเหลือองค์กรขนาดกลางและขนาดใหญ่ให้มีการเติบโตทางผลผลิต ผลกำไร และความสามารถในการแข่งขัน เพื่อรองรับธุรกิจรูปแบบใหม่ที่จะเกิดขึ้น เพิ่มศักยภาพและขีดความสามารถขององค์กรต่าง ๆ ทั้งภาครัฐและเอกชน และการปรับตัวเข้าสู่ยุคดิจิทัลให้มีประสิทธิภาพและรองรับการแข่งขันในเวทีโลก

เนื่องจากมีโรงพยาบาลที่มีกลุ่มโรงพยาบาลอยู่ในเครือข่ายต้องการควบคุมมาตรฐานงานบริการของทุกโรงพยาบาล โดยจะมีกลุ่มบริหารความเสี่ยง (Risk Management) ที่จัดตั้งขึ้นมาเพื่อจัดการความเสี่ยงต่าง ๆ ที่เกิดขึ้น และนำซอฟต์แวร์ระบบบริหารจัดการอุบัติเหตุในโรงพยาบาลมาใช้รับการแจ้งเหตุต่าง ๆ เพื่อนำเหตุการณ์นั้น ๆ มาวิเคราะห์เพื่อหาทางแก้ปัญหาและวิธีป้องกันในอนาคตเพื่อพัฒนางานบริการให้ดีขึ้น

การเข้าร่วมโครงการสหกิจศึกษากับทางบริษัท อัฟวาแลนท์ จำกัด ในตำแหน่ง นักพัฒนาซอฟต์แวร์ (Developer) ภายในแผนกฝ่ายพัฒนาซอฟต์แวร์และโซลูชั่น (Professional Service) มีหน้าที่หลักในการพัฒนาซอฟต์แวร์ให้กับองค์กรภายในและภายนอก โดยถูกมอบหมายให้รับผิดชอบการพัฒนาระบบบริหารจัดการอุบัติเหตุในโรงพยาบาลให้สามารถใช้งานได้ตรงตามวัตถุประสงค์ที่ได้กำหนดไว้

1.2 วัตถุประสงค์ของโครงการ

1.2.1 พัฒนาให้ระบบมีคุณภาพดีขึ้น และใช้งานง่ายขึ้น

1.2.2 ลดระยะเวลาการทำงานให้สะดวก และรวดเร็วยิ่งขึ้น

1.2.3 ได้รับความรู้ใหม่และทักษะเฉพาะทางจากการได้ทำงานจริง

1.2.4 ฝึกความรับผิดชอบในหน้าที่ที่ได้รับมอบหมาย

1.3 ขอบเขตของโครงการ

ขอบเขตของงานที่ได้รับมอบหมายในช่วงโครงการสหกิจศึกษาที่ได้รับผิดชอบจากบริษัท

1.3.1 การกำหนดคุณสมบัติของโปรแกรม (Configuration) เพื่อให้โปรแกรมทำงานตามข้อกำหนดคุณลักษณะโปรแกรม (Program Specification) โดยใช้ซอฟต์แวร์ ONEWEB 4.0

1.3.2 การปรับแต่งโปรแกรมเพิ่มเติม (Customization) ในสิ่งที่การ Configuration พื้นฐานทำไม่ได้ โดยการเขียนภาษา Java, JavaScript, JSP ด้วยซอฟต์แวร์ Eclipse

1.3.3 การเขียน Query เพื่อให้โปรแกรมแสดงข้อมูลจากฐานข้อมูล DbVisualizer โดยใช้ภาษา SQL

1.3.4 การทดสอบการทำงานของโปรแกรมให้เป็นไปตามข้อกำหนดคุณลักษณะโปรแกรม (Program Specification)

1.4 วิธีดำเนินการโครงการ

การวางแผนการทำงานในขั้นตอนต่าง ๆ ในการดำเนินงานโครงการสามารถจำแนกออกได้ตามแต่ละส่วนของขอบเขตโครงการดังกล่าวไว้ข้างต้นได้ ดังนี้

1.4.1 การกำหนดคุณสมบัติของโปรแกรม (Configuration) โดยใช้ซอฟต์แวร์ ONEWEB 4.0 ในการสร้างหน้าจอสำหรับผู้ใช้งาน (User Interface) และการปรับแต่งโปรแกรมเพิ่มเติม (Customization) ด้วยซอฟต์แวร์ Eclipse ในการกำหนดเงื่อนไขต่าง ๆ ของระบบให้ได้ผลลัพธ์ตามที่มีการออกแบบไว้แล้ว

1.4.2 การทดสอบการทำงานของโปรแกรมเบื้องต้น (Unit Test) ด้วยตนเอง เพื่อหาจุดบกพร่องหรือข้อผิดพลาดของระบบและทำการแก้ไขให้ถูกต้อง ก่อนระบบจะถูกนำไปทดสอบโดยผู้ทดสอบคุณภาพซอฟต์แวร์ (Tester) ในขั้นตอนต่อไป

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่ได้รับในการพัฒนาซอฟต์แวร์ที่ได้รับมอบหมายในการเข้าร่วมโครงการสหกิจศึกษากับทางบริษัท อัฟวาแลนท์ จำกัด สามารถได้ออกเป็น 3 ส่วน ดังนี้

1.5.1 ประโยชน์ต่อบุคลากรที่เกี่ยวข้อง

1.5.1.1 บุคลากรสามารถใช้ซอฟต์แวร์นี้ในการทำงานได้สะดวก และรวดเร็วมากขึ้น

1.5.1.2 เพิ่มความแม่นยำ และทำให้ระบบมีประสิทธิภาพมากขึ้น

1.5.1.3 องค์กรสามารถนำวิธีการพัฒนาซอฟต์แวร์ไปต่อยอดให้กับผลิตภัณฑ์ซอฟต์แวร์ชิ้นอื่นต่อไปได้

1.5.2 ประโยชน์ต่อตนเอง

1.5.2.1 ได้รับประสบการณ์วิชาชีพตามสาขาวิชาที่เรียนเพิ่มเติมนอกเหนือไปจากการเรียนในห้องเรียน

1.5.2.2 ได้นำความรู้จากการเรียนมาประยุกต์ใช้ในการทำงานจริง

1.5.2.4 ได้เรียนรู้ พัฒนาตนเอง มีความรับผิดชอบ และมั่นใจในตนเองมากขึ้น

1.5.2.5 ได้ศึกษาขั้นตอนและกระบวนการต่าง ๆ ของการทำงาน และเรียนรู้ในการปรับตัวให้เข้ากับสภาพแวดล้อมในการทำงาน

1.5.2.6 ได้ฝึกระเบียบวินัย กระบวนการคิด และการทำงานร่วมกับผู้อื่น

บทที่ 2
แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

โครงการนี้เป็นการพัฒนาโปรแกรมสำหรับนำมาใช้ในการจัดการโครงการ (Project) ภายในองค์กร การศึกษาค้นคว้าซอฟต์แวร์ต่าง ๆ ที่เกี่ยวข้องนั้นเป็นสิ่งจำเป็น แล้วนำทฤษฎีเหล่านั้นมาประยุกต์ใช้ในการทำงานที่ได้รับมอบหมายให้สำเร็จสมบูรณ์ โดยทฤษฎีที่เกี่ยวข้องที่นำมาศึกษาในการทำโครงการมีรายละเอียด ดังต่อไปนี้

2.1 ทฤษฎีที่เกี่ยวข้องในด้านภาษาคอมพิวเตอร์

2.1.1 ภาษาเอชทีเอ็มแอล (HTML)

HTML ย่อมาจาก Hypertext Markup Language เป็นภาษาคอมพิวเตอร์รูปแบบหนึ่ง ที่มีโครงสร้างการเขียนโดยอาศัยตัวกำกับ (Tag) ควบคุมการแสดงผลข้อความ รูปภาพ หรือวัตถุอื่น ๆ ผ่านโปรแกรมเบราว์เซอร์ แต่ละ Tag อาจจะมีส่วนขยาย เรียกว่า Attribute สำหรับระบุ หรือควบคุมการแสดงผล ของเว็บได้ด้วย HTML เป็นภาษาที่ถูกพัฒนาโดย World Wide Web Consortium (W3C) จากแม่แบบของภาษา SGML (Standard Generalized Markup Language) โดยตัดความสามารถบางส่วนออกไป เพื่อให้สามารถทำความเข้าใจและเรียนรู้ได้ง่าย และด้วยประเด็นดังกล่าว ทำให้บริการ www เติบโตขยายตัวอย่างกว้างขวาง [6]

เอกสาร HTML จะประกอบด้วยส่วนประกอบ 2 ส่วน ดังนี้

1. ส่วน Head คือส่วนที่จะเป็นหัว (Header) ของหน้าเอกสารทั่วไป หรือส่วนชื่อเรื่อง (Title) ของหน้าต่างการทำงานในระบบ Windows

2. ส่วน Body จะเป็นส่วนเนื้อหาของเอกสารนั้น ๆ ซึ่งจะประกอบด้วย Tag คำสั่งในการจัดรูปแบบ หรือตกแต่งเอกสาร HTML

ในทั้งสองส่วนนี้จะอยู่ภายใน Tag <HTML>...</HTML> ดังนี้

```
<html>
<head> <title> ส่วนชื่อเอกสาร </title> </head>
<body> tag คำสั่ง </body>
</html>
```

ภาพที่ 2.1 ตัวอย่างส่วนประกอบภาษา HTML

ในการเขียนคำสั่งภาษา HTML สามารถเขียน ด้วยตัวอักษร เล็กหรือใหญ่ ทั้งหมดหรือเขียนคละกันได้ เช่น <HTML> หรือ <Html> หรือ <html> ซึ่งจะให้ผลเหมือนกัน

คำสั่งเริ่มต้นของเอกสาร HTML

คำสั่ง <HTML> เป็นคำสั่งเริ่มต้นในการเขียนโปรแกรมและคำสั่ง </HTML> เป็นการสิ้นสุดโปรแกรม HTML คำสั่งนี้จะไม่แสดงผลในโปรแกรมเว็บเบราว์เซอร์ แต่ต้องเขียนเพื่อให้เกิดความเป็นระบบของงาน และเพื่อจะให้รู้ว่าเอกสารนี้เป็นเอกสารของภาษา HTML ส่วนหัวเรื่องเอกสารเว็บ (Head Section)

คำสั่ง <HEAD> Head Section เป็นส่วนที่ใช้อธิบายเกี่ยวกับข้อมูลเฉพาะของหน้าเว็บนั้น ๆ เช่น ชื่อเรื่องของหน้าเว็บ (Title), ชื่อผู้จัดทำเว็บ(Author), คีย์เวิร์ดสำหรับการค้นหา (Keyword) โดยมี Tag สำคัญ คือ <TITLE>...</TITLE > โดยข้อความที่ใช้เป็น TITLE ไม่ควรพิมพ์เกิน 64 ตัวอักษร ต้องไม่ใส่ลักษณะพิเศษ เช่น ตัวหนา เอียง หรือสี และควรใช้ภาษาที่มีความหมายครอบคลุมถึงเนื้อหาของเว็บเพจนั้น หรือเป็นคำสำคัญในการค้นหา (Keyword)

คำสั่ง <BODY> Body Section เป็นส่วนเนื้อหาหลักของหน้าเว็บ ซึ่งการแสดงผลจะต้องใช้ Tag จำนวนมาก ขึ้นอยู่กับลักษณะของข้อมูล เช่น ข้อความ รูปภาพ เสียง วีดิโอ หรือไฟล์ต่าง ๆ ส่วนเนื้อหาเอกสารเว็บ เป็นส่วนการทำงานหลักของหน้าเว็บ ประกอบด้วย Tag มากมายตามลักษณะของข้อมูลที่ต้องการนำเสนอ การป้อนคำสั่งในส่วนนี้ ไม่มีข้อจำกัดสามารถป้อนติดกัน หรือ 1 บรรทัดต่อ 1 คำสั่งก็ได้ แต่ส่วนใหญ่จะยึดรูปแบบที่อ่านง่าย คือ การทำย่อหน้าในชุดคำสั่งที่เกี่ยวข้องกัน ทั้งนี้ให้ป้อนคำสั่งทั้งหมดภายใต้ Tag <BODY>...</BODY>

2.1.2 ภาษาซีเอสเอส (CSS)

CSS ย่อมาจาก Cascading Style Sheets เป็นภาษาที่มีรูปแบบการเขียน Syntax ที่เฉพาะและถูกกำหนดมาตรฐานโดย W3C (World Wide Web Consortium) เช่นเดียวกับ HTML และ XHTML ใช้สำหรับตกแต่งเอกสาร HTML/XHTML ให้มีหน้าตา สีสัน ตัวอักษร เส้นขอบ พื้นหลัง ระยะห่าง ฯลฯ อย่างที่เราต้องการ ด้วยการกำหนดคุณสมบัติให้กับ Element ต่างๆ ของ HTML เช่น <body>, <p>, <h1> เป็นต้น

โครงสร้างคำสั่งของ CSS ประกอบด้วย selector, property และ value โดย selector สามารถเป็น HTML Tag ต่างๆ เช่น <body>, <p> หรือเป็น Class Name หรือ ID ก็ได้ ส่วน property คือ คุณสมบัติในการจัดรูปแบบการแสดงผล เช่น color สำหรับกำหนดสี, font-size สำหรับกำหนดขนาดตัวอักษร และ value เป็นค่าที่เรากำหนดให้กับ property ต่าง ๆ เช่น font-size:14px เป็นต้น [12]

```
/* selector ที่เป็น HTML Tag */
p {
color: #000000;
text-align: center
}
```

ภาพที่ 2.2 ตัวอย่างคำสั่ง CSS

2.1.3 ภาษาจาวาสคริปต์ (JavaScript)

จาวาสคริปต์เป็นภาษาที่มีลักษณะการเขียนแบบโปรโตไทป์ (Prototyped-based Programming) โดยมีโครงสร้างของภาษาและไวยกรณ์อยู่บนพื้นฐานของภาษาซี ทำงานได้ข้ามระบบได้ มีขนาดเล็กและใช้ง่าย ถูกออกแบบมาเพื่อการทำงานในแอปพลิเคชันและผลิตภัณฑ์อื่น ๆ ส่วนมากใช้ในหน้าเว็บเพื่อประมวลผลข้อมูลที่ฝั่งของผู้ใช้งาน แต่ก็ยังมีใช้เพื่อเพิ่มเติมความสามารถในการเขียนสคริปต์โดยฝังอยู่ในโปรแกรมอื่น ๆ [5]

ถ้าพิจารณาการเขียนโปรแกรมบนเว็บจะเห็นว่ามีการใช้งานอยู่ 2 ลักษณะ ได้แก่

2.1.2.1 จาวาสคริปต์ฝั่งเซิร์ฟเวอร์ (Server)

ขยายความสามารถของภาษาหลักด้วยการเพิ่มอ็อบเจกต์ที่จำเป็นต่อการทำงานของจาวาสคริปต์บนเซิร์ฟเวอร์ ตัวอย่างเช่น ส่วนขยายฝั่งเซิร์ฟเวอร์ทำให้แอปพลิเคชันสามารถติดต่อกับระบบฐานข้อมูลได้ เรียกใช้ข้อมูลจากหลายส่วนของแอปพลิเคชันได้ หรือจัดการไฟล์บนเซิร์ฟเวอร์ได้ ได้แก่ ภาษา PHP ASP และ JSP

2.1.2.2 จาวาสคริปต์ฝั่งไคลเอนต์ (Client)

ขยายความสามารถของภาษาหลักด้วยการเพิ่มอ็อบเจกต์ที่ควบคุมเบราว์เซอร์ (เนวิเกเตอร์ หรือเว็บเบราว์เซอร์อื่น) และ Document Object Model (DOM) ตัวอย่างเช่น ส่วนขยายฝั่งไคลเอนต์ทำให้แอปพลิเคชันสามารถจัดวางส่วนประกอบต่าง ๆ ลงในหน้าเว็บฟอร์ม และตอบสนองต่อเหตุการณ์ที่ผู้ใช้กระทำ เช่น กดเมาส์ ป้อนข้อมูลลงฟอร์ม และเปลี่ยนหน้าข้อมูลได้ ได้แก่ ภาษา HTML และ JavaScript

การเขียนคำสั่งของ JavaScript ต้องเขียนร่วมกับภาษา HTML โดยสามารถเขียนแทรกอยู่ภายในระหว่างคำสั่ง <Head> กับ </Head> หรือหลังจาก <Body> และการเขียนคำสั่งตัวอักขระพิมพ์เล็กและตัวอักขระพิมพ์ใหญ่ถือว่ามีความแตกต่างกัน (Case sensitive) [15]

คำสั่งการเขียนสคริปต์ของ JavaScript เริ่มต้นด้วย <script language="JavaScript"> และลงท้ายด้วย </script> ตัวอย่างเช่น

```
<html>
<head>
<title>This is a JavaScript example</title>
        <script language="JavaScript">
                document.write("Hello World!");
        </script>
</head>
<body> สวัสดี ! </body>
</html>
```

ภาพที่ 2.3 ตัวอย่างการเขียนคำสั่ง JavaScript

ฟังก์ชั่นใน JavaScript มีหน้าที่เหมือนฟังก์ชั่นในภาษาต่าง ๆ เช่น ภาษาซี ซึ่งทำหน้าที่เป็นโปรแกรมย่อย ๆ ที่เราสามารถสร้างขึ้นมาเองได้ นิยมนำมาใช้กับคำสั่งที่เราต้องเขียนคำสั่งนั้นซ้ำ ๆ เป็นการประหยัดเนื้อที่และประหยัดเวลา เริ่มจากประกาศว่าเป็นฟังก์ชั่นและตามด้วยชื่อฟังก์ชั่นและมีวงเล็บเปิดวงเล็บปิด แล้วจึงเขียนคำสั่งภายในวงเล็บ วิธีการเรียกใช้คำสั่งทำโดยการนำชื่อฟังก์ชั่นมาใส่ ณ ตำแหน่งที่ต้องการให้ฟังก์ชั่นนั้นทำงาน ตัวอย่างเช่น

```
<script>
function test()
{
   document.write('Test function of JavaScript? ')
}
test()
</script>
```

ภาพที่ 2.4 ตัวอย่างการเขียนและเรียกใช้ Function

Event ใน Java Script คือ เหตุการณ์ที่เกิดขึ้นขณะผู้ใช้กระทำกับวัตถุหรือสิ่งต่าง ๆ ที่เกิดขึ้นบนจอภาพ เช่น การคลิกปุ่ม การวางเมาส์บนรูปภาพ การวางเมาส์เหนือข้อความ หรือการกดแป้นพิมพ์ เป็นต้น ให้พิจารณาตัวอย่าง Event ต่อไปนี้

- onClick เป็นเหตุการณ์หรือพฤติกรรมที่ผู้ใช้คลิกเมาส์ที่ปุ่มใด ๆ ที่เราเขียนคำสั่งไว้รองรับการกดปุ่มจากผู้ใช้
- onLoad จะใช้ในกรณีที่ผู้ใช้เรียกใช้เว็บหรือการ Load เว็บนั้นมาครั้งแรก การใช้ Event onLoad นั้นนิยมใช้กับ Tag ของ HTML
- onMouseover และ onMouseout เป็น Event ที่เกี่ยวกับการกระทำของเมาส์ กล่าวคือ onMouseover จะกระทำเมื่อเมาส์อยู่เหนือวัตถุหรือตัวอักษร ในทางกลับกัน Event onMouseout จะกระทำเมื่อเคลื่อนเมาส์ออกจากวัตถุหรือข้อความนั้น ๆ
- onUnload เป็น Event สำหรับเหตุการณ์ที่ผู้ใช้ปิดการใช้ Browser หลังจากที่เข้ามาเยี่ยมในเว็บไซต์ของเราหรือผู้ใช้เปลี่ยน URL ใหม่ก็ได้
- onBlur เป็น Event ที่ใช้ควบคุมการใช้ของผู้ใช้ในกรณีที่ผู้ใช้ป้อนข้อมูลในช่อง Text Field แล้วออกจาก Text Field นั้นเพื่อกดปุ่ม Submit หรือเพื่อป้อนข้อมูล Text Field อื่นก็ตาม เราอาจจะต้องตรวจสอบก่อนว่าข้อมูลที่ป้อนนั้นถูกต้องหรือไม่ เราต้องใช้ Event onBlur
- onSubmit ไม่เหมือนกับ onBlur onSubmit จะอยู่ใน Form และไม่อยู่ใน Element ใด ๆ

2.1.4 ภาษาจาวา (Java)

จาวาเป็นภาษาเขียนโปรแกรมเพื่อวัตถุประสงค์ทั่วไป โดยสามารถทำงานได้พร้อมกัน เป็นภาษาที่สร้างมาจากคลาส และสนับสนุนการเขียนโปรแกรมแบบออบเจ็คอย่างสมบูรณ์ และถูกออกแบบมาให้พร้อมสำหรับการใช้งานมากที่สุด ซึ่งมีเมธอดและคลาสต่างๆ อำนวยความสะดวกให้ใช้มากมาย โดยภาษา Java นั้นมีความตั้งใจว่าจะทำให้นักพัฒนาออกแบบและพัฒนาโปรแกรมน้อยลง นั่นคือการเขียนเพียงครั้งเดียว แต่นำไปใช้งานได้ทุกที่หรือทุกแพลตฟอร์ม [7]

ภาพที่ 2.5 โลโก้ของภาษา Java

แอปพลิเคชันของภาษา Java นั้นโดยปกติแล้วจะคอมไพล์เป็น bytecode ที่สามารถรันได้ใน Java virtual machine (JVM) ขึ้นกับสถาปัตยกรรมของคอมพิวเตอร์นั้น ๆ และใน ปี 2016 ภาษา Java เป็นภาษาที่ได้รับความนิยมและใช้มากที่สุดในโลก โดยเฉพาะการใช้พัฒนาเว็บแอปพลิเคชัน

```java
public class Demo {
    public static void main(String[] args) throws IOException {
        //declare new File and Scanner objects
        File file = new File("input.txt");
        Scanner inputFile = new Scanner(file);
        //loop through txt file
        while(inputFile.hasNext()){
            //read next line
            String line = inputFile.nextLine();
            System.out.print(line);
            //call check method to determine balance
            if(check(line))
                System.out.print("\t--> correct\n");
            else
                System.out.print("\t--> incorrect\n");
        }
        inputFile.close();
    }
```

ภาพที่ 2.6 ตัวอย่างคำสั่ง Java

โปรแกรมจาวามีส่วนประกอบที่สำคัญดังนี้

Package: เป็นกลุ่มของคลาสหรือไลบรารี่มาตรฐานของภาษา Java ที่มีฟังก์ชันต่างๆ ให้ใช้มากมาย ในตัวอย่างนั้นไม่มี เราจะพูดเกี่ยวกับเรื่องนี้ในภายหลัง

Class: ในส่วนของการประกาศคลาส จะต้องประกาศคลาสให้ชื่อตรงกับไฟล์เสมอ นอกจาก Inner คลาสที่อยู่ในคลาสเดียวกัน โดยชื่อคลาสนั้นควรจะขึ้นต้นด้วยตัวใหญ่ และถ้ามีหลายคำให้ใช้ตัวพิมพ์ใหญ่แบ่ง ดังตัวอย่างด้านล่าง

Method: หลังจากคลาสสร้างแล้ว จะเป็นประกาศเมธอดภายในคลาส โดยในการที่จะรันโปรแกรมได้จะต้องมีเมธอดที่ชื่อว่า Main ดังตัวอย่างในโปรแกรมด้านบน มันเป็นที่แรกที่โปรแกรมจะเริ่มทำงาน

Statements: เป็นคำสั่งของโปรแกรมเพื่อให้โปรแกรมทำงานตามต้องการ เช่น System.out.print("Hello World!"); เป็นการแสดงผลข้อความออกทางหน้าจอ โดยปกติโปรแกรมมักจะมีหลายคำสั่ง

2.1.5 เจคิวรี่ (jQuery)

jQuery เป็น JavaScript Library ที่มีการรวบรวมฟังก์ชันของ JavaScript ต่าง ๆ ให้อยู่ในรูปแบบ Patterns Framework ที่สะดวกและง่ายต่อการใช้งาน มีความยืดหยุ่นรองรับต่อการใช้งาน Cross Browser คือไม่ว่าจะใช้งานบนเว็บเบราว์เซอร์ใด ใน Library ของ jQuery จะมีการเลือกใช้ฟังก์ชันที่ เหมาะสมต่อการทำงานและแสดงผลในเว็บเบราว์เซอร์ที่กำลังทำงานอยู่ ซึ่งช่วยลดปัญหาการทำงานที่ผิดพลาดในฝั่งของ Client ได้ จากปัญหาก่อนนี้ นักโปรแกรมเมอร์ทั้งหลายในสมัยก่อนมักจะทดสอบโปรแกรมและพัฒนาบน IE แต่ตอนนี้ได้มีหลายเว็บเบราว์เซอร์ได้เกิดขึ้นมากมาย เช่น Chrome Firefox หรือ Safari และบางคำสั่งของ JavaScript จะไม่ทำงานหรือไม่ support ในเว็บเบราว์เซอร์บางตัว ด้วยเหตุผลนี่เองการใช้ jQuery มาเป็นทางเลือกก็สามารถช่วยแก้ปัญหานี้เป็นได้อย่างดี ทั้งยังสะดวกต่อการใช้งาน เพราะเป็น Syntax ที่เข้าใจง่าย และเขียนได้ในรูปแบบที่สั้น ๆ รองรับการทำงานทั้งใน HTML รูปแบบเดิม หรือ CSS Element DOM element Effect การจัดการ Event ต่าง ๆ หรือแม้กระทั่งการพัฒนา Ajax ด้วย jQuery ก็สามารถ ทำได้อย่างง่ายดาย โดย Syntax เหล่านี้ยังคงทำงานอยู่ภายใต้คำสั่งของภาษา JavaScript แต่การเรียกใช้งาน Framework หรือฟังก์ชันต่าง ๆ จะถูกกำหนดรูปแบบโดย Patterns ที่ได้ถูกออกแบบไว้ใน Library ของ jQuery

ภาพที่ 2.7 jQuery

2.1.6 ภาษาเอสคิวแอล (SQL)

ภาษาเอสคิวแอล หรือ Standard relational database Query Language (SQL) หรือ ซีควอล (SE-QUEL) เปนภาษามาตรฐานสำหรับระบบฐานขอมูล ซึ่งเปนภาษาที่พัฒนาขึ้นมาโดยบริษัท ไอบีเอ็ม ภาษา SQL (Standard Query Language) เปนสวนหนึ่งของระบบฐานขอมูลเชิงสัมพันธ์ (Relational Database) ที่ได้รับความนิยมมากเพราะงายตอความเขาใจ และอยูในรูปภาษาอังกฤษ [1] [2] [3] [4] ภาษา SQL แบงออกเปน 3 สวน คือ

2.1.6.1 ภาษาสำหรับนิยามข้อมูล (Data Definition Language: DDL)

Data Definition Language (DDL) เป็นภาษาที่ใชนิยามโครงสรางขอมูล เพื่อเปลี่ยนแปลง หรือยกเลิกโครงสรางฐานขอมูลตามที่ออกแบบไว้ โครงสรางดังกลาวคือ สคีมา (Schema) นั้นเอง ตัวอยางเชน การกำหนดใหฐานขอมูลประกอบดวยตารางอะไรบาง ชื่ออะไร ประเภทใด มีอินเด็กซ (Index) ประกอบดวย 3 คำสั่งคือ คำสั่งการสราง (Create) คำสั่งเปลี่ยนแปลงโครงสราง (Alter) และคำสั่งยกเลิก (Drop) กล่าวคือ เป็นภาษาที่ใชสรางฐานขอมูลลงในคอมพิวเตอรหลังจากที่เราไดออกแบบแลวว่าฐานข้อมูลมีกี่รีเลชั่น แตละรีเลชั่นมีความสัมพันธอยางไร จากนั้นการใชภาษา DDL นี้แปลงรีเลชั่นตาง ๆ ใหอยูในรูปภาษาสำหรับนิยามขอมูล เพื่อปอนเขาสูระบบฐานขอมูล เพื่อสรางฐานขอมูลที่แทจริงใหเกิดขึ้น ภาษา DDL สามารถสรุปคำสั่งตาง ๆ ไดดังต่อไปนี้

ตารางที่ 2.1 แสดงคำสั่งต่าง ๆ ของภาษา DDL

คำสั่ง	ความหมาย
CREATE TABLE	นิยามโครงสรางขอมูลในรูปตารางบนฐานขอมูล
DROP TABLE	ลบโครงสรางตารางขอมูลออกจากระบบ
ALTER TABLE	แกไขปรับปรุงโครงสรางตาราง
CREATE INDEX	สรางดัชนีของตาราง
DROP INDEX	ลบดัชนีของตารางออกจากระบบ
CREATE VIEW	กำหนดโครงสรางวิวของผูใช
DROP VIEW	ลบโครงสรางวิวออกจากระบบ

นอกจากนั้นยังมีตัวแปรที่ใช้กำหนดโครงสร้างข้อมูล ดังต่อไปนี้

ตารางที่ 2.2 แสดงตัวแปรที่ใช้กำหนดโครงสร้างข้อมูล

ประเภทข้อมูล	ความหมาย
SMALLINT	เลขจำนวนเต็ม ค่าระหว่าง -99999 ถึง 99999
INTEGER	เลขจำนวนเต็ม ความกว้าง ประมาณ ลบ 10 หลัก ถึงบวก 11 หลัก
NUMBER (X,Y)	เก็บจำนวนตัวเลขที่มีความกว้างรวม X หลัก และมีทศนิยม Y ตำแหน่ง
FLOAT (X,Y)	เก็บจำนวนตัวเลขที่เขียนในรูปของ 10 ยกกำลัง
CHAR (N)	เก็บสายอักขระที่มีขนาดคงที่ตามที่ระบุ
VARCHER (M)	เก็บสายอักขระที่มีขนาดไม่คงที่
DATE	เก็บ เดือน/วัน/ปี
LOGICAL	เก็บค่าตรรกะ คือ จริง (T หรือ True) หรือเท็จ (F หรือ False)

(1) คำนิยามโครงสร้างตาราง

การสร้างตารางในฐานข้อมูลเชิงสัมพันธ์ โดยเฉพาะฐานข้อมูลขนาดใหญ่บนระบบ UNIX จะทำด้วยการป้อนคำสั่งในลักษณะเท็กซ์โหมด (Text Mode) เข้าไปในระบบฐานข้อมูลดังรูปแบบต่อไปนี้

CREATE TABLE <ชื่อตาราง>
(<ชื่อคอลัมน์ ประเภทของข้อมูล>
[,<ชื่อคอลัมน์ ประเภทของข้อมูล>]...);

(2) คำสั่งการลบโครงสร้างตาราง

DROP TABLE <ชื่อตารางที่ต้องการลบ>

(3) คำสั่งการเปลี่ยนแปลงโครงสร้างตาราง

ALTER TABLE <ชื่อตารางที่ต้องการเปลี่ยนแปลง>
<คำสั่งการเปลี่ยนแปลง><[,<ชื่อคอลัมน์ ประเภทของข้อมูล>]>

(4) คำสั่งดัชนี

ดัชนี (INDEX) มีความสำคัญมากต่อฐานข้อมูลเชิงสัมพันธ์ เนื่องจากฐานข้อมูลเชิงสัมพันธ์ (RBDMS) จะใช้ดัชนีในการค้นหาระเบียนที่ต้องการได้อย่างรวดเร็ว โดยดัชนีที่ถูกสร้างขึ้นจะเก็บไว้แยกจากตารางในอีกพื้นที่ของคอมพิวเตอร์ โดยปกติ ถ้ามีการประกาศดัชนีไว้ การค้นหาข้อมูลในตารางนั้นจะต้องทำแบบเรียงลำดับจากแถวที่หนึ่งจนถึงแถวสุดท้าย การสร้างดัชนีสำหรับตารางใด ๆ จะทำ

ได้โดยการเลือกคอลัมน์ใดคอลัมน์หนึ่งในตารางมาเป็นดัชนี และตารางหนึ่ง ๆ สามารถมีได้หลายดัชนี นอกจากเพิ่มความรวดเร็วในการดึงข้อมูลแล้ว ยังสามารถนำไปใช้ในการควบคุมคอลัมน์ที่นำมาสร้างเป็นดัชนีให้ให้มีการเก็บข้อมูลที่ไม่ซ้ำกัน (Unique) อีกด้วย

การสรางดัชนีจะใช้รูปแบบตอไปนี้

CREATE [UNIQUE] INDEX <ชื่อตาราง>

ON (<ชื่อตารางที่สรางดัชนี> (< ชื่อคอลัมน_1>

[,< ชื่อคอลัมน_2>]...);

(5) การลบดัชนี

เมื่อตองการลบดัชนีที่สรางขึ้นก็สามารถทำไดดวยรูปแบบต่อไปนี้

DROP INDEX <ชื่อดัชนี>

(6) ประเภทของข้อมูล

ประเภทของข้อมูลแบ่งเป็น 5 ประเภทใหญ่ ๆ ขึ้นอยู่กับซอฟต์แวร์ระบบฐานข้อมูลที่ใช้ ตัวอย่างเช่น CHAR, INTEGER, DATE ฯลฯ

2.1.6.2 ภาษาสำหรับจัดการข้อมูล (Data Manipulation Language: DML)

หลังจากที่เราสร้างโครงสร้างฐานข้อมูลขึ้นแล้ว คำสั่งต่อไปในการป้อนข้อมูลลงในฐานข้อมูลและเปลี่ยนแปลงข้อมูลในฐานข้อมูล โดยการใช้ภาษาสำหรับการการจัดการข้อมูล (Data Manipulation Language: DML) ใช้จัดการข้อมูลภายในตารางในฐานข้อมูล และภาษาสำหรับแก้ไขและเปลี่ยนแปลงตาราง แบ่งออกเป็น 4 Statement คือ Select Statement สำหรับการเรียกหา (Retrieve) ข้อมูลจากฐานข้อมูล Insert Statement สำหรับการเพิ่มข้อมูลลงในตาราง (Table) Delete Statement สำหรับการลบข้อมูลออกจากตาราง (Table) และ Update Statement การเปลี่ยนแปลงข้อมูลลงในตาราง (Table)

ตารางที่ 2.3 แสดงคำสั่งต่าง ๆ ของภาษา DML

คำสั่ง	ความหมาย
SELECT	เรียกค้นข้อมูลในตาราง
INSERT	เพิ่มแถวข้อมูลลงในตาราง
DELETE	ลบแถวข้อมูลออกจากตาราง
UPDATE	ปรับปรุงแถวข้อมูลในตาราง

(1) คำสั่งค้นหาข้อมูล (Query Statement)

คำสั่ง SELECT เป็นคำสั่งในการเรียกดูข้อมูล หรือค้นหาข้อมูลตามเงื่อนไขที่ระบุ เนื่องจากคำสั่ง SELECT เป็นคำสั่งที่มีรูปแบบการใช้งานง่ายเพื่อช่วยในการค้นหาข้อมูลที่ซับซ้อนดวย รูปแบบต่อไปนี้

SELECT <ชื่อคอลัมน์ที่ต้องการดูข้อมูล>

FROM <ชื่อตาราง>

WHERE <เงื่อนไขตามที่ระบุ>

(1.1) SELECT เป็นคำสั่งให้ทำการเรียกดูข้อมูลในคอลัมน์ที่ระบุ ซึ่งอาจจะมากกว่า หนึ่งก็ได้ และถ้ามีมากกว่าหนึ่งคอลัมน์ต้องคั่นด้วยเครื่องหมายคอมม่า (,) และนอกจากนี้ยังสามารถใช้เครื่องหมายดอกจัน (*) เพื่อแสดงถึงการขอดูข้อมูลทั้งหมดได้อีกด้วย

(1.2) FROM เป็นส่วนประกอบของคำสั่งที่บอกถึงตารางที่ต้องการดูซึ่งอาจจะมีมากกว่าหนึ่งตารางก็ได้ที่จะถูกเรียกใช้จากคำสั่ง SELECT

(1.3) WHERE เป็นส่วนประกอบของคำสั่ง ที่ใช้บ่งบอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล ขึ้นมาจากตารางใด ๆ ที่อยู่หลัง FROM

การเรียกดูแบบซ้อนกัน (Nested SELECT Statement)

SELECT <ชื่อคอลัมน์>

FROM <ชื่อตาราง>

WHERE <ชื่อคอลัมน์> IN

(SELECT <ชื่อคอลัมน์>

FROM <ชื่อตาราง>

WHERE <ชื่อคอลัมน์>)

(2) คำสั่งเติมข้อมูล (Insert Statement)

INSERT INTO < ชื่อตาราง >

VALUES (< ชื่อคอลัมน์_1> [, < ชื่อคอลัมน์_2>] ...);

(3) คำสั่งแก้ไขและลบแถว (Update Statement)

UPDATE < ชื่อตาราง >

SET <ค่าที่ต้องการ>

WHERE <เงื่อนไข>

2.1.6.3 ภาษาสำหรับควบคุมข้อมูล (Data Control Language: DCL)

ใช้เป็นภาษาที่ใช้ควบคุมระบบรักษาความปลอดภัยของฐานข้อมูล ประกอบด้วยคำสั่ง 2 คำสั่ง คือ คำสั่ง GRANT เป็นคำสั่งที่ใช้กำหนดสิทธิให้กับผู้ใช้แต่ละคนให้มีสิทธิกระทำการใดกับข้อมูล เช่นการเพิ่มข้อมูล การแก้ไข หรือการลบข้อมูลในตาราง และคำสั่ง REVOKE เป็นคำสั่งให้มีการยกเลิกสิทธินั้นหลังจากที่ได้ GRANT แล้ว

2.2 ทฤษฎีที่เกี่ยวข้องในด้านซอฟต์แวร์คอมพิวเตอร์

2.2.1 ONEWEB 4.0

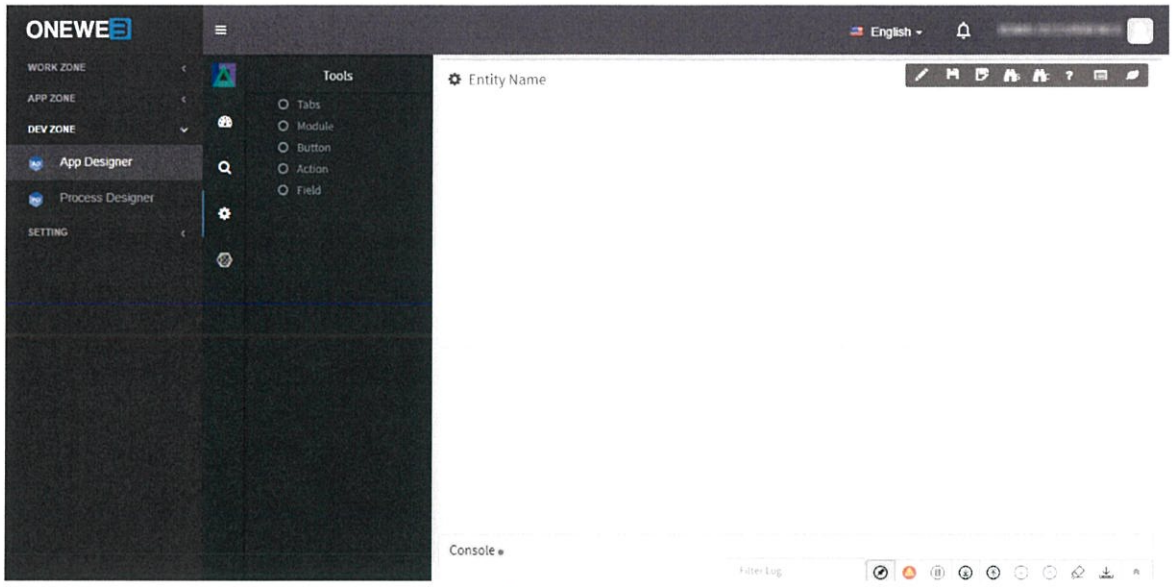
เป็นแพลตฟอร์มสำหรับการพัฒนาซอฟต์แวร์แบบ Configuration-based ซึ่งองค์กรไม่จำเป็นต้องจ้างนักพัฒนา ผู้ใช้งานหรือที่ปรึกษาทางธุรกิจหรือนักวิเคราะห์ระบบนั้นสามารถเข้ามาพัฒนาซอฟต์แวร์ได้โดยไม่ต้องมีความรู้เรื่องการเขียนโปรแกรม ทำให้การพัฒนาซอฟต์แวร์ให้ตรงต่อความต้องการในการใช้งานนั้นสามารถทำได้อย่างรวดเร็วในระยะเวลาอันสั้น โดยใช้นักพัฒนาซอฟต์แวร์จำนวนน้อยลง และเป็นทางเลือกใหม่สำหรับองค์กรที่ต้องการซอฟต์แวร์สำหรับมาใช้ตอบโจทย์การทำงานในแต่ละแผนกขององค์กร

นอกจากช่วยให้การพัฒนาซอฟต์แวร์ตรงตามความต้องการมากขึ้นแล้ว OneWeb ยังช่วยลดค่าใช้จ่ายในการลงทุนด้านการพัฒนาซอฟต์แวร์ขององค์กรในระยะยาวลง ในขณะที่เพิ่มความยืดหยุ่นในการปรับซอฟต์แวร์ให้สอดคล้องไปกับการเปลี่ยนแปลงในการดำเนินธุรกิจ และช่วยให้พนักงานทุกคนสามารถทำงานได้อย่างมีประสิทธิภาพสูงสุดจากการใช้งานซอฟต์แวร์ที่ตอบโจทย์ในการทำงานอย่างแท้จริง

จุดเด่นหลักของ OneWeb ที่เปิดให้ผู้ใช้งานสามารถสร้างแอปพลิเคชันต่าง ๆ ของตนเองขึ้นมาได้เลยนั้น ทางบริษัทได้จัดการแบ่งหมวดหมู่ความสามารถในการพัฒนาแอปพลิเคชันใหม่ ๆ ให้ง่ายต่อการนำมาใช้งานมากขึ้น โดยแบ่งเป็นระบบ AppDesigner ProcessDesigner และ AppSpace โดยในโครงการนี้ได้ใช้งานส่วน AppDesigner เป็นหลัก

2.2.1.1 AppDesigner

สำหรับใช้ในการสร้างหน้า Web Application แบบ Responsive ดูแลเรื่องฐานข้อมูล และการปรับแต่ง User Interface (UI) ได้ด้วยการลากวางปุ่มและฟอร์มต่าง ๆ ได้ตามต้องการ โดยมีส่วนประกอบดังนี้

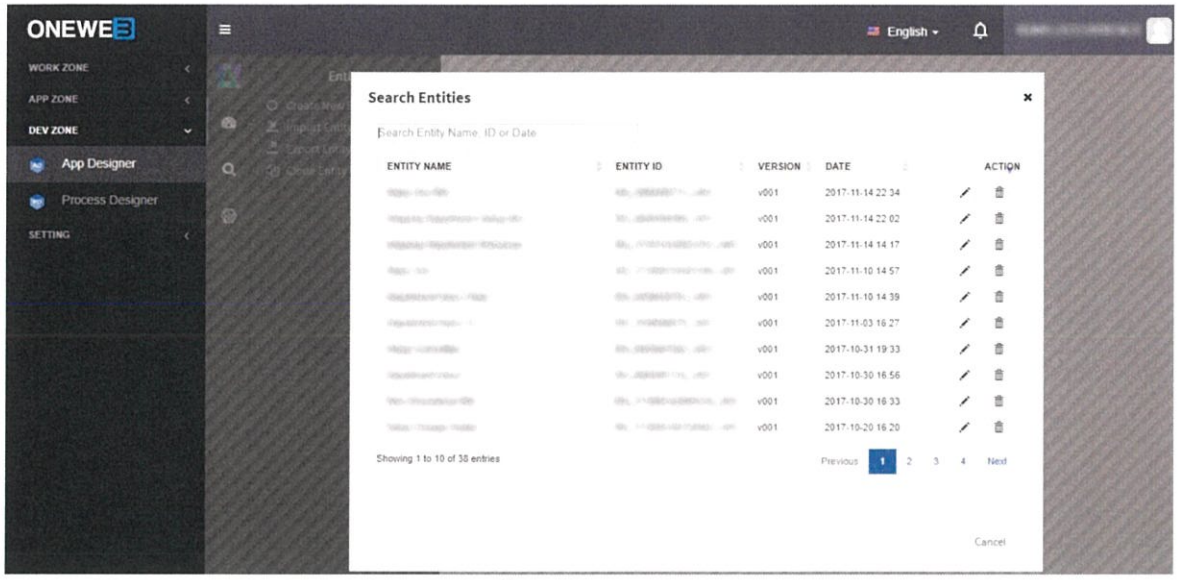


ภาพที่ 2.8 AppDesigner

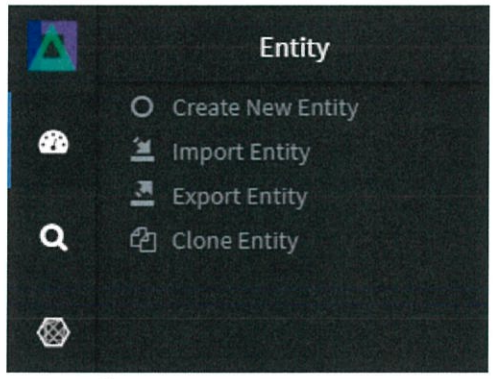
(1) Entity

Entity หรือ Business Entity คือ กลุ่มของข้อมูลทางธุรกิจและการทำธุรกรรมที่เกี่ยวข้องกับกระบวนการทางธุรกิจ เช่น ข้อมูลลูกค้า ใบสั่งซื้อ ผลิตภัณฑ์ ใบกำกับภาษี เป็นต้น โดยต้องกำหนดขอบเขตทางธุรกิจที่เหมาะสมอย่างรอบคอบก่อนที่จะเริ่มการกำหนดค่าใน AppDesigner ซึ่งมาการทำงานดังนี้

- Search Entity สำหรับค้นหา Entity ที่ได้สร้างไว้แล้ว
- Create New Entity สำหรับสร้าง Entity ใหม่
- Import Entity สำหรับนำเข้า Entity โดยอัพโหลดเข้าไป
- Export Entity สำหรับส่งออก Entity โดยดาวน์โหลดออกมา
- Clone Entity สำหรับคัดลอก Entity นั้นไปสร้างเป็นอีก Entity



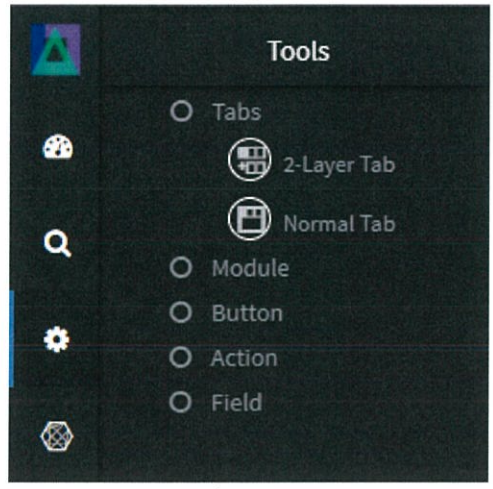
ภาพที่ 2.9 Search Entity



ภาพที่ 2.10 Entities Management

(2) Tab

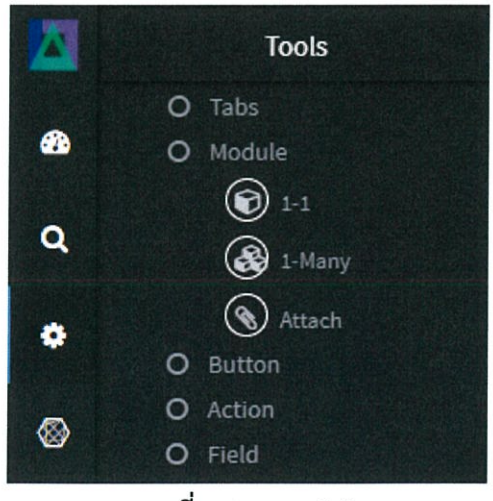
Tab คือ การแสดงหน้าจอแบบแท็บมีคุณสมบัติการเปิดหน้าโปรแกรมผ่านหลายแท็บในหน้าต่างเดียว แบ่งเป็น 2 แบบ คือ **Normal Tab** ใช้เพื่อสร้างเอนทิตีเพียงหนึ่งเลเยอร์เท่านั้น นอกจากนี้ยังใช้เพื่อสร้างแท็บใหม่สำหรับเอนทิตี 2 ชั้นด้วยเช่นกัน และ **2-Layer Tab** ใช้สำหรับออกแบบอินเตอร์เฟสที่ซับซ้อนมากขึ้น มีแท็บบนและแท็บล่าง



ภาพที่ 2.11 Tab

(3) Module

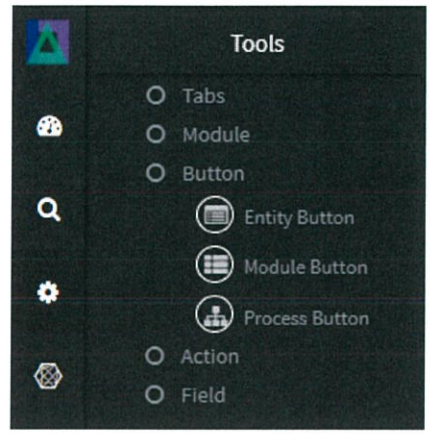
Module คือ ส่วนที่อยู่ภายใต้ Tab ใช้สำหรับแบ่งการส่วนรับหรือแสดงข้อมูลไปตามตารางต่าง ๆ แบ่งเป็น 3 แบบ คือ **1-1** (One to One) เป็น Module สำหรับข้อมูลแบบ 1-1 คือข้อมูล 1 ชุดใน 1 Table **1-Many** (One to Many) เป็น Module สำหรับข้อมูลแบบ 1-Many คือข้อมูลหลายชุดใน 1 Table และ **Attach** เป็น Module ที่มีการแนบไฟล์จากที่อื่น



ภาพที่ 2.12 Module

(4) Button

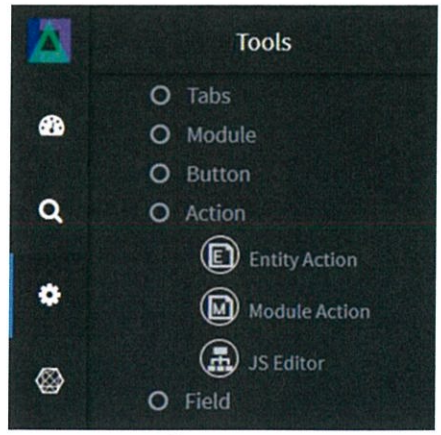
Button คือ ปุ่มที่จะแสดงและควบคุมหน้าจอ แบ่งเป็น 3 ชนิด คือ **Entity Button** เป็นปุ่มสำหรับ Entity จะปรากฏเมื่ออยู่ตามหน้าจอที่ตั้งค่าไว้ โดย Action ปุ่มจะส่งผลทั้งหน้าจอ Entity **Module Button** เป็นปุ่มสำหรับ Module จะรองลงมาจาก Entity Button โดย Action ปุ่มจะส่งผลแค่เฉพาะ Module นั้น ๆ และ **Process Button** เป็นปุ่มที่เมื่อกดจะทำตาม Process ที่ออกแบบใน Process Designer



ภาพที่ 2.13 Button

(5) Action

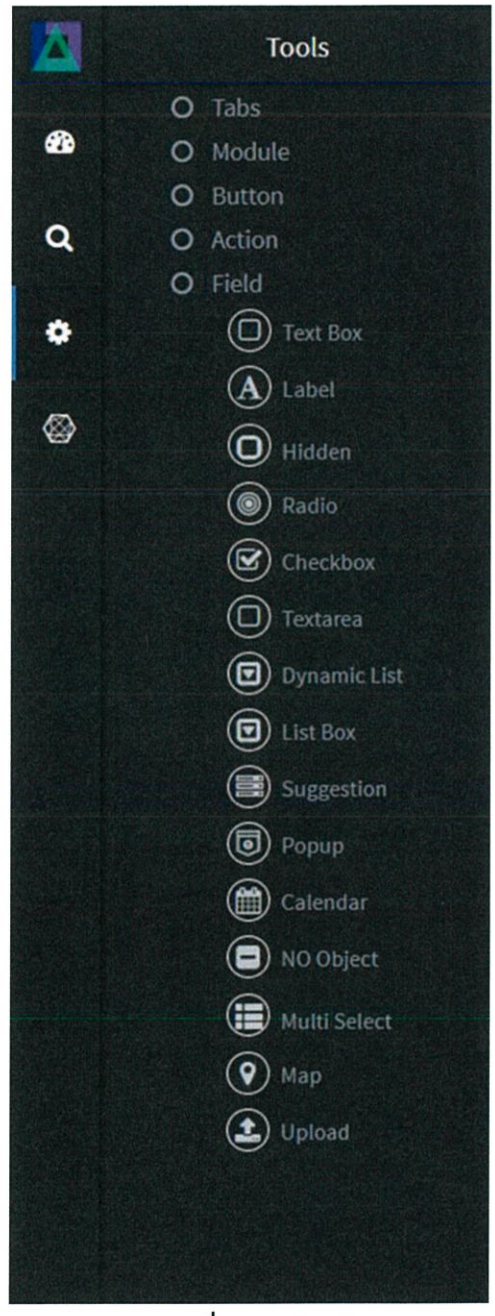
Action คือ ส่วนที่ใช้ผูกหน้าจอกับโค้ด (Customization) เพื่อเพิ่มความสามารถการทำงานที่ตัว ONEWEB ไม่สามารถทำได้ แบ่งเป็น 3 แบบ คือ **Entity Action** โดย Action จะสามารถใช้ได้ทั้ง Entity ครอบคลุมไปถึง Module ต่าง ๆ **Module Action** สำหรับ Action นี้จะส่งผลแค่ใน Module นั้น ๆ และ **JS Editor** เป็นเครื่องมือของ ONEWEB เอง สามารถนำฟิลด์ต่าง ๆ มาสร้างสมการได้โดยไม่ต้องเขียน Code



ภาพที่ 2.14 Action

(6) Field

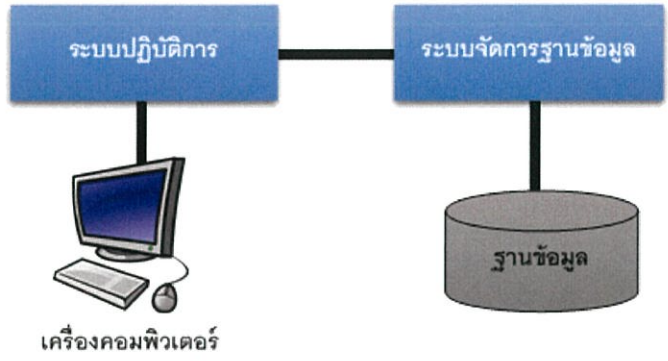
Field คือ ส่วนหนึ่งของระเบียน (record) ซึ่งเก็บข้อมูลเฉพาะชนิดใด ชนิดหนึ่งไว้ ใช้สำหรับเก็บข้อมูลหรือแสดงข้อมูลโดยผูกกับคอลัมน์ในตาราง



ภาพที่ 2.15 Field

2.2.2 ระบบจัดการฐานข้อมูล (Database Management System: DBMS)

ซอฟต์แวร์ที่ดูแลจัดการเกี่ยวกับฐานข้อมูล โดยอำนวยความสะดวกให้แก่ผู้ใช้ทั้งในด้านการสร้าง การปรับปรุงแก้ไข การเข้าถึงข้อมูล และการจัดการเกี่ยวกับระบบแฟ้มข้อมูลทางกายภาพ (Physical file organization)



ภาพที่ 2.16 แสดงโครงสร้างของระบบจัดการฐานข้อมูลและระบบปฏิบัติการ

จากภาพที่ 2.1 จะเห็นได้ว่า ผู้ใช้สามารถเข้าถึงข้อมูลในฐานข้อมูลโดยผ่านระบบจัดการฐานข้อมูล โดยที่ผู้ใช้อาจจะเขียนโปรแกรมประยุกต์หรือใช้ภาษาเรียกค้น ดังนั้นระบบจัดการฐานข้อมูลจะเป็นตัวแยกโปรแกรมออกจากโครงสร้างข้อมูลทางกายภาพ โดยจะทำหน้าที่ติดต่อติดต่อกับระบบปฏิบัติการ (Operating system) ในส่วนของ file management เพื่อดึงฐานข้อมูลในฐานข้อมูลเข้ามาเก็บไว้ใน buffer สำหรับการประมวลผล แต่ก็มีข้อเสียคือ ระบบจัดการฐานข้อมูลประกอบด้วยสิ่งอำนวยความสะดวกและเทคนิคต่าง ๆ มากมาย ทำให้ขนาดใหญ่ใช้ทรัพยากรมาก ทำให้มีราคาแพง และยังต้องอาศัยคนที่มีความรู้เกี่ยวกับเทคโนโลยีของระบบจัดการฐานข้อมูลด้วย

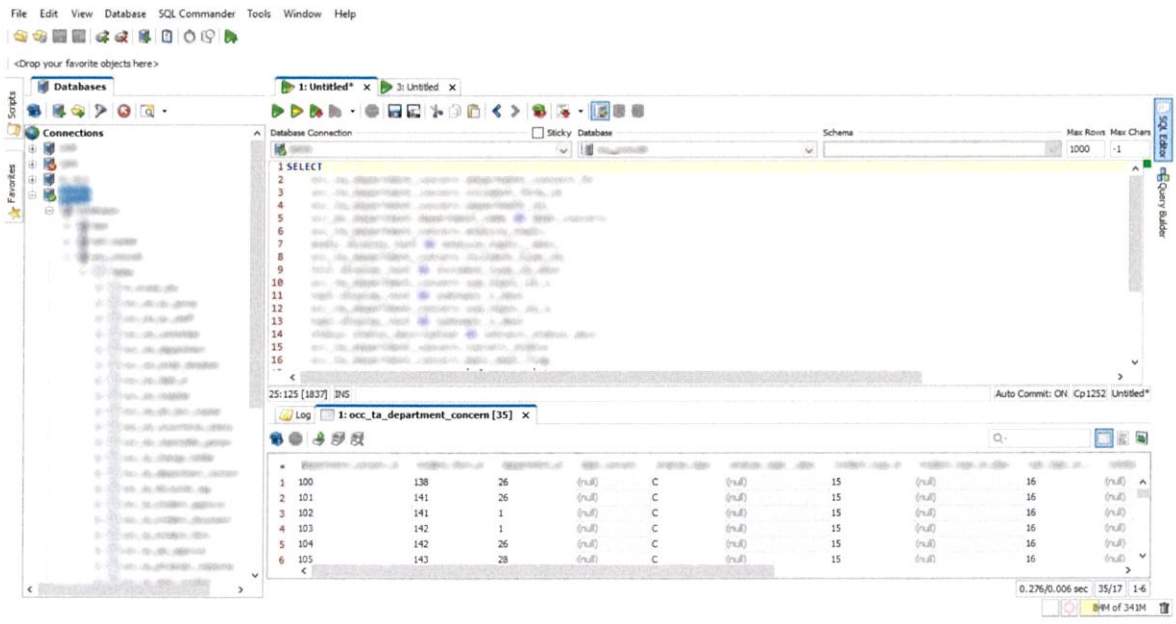
ระบบจัดการฐานข้อมูลช่วยให้การพัฒนาโปรแกรมสามารถใช้งานได้ง่ายขึ้น รวดเร็วขึ้น มีความถูกต้อง และลดค่าใช้จ่าย ระบบจัดการฐานข้อมูลมีตั้งแต่ระดับมาตรฐานที่มีครบตามสถาปัตยกรรม ISO มีระบบดูแลความปลอดภัยที่มีความสามารถสูง มีระบบควบคุมความถูกต้อง มีความเป็นอิสระของข้อมูล ดูแลการใช้ข้อมูลร่วมกันในช่วงเวลาเดียวกัน (concurrency control) มีระบบสำรองข้อมูลและการฟื้นสภาพที่มีประสิทธิภาพ เป็นต้น จนถึงระบบจัดการฐานข้อมูลขนาดเล็กบนเครื่องไมโครคอมพิวเตอร์สำหรับผู้ใช้คนเดียว และมีสถาปัตยกรรมที่พัฒนาจากระบบจัดการแฟ้มข้อมูล (file management) ส่วนใหญ่แล้วระบบจัดการฐานข้อมูลที่ใช้กันในปัจจุบันจะนำเสนอความสัมพันธ์ระหว่างข้อมูลในรูปแบบของตาราง (relation) ซึ่งใช้ง่ายเนื่องจากโครงสร้างข้อมูลไม่สลับซับซ้อนและมีภาษาที่เหมาะสม เช่น SQL เป็นต้น และเนื่องจากไมโครคอมพิวเตอร์มีความสามารถสูงขึ้นจนสามารถใช้ระบบปฏิบัติการของ

เครื่องระดับมินิคอมพิวเตอร์ขึ้นไปดังเช่น Unix ดังนั้นระบบจัดการฐานข้อมูลที่มีความสามารถสูง เช่น ORACLE, SYBASE หรือ INFORMIX เป็นต้น จึงเป็นที่นิยมใช้กันในปัจจุบันนี้

MariaDB คือ MySQL ที่ถูกนำมาแตกแขนงออกมาจากต้นสาย MySQL โดย Michael Widenius ซึ่งเป็นผู้ก่อตั้ง MySQL เดิม ได้แยกตัวโครงการออกมาใหม่ และทำงานภายในชื่อบริษัท Monty Program AB โครงสร้างต่าง ๆ ของ MariaDB แทบจะเรียกได้ว่าเหมือนกับ MySQL เช่น โครงสร้างไฟล์ โครงสร้างของ Database และ Table เพราะฉะนั้นการใช้งาน MariaDB แทนที่ MySQL แทบจะไม่ต้องทำอะไรเพิ่มเติมเลย แม้แต่ Code ของ PHP ก็สามารถที่จะใช้งาน function ของ MySQL ได้เช่นเดิม

2.2.3 โปรแกรมระบบจัดการฐานข้อมูล

DbVisualizer คือ โปรแกรมที่ใช้บริหารจัดการฐานข้อมูล เนื่องจากถ้าเราใช้ฐานข้อมูลเป็น PostgreSQL บางครั้งจะมีความลำบากและยุ่งยากในการใช้งาน ดังนั้นจึงมีเครื่องมือในการจัดการฐานข้อมูล PostgreSQL ขึ้นมาเพื่อให้สามารถจัดการระบบจัดการฐานข้อมูล เรียกดูข้อมูลได้ง่ายและสะดวกยิ่งขึ้น โดย DbVisualizer ก็ถือเป็นเครื่องมือชนิดหนึ่งในการจัดการนั่นเอง

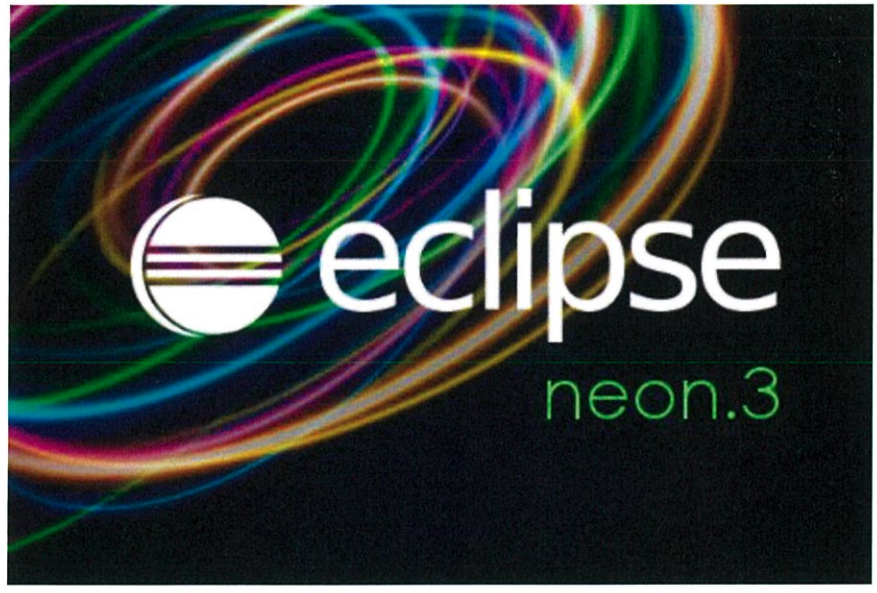


ภาพที่ 2.17 โปรแกรม DbVisualizer Pro

2.2.4 โปรแกรม Eclipse

Eclipse คือ โปรแกรมที่ใช้สำหรับพัฒนาภาษาจาวา (Java) ซึ่งโปรแกรม Eclipse เป็นโปรแกรมหนึ่งที่ใช้ในการพัฒนา Application Server ได้อย่างมีประสิทธิภาพ และเนื่องจาก Eclipse เป็นซอฟต์แวร์ Open Source ที่พัฒนาขึ้นเพื่อใช้โดยนักพัฒนาเอง ทำให้ความก้าวหน้าในการพัฒนาของ Eclipse เป็นไปอย่างต่อเนื่องและรวดเร็ว [13]

Eclipse มีองค์ประกอบหลักที่เรียกว่า Eclipse Platform ซึ่งให้บริการพื้นฐานหลักสำหรับรวบรวมเครื่องมือต่าง ๆ จากภายนอกให้สามารถเข้ามาทำงานร่วมกันในสภาพแวดล้อมเดียวกัน และมีองค์ประกอบที่เรียกว่า Plug-in Development Environment (PDE) ซึ่งใช้ในการเพิ่มความสามารถในการพัฒนาซอฟต์แวร์มากขึ้น เครื่องมือภายนอกจะถูกพัฒนาในรูปแบบที่เรียกว่า Eclipse plug-ins ดังนั้นหากต้องการให้ Eclipse ทำงานใดเพิ่มเติม ก็เพียงแต่พัฒนา plugin สำหรับงานนั้นขึ้นมา และนำ Plug-in นั้นมาติดตั้งเพิ่มเติมให้กับ Eclipse ที่มีอยู่เท่านั้น Eclipse Plug-in ที่มีมาพร้อมกับ Eclipse เมื่อเรา download มาครั้งแรกก็คือองค์ประกอบที่เรียกว่า Java Development Toolkit (JDT) ซึ่งเป็นเครื่องมือในการเขียนและ Debug โปรแกรมภาษา Java



ภาพที่ 2.18 Eclipse neon.3

2.2.5 GitLab

ในการพัฒนาซอฟต์แวร์เราจำเป็นต้องเก็บซอร์สโค้ดไว้ในเวอร์ชันคอนโทรล (version control) เพื่อที่บันทึกการเปลี่ยนแปลงต่างๆ ในซอร์สโค้ดของเราและการทำงานร่วมกับทุกคนในทีม ซึ่งซอฟต์แวร์ประเภทเวอร์ชันคอนโทรลที่นิยมกันในปัจจุบันนั้นก็คือ Git และก็มีคลาวด์เซอร์วิสที่ควบคุมการทำงาน Git อีกทีเพื่อเราทำงานกับคนในทีมได้สะดวกยิ่งขึ้นด้วย เพราะฉะนั้นจึงมีคนคิดค้นให้เซอร์วิสพวกนี้ให้ทำงานได้สำหรับองค์กร ซึ่งมีชื่อว่า GitLab [14]

ภาพที่ 2.19 GitLab

2.2.6 Chrome Developer Tools

Chrome Developer Tools คือ เครื่องมือสำหรับนักพัฒนาเว็บแอปพลิเคชัน เพื่อตรวจสอบ และแก้ไขจุดบกพร่อง ของเว็บแอปพลิเคชันของเรา ซึ่งเราสามารถใช้งานเครื่องมือสำหรับนักพัฒนา เพื่อติดตามปัญหารูปแบบ (Inspect element) การแก้ปัญหาโดยใช้ JavaScript โดยสามารถเลือกเมนู Tools เครื่องมือสำหรับนักพัฒนา หรือกด ctrl + shift + i หรือเพียงกด F12 เพื่อเปิดใช้งานเครื่องมือ Developer tools [8]

องค์ประกอบของเครื่องมือ และรายละเอียดเฉพาะส่วนที่ได้ใช้งาน [11]

- Elements (องค์ประกอบหน้าเว็บแอปพลิเคชัน)

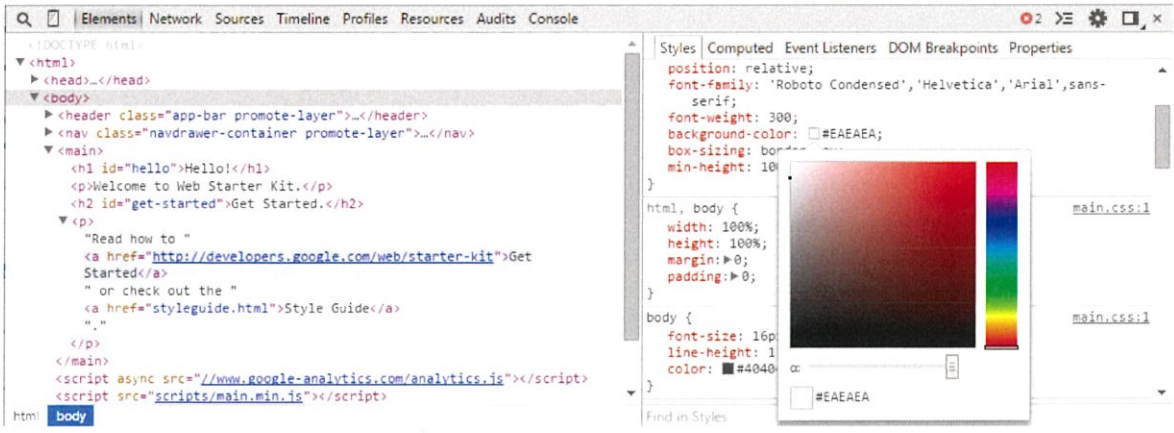
สามารถมองเห็นโครงสร้างหน้าเว็บเป็น HTML Source Code, CSS Styles รวมถึง JavaScript ด้วย โดยการแสดงผลก็จะอยู่ในรูปแบบของ Document Object Model

- Source (แหล่งที่มา)

สามารถระบุให้ทราบถึงแหล่งที่มาของไฟล์เอกสาร Source Code ต่างๆ เช่น HTML CSS และ JS ว่าได้ถูกโหลดขึ้นมา หรือเรียกใช้งานมาจากไดเรคทอรี่ไหน และยังสามารถแก้ไขไฟล์ Source Code นั้นได้ ซึ่งสามารถลากหรือวางไฟล์จากตำแหน่งของแฟ้มที่จะแก้ไขในระหว่างการทำงานได้ ซึ่งจะทำงานเหมือน IDE

- Console (คอนโซลตรวจสอบองค์ประกอบ)

สามารถใช้ในการดูการทำงานระหว่าง Client และ Server ได้ ซึ่งหากพบข้อผิดพลาดก็จะแจ้งเตือนข้อผิดพลาดให้ทราบ รวมถึงการแจ้งเตือน Warning ในส่วนที่บกพร่องของเว็บแอปพลิเคชันที่พัฒนา



ภาพที่ 2.20 Chrome Developer Tools

2.2.7 โปรแกรม SourceTree

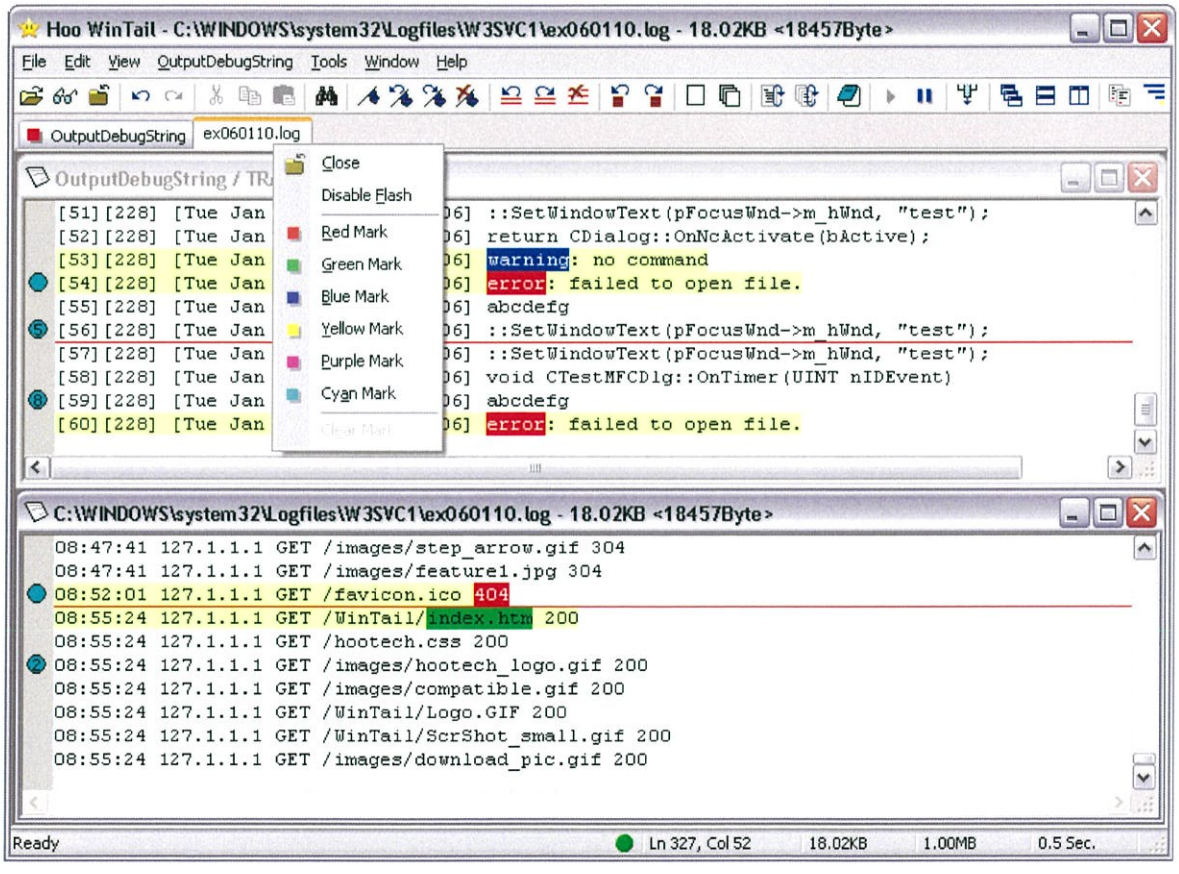
SourceTree คือ ซอฟต์แวร์ที่ช่วยจัดการ Git repository โดย Git เป็นระบบที่มีหน้าที่ในการจัดเก็บการเปลี่ยนแปลงของไฟล์ในโปรเจกต์ มีการสำรองโค้ดและสามารถที่จะเรียกดูหรือย้อนกลับไปดูเวอร์ชันต่าง ๆ ของโปรเจกต์ที่ใด เวลาใดก็ได้ หรือแม้แต่ดูว่าไฟล์นั้น ๆ ใครเป็นคนเพิ่มหรือแก้ไข หรือว่าจะดูว่าไฟล์นั้น ๆ ถูกเขียนโดยใครบ้างก็สามารถทำได้ ฉะนั้น Git จึงเหมาะอย่างยิ่งสำหรับนักพัฒนาไม่ว่าจะเป็นคนเดียวโดยเฉพาะอย่างยิ่งจะมีประสิทธิภาพมากหากเป็นการพัฒนาเป็นทีม

ภาพที่ 2.21 SourceTree

2.2.8 โปรแกรม Hoo WinTail

Hoo WinTail คือ โปรแกรมมอนิเตอร์ Log แบบเรียลไทม์และดู Log สำหรับ Windows สามารถแสดง OutputDebugString โดยมีคุณสมบัติเหมาะสำหรับการแก้จุดบกพร่องของโปรแกรมและการบริหารระบบ

ในการทำงานที่เข้าผ่าน Server บนเครื่องเราเองแล้วเกิดข้อผิดพลาดขึ้น จึงจำเป็นต้องทราบ Log เพื่อดูค่า Error ที่เกิดขึ้นว่าเกิดขึ้นจากอะไร โดยโปรแกรม Hoo WinTail สามารถเปิดไฟล์ Log ที่ Server กำลังทำงานได้ โดยไม่ต้องดูใน Console หรือใช้ Notepad เพื่อเปิดไฟล์ Server.log สามารถกดหยุดค้างไว้ (Pause Monitoring) หรือ ให้มันทำงานต่อได้ (Start Monitoring) หรือถ้าต้องการตรวจค่าเป็นเป็นปัจจุบัน ก็สามารถ Clear Log ที่ค้างไว้ได้เช่นกัน



ภาพที่ 2.22 Hoo WinTail

บทที่ 3
วิธีดำเนินงาน

การดำเนินโครงการในการทำสหกิจศึกษาครั้งนี้ได้รับมอบหมายให้พัฒนาหน้าจอของระบบบริหาร-จัดการอุบัติเหตุในโรงพยาบาลให้ถูกต้องตามเอกสารการออกแบบ ด้วยซอฟต์แวร์ ONEWEB 4.0 และการ Customization โดยสามารถแบ่งเป็นขั้นตอนได้ดังนี้

3.1 ศึกษาและเรียนรู้ข้อมูลที่จำเป็นต่อการทำงาน

3.1.1 ศึกษาภาษาพื้นฐานการเขียนโปรแกรม Java JavaScript jQuery HTML และ CSS

3.1.2 ศึกษาซอฟต์แวร์ ONEWEB 4.0 การจัดวางหน้าจอ ฟังก์ชันต่าง ๆ ของ ONEWEB การ Customization บนหน้าเว็บเพิ่มเติม โดยใช้ภาษา Java JavaScript และ jQuery สำหรับพัฒนาระบบงาน

3.1.3 ศึกษาฐานข้อมูลที่ใช้งาน ข้อมูลมาสเตอร์ในฐานข้อมูลต่าง ๆ และการเขียน Query ข้อมูล

3.1.4 ขั้นตอนการพัฒนา Configuration ONEWEB บน Server กลาง รวมถึงการเขียน Code ในส่วน Customization เพิ่มเติม การดูข้อมูลในฐานข้อมูล และการ Merge Code บน GitLab

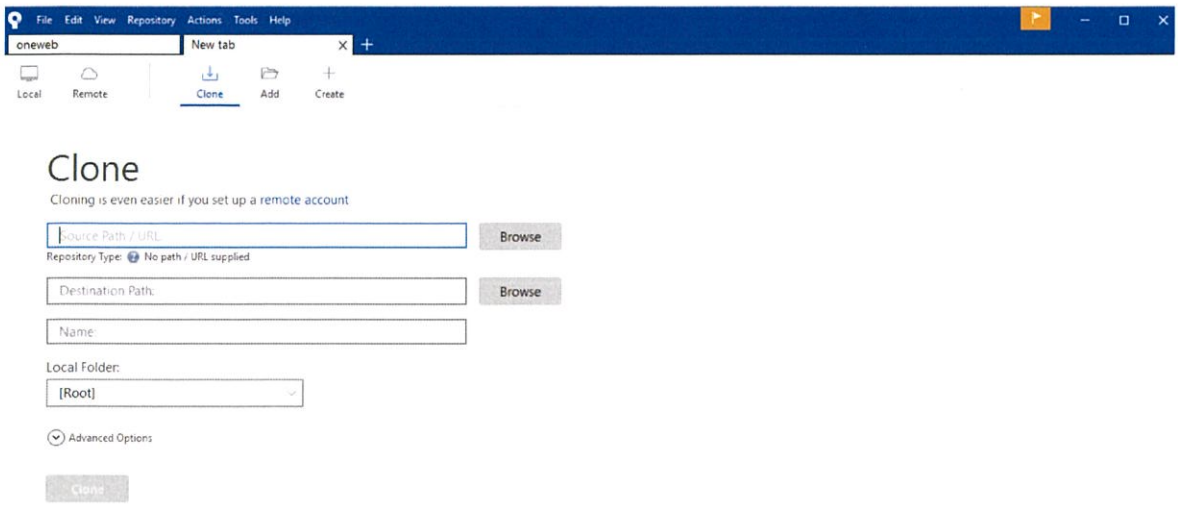
3.2 เตรียมเครื่องมือหรือโปรแกรมที่ใช้ในการทำงาน

โปรแกรมที่ใช้งานมีโปรแกรม Eclipse สำหรับเขียนไฟล์ Java JavaScript JSP และเปิด Server บนเครื่องตัวเอง สาเหตุที่จำเป็นต้องเปิดเซิร์ฟเวอร์บนเครื่องคอมพิวเตอร์ส่วนตัวก็เพราะ เมื่อต้องมีการ Customization ที่นอกเหนือจากการ Configuration บน ONEWEB ส่วนของ Code ที่เราเขียนขึ้นมาจะทำงานบนเครื่องก่อนเท่านั้น ถ้าสามารถใช้งานได้ เราจึงใช้โปรแกรม SourceTree ในการ Commit Code ขึ้นไปรวมกับโค้ดส่วนกลาง โปรแกรม DbVisualizer เป็นโปรแกรมสำหรับดูข้อมูลที่อยู่บนฐานข้อมูล และทดลองเขียน SQL ก่อนนำไปใช้จริง และสุดท้าย โปรแกรม Hoo WinTail สำหรับตรวจสอบ Log ของเซิร์ฟเวอร์บนเครื่องคอมพิวเตอร์ส่วนตัว เพื่อตรวจสอบความผิดพลาดของระบบ

สำหรับขั้นตอนการเตรียมโปรแกรม มีดังนี้

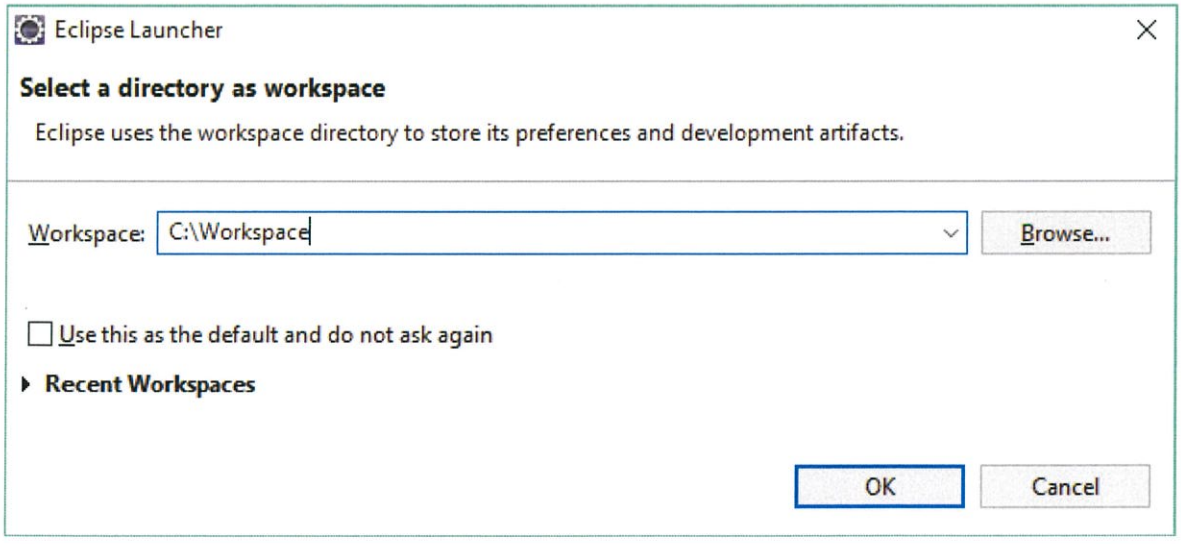
1. สมัคร GitLab ของโปรเจกต์ ซึ่งลิงก์ GitLab จะได้จากทีมงาน จากนั้นก็ให้ทีมงานที่ดูแลโปรเจกต์เชิญเข้าโปรเจกต์ เพื่อสามารถเข้าถึงไฟล์ต่าง ๆ ได้

2. ลงโปรแกรม SourceTree สามารถโหลดได้ที่เว็บไซต์หลัก จากนั้นเชื่อมต่อ Git โดยใช้ URL ของ Git Repository ของโปรเจกต์ เพื่อ Clone Workspace มาเก็บในเครื่องคอมพิวเตอร์ส่วนตัว จากนั้นก็รอจนกว่าจะ Clone สำเร็จ



ภาพที่ 3.1 หน้าโปรแกรม SourceTree สำหรับ Clone Workspace

3. เมื่อได้ Workspace ที่ Clone มาแล้ว ก็ลงโปรแกรม Eclipse โดยเวอร์ชันที่ใช้เป็น Eclipse IDE Java EE Developers ซึ่งจะแตกต่างกับเวอร์ชันหลัก โดยเมื่อเปิดโปรแกรม Eclipse ขึ้นมา ก็ให้เลือก Workspace ตามที่อยู่ที่ได้ Clone เก็บไว้

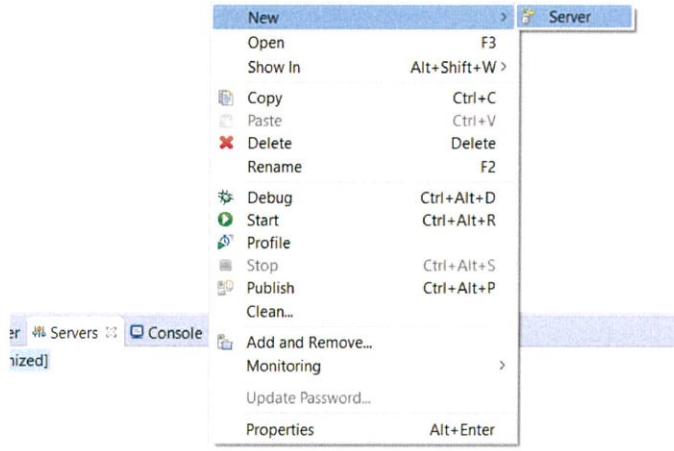


ภาพที่ 3.2 หน้าต่าง Eclipse สำหรับเลือก Workspace

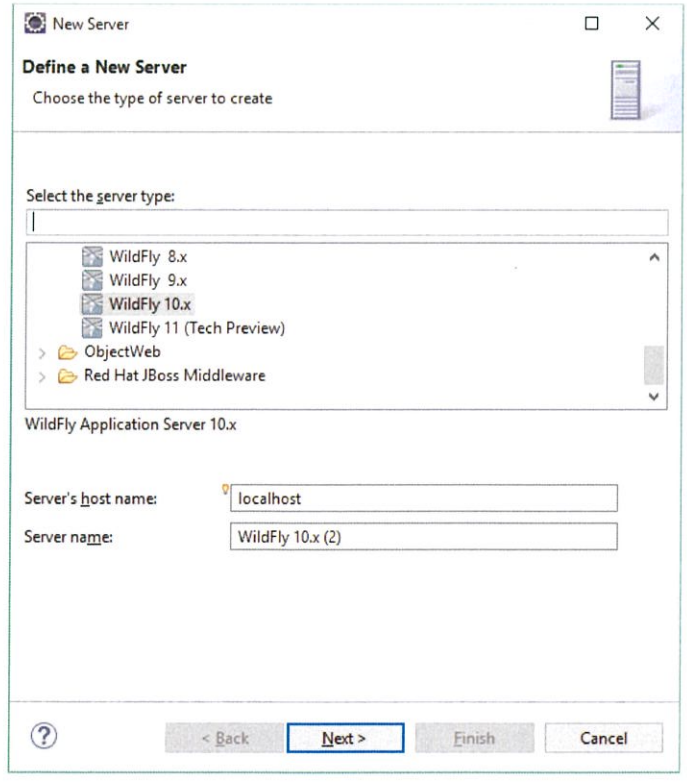
4. ลง WildFly 10.x สำหรับเปิดเซิร์ฟเวอร์เมื่อลงโปรเจกต์ ทางโปรเจกต์แต่ละโปรเจกต์ก็จะใช้ Server ที่แตกต่างกันไป สำหรับโปรเจกต์ที่ใช้ทำงานครั้งนี้ใช้เซิร์ฟเวอร์ ของ WildFly โดยทาง

โปรเจกต์จะให้โฟลเดอร์สำหรับที่อยู่ของเซิร์ฟเวอร์และเมื่อเซ็ตค่าเสร็จเรียบร้อยแล้วก็เพิ่มเซิร์ฟเวอร์ไปยัง Eclipse ขั้นตอนดังนี้

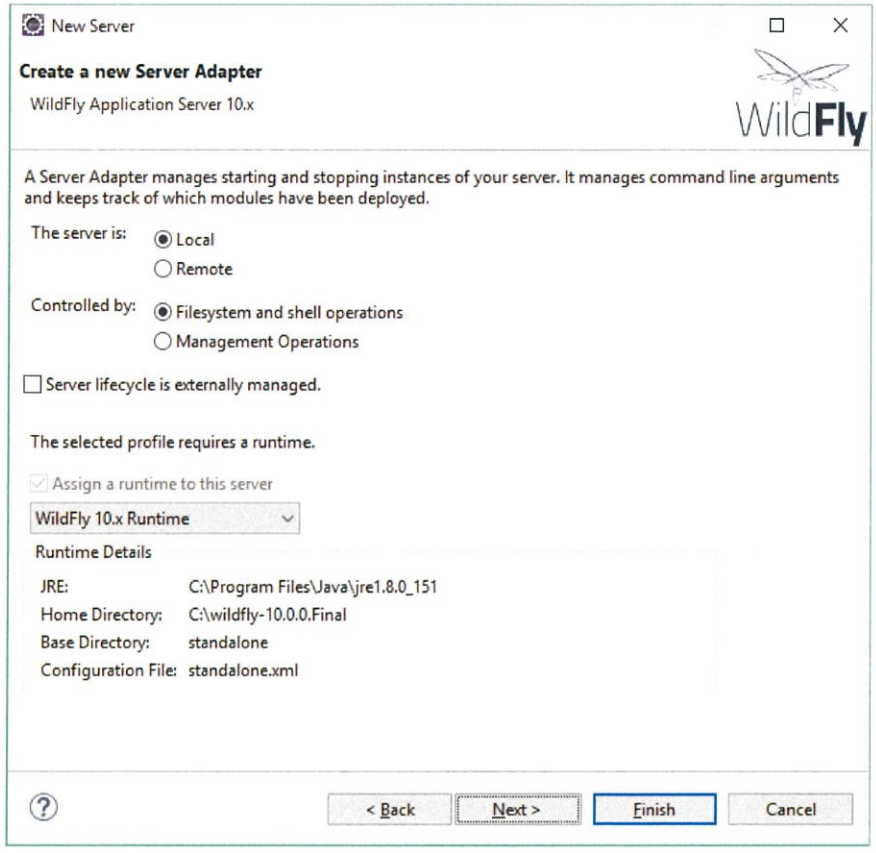
คลิกขวาที่แท็บ Server จากนั้นเลือก New > Server ถ้ายังไม่ขึ้นแท็บ Server ด้านล่างให้ไปที่แถบเครื่องมือด้านบน Window > Show view > Server



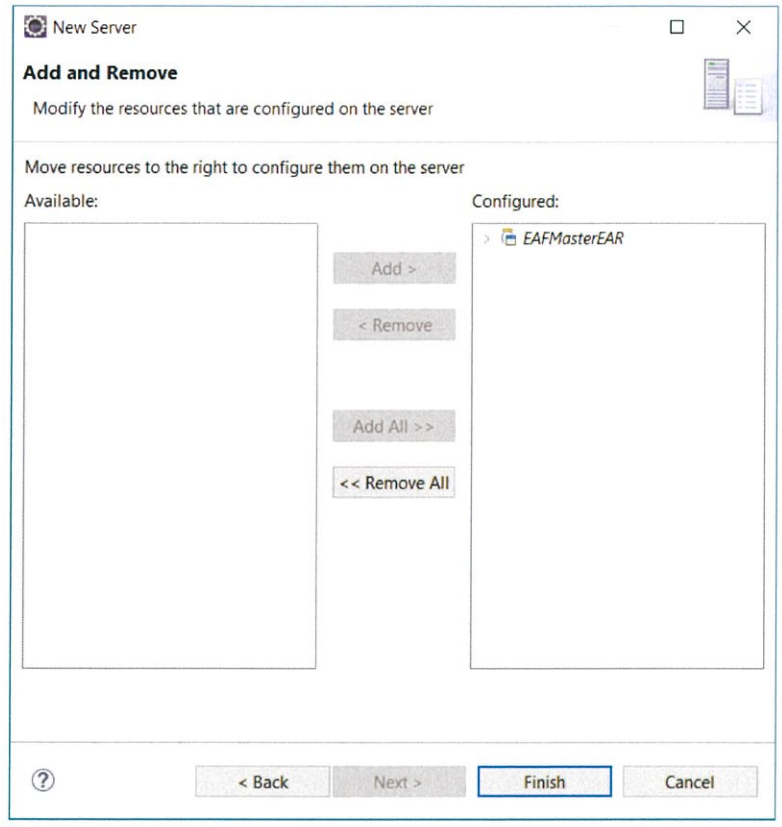
ภาพที่ 3.3 ขั้นตอนการ Setting Server ที่ 1



ภาพที่ 3.4 ขั้นตอนการ Setting Server ที่ 2

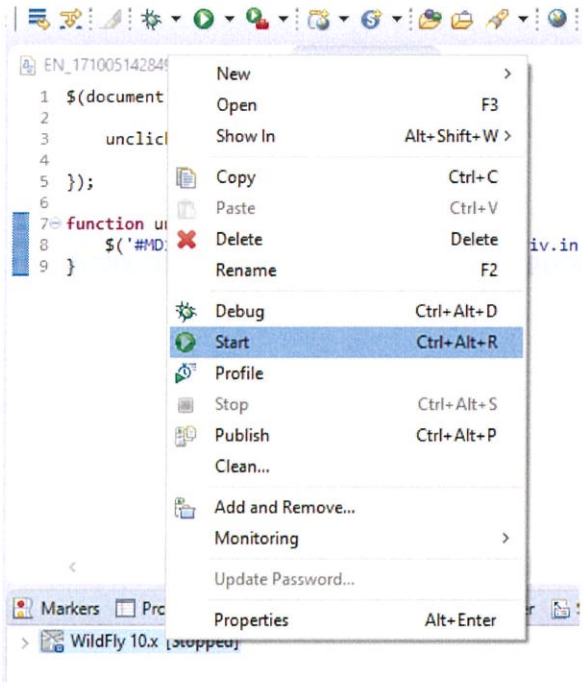


ภาพที่ 3.5 ขั้นตอนการ Setting Server ที่ 3



ภาพที่ 3.6 ขั้นตอนการ Setting Server ที่ 4

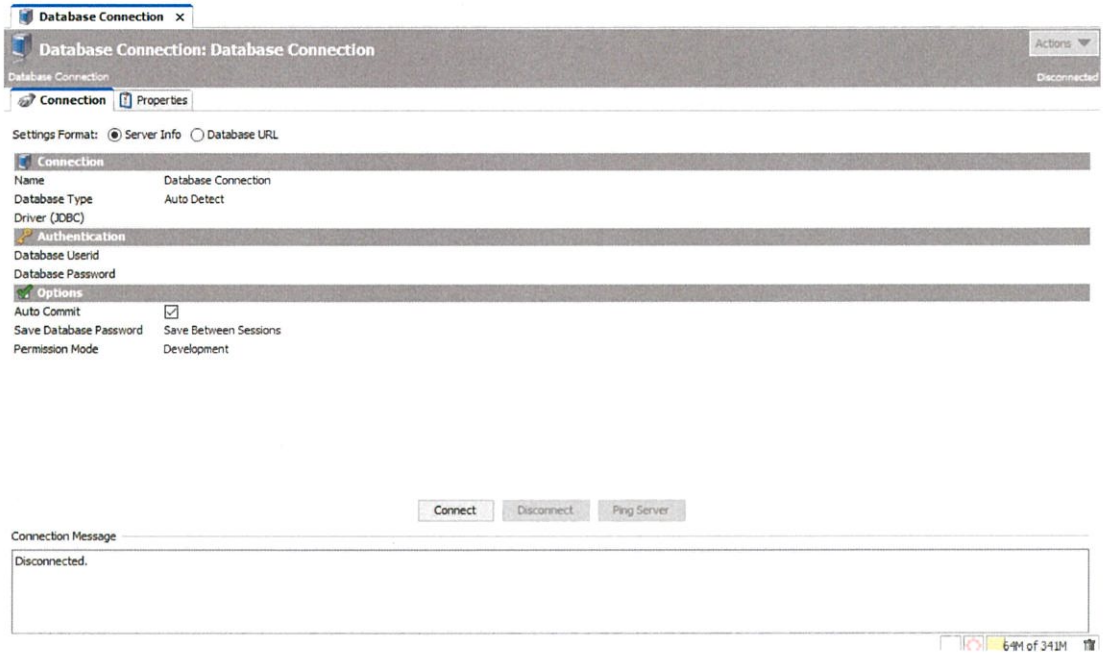
5. Config Server โดยการเริ่ม Start Server จากนั้นเข้าหน้าล็อกอิน ONEWEB ด้วยลิงก์ localhost:8080/FrontWeb/index.jsp และล็อกอินโดยใช้รหัสสำหรับนักพัฒนาของโปรเจกต์



ภาพที่ 3.7 การ Start Server

ภาพที่ 3.8 หน้าจอล็อกอินเข้าสู่ระบบ ONEWEB 4.0

6. ลงโปรแกรม DbVisualizer และเชื่อมต่อกับฐานข้อมูล (Database) ของโปรเจกต์ โดยเริ่มจาก คลิกขวาที่หน้าต่าง Connections > Create Database Connection แล้วเลือก No Wizard จากนั้นก็ใส่ที่อยู่สำหรับเชื่อมต่อฐานข้อมูลภาษา Database ที่ใช้ ซึ่งโปรเจกต์นี้ใช้ MariaDB แล้วจึงกรอก Username และ Password สำหรับเข้าถึงฐานข้อมูล



ภาพที่ 3.9 หน้าจอ Create Database Connection

3.3 พัฒนาและปรับปรุงระบบให้เป็นไปตามกำหนด

ขั้นตอนในการพัฒนาหน้าจอของระบบงาน มีดังนี้

1. ออกแบบหน้าจอตามความต้องการ และบันทึกลงเอกสารประกอบการออกแบบ CSD (Customization Specifications Document)

2. พัฒนาหน้าจอระบบงานตามเอกสารประกอบการออกแบบ

3. ผู้พัฒนาทดสอบการใช้งานเบื้องต้น (Unit Test)

4. เมื่อผ่านการทดสอบ ปรับปรุงความเข้ากันไปยังเซิร์ฟเวอร์ SIT (System Integration Testing)

5. ผู้ทดสอบระบบ ทดสอบการใช้งาน หากผ่านการทดสอบข้ามไปข้อ 11.

6. เมื่อไม่ผ่านการทดสอบ ผู้ทดสอบระบบจะเปิดข้อปัญหา

7. ผู้ทดสอบระบบจะมอบหมายงานที่ต้องแก้ไขให้กับนักพัฒนา

8. นักพัฒนาแก้ไขข้อบกพร่องของระบบ

9. ปรับปรุงความเข้ากันไปยังเซิร์ฟเวอร์ที่ผิดพลาด

10. เมื่อทดสอบอีกครั้งไม่ผ่าน ผู้ทดสอบระบบจะแจ้งกลับไปที่ผู้พัฒนา กลับไปข้อ 7.

11. เมื่อผ่านการทดสอบ ปรับปรุงความเข้ากันไปยังเซิร์ฟเวอร์ UAT (User Acceptance Testing)

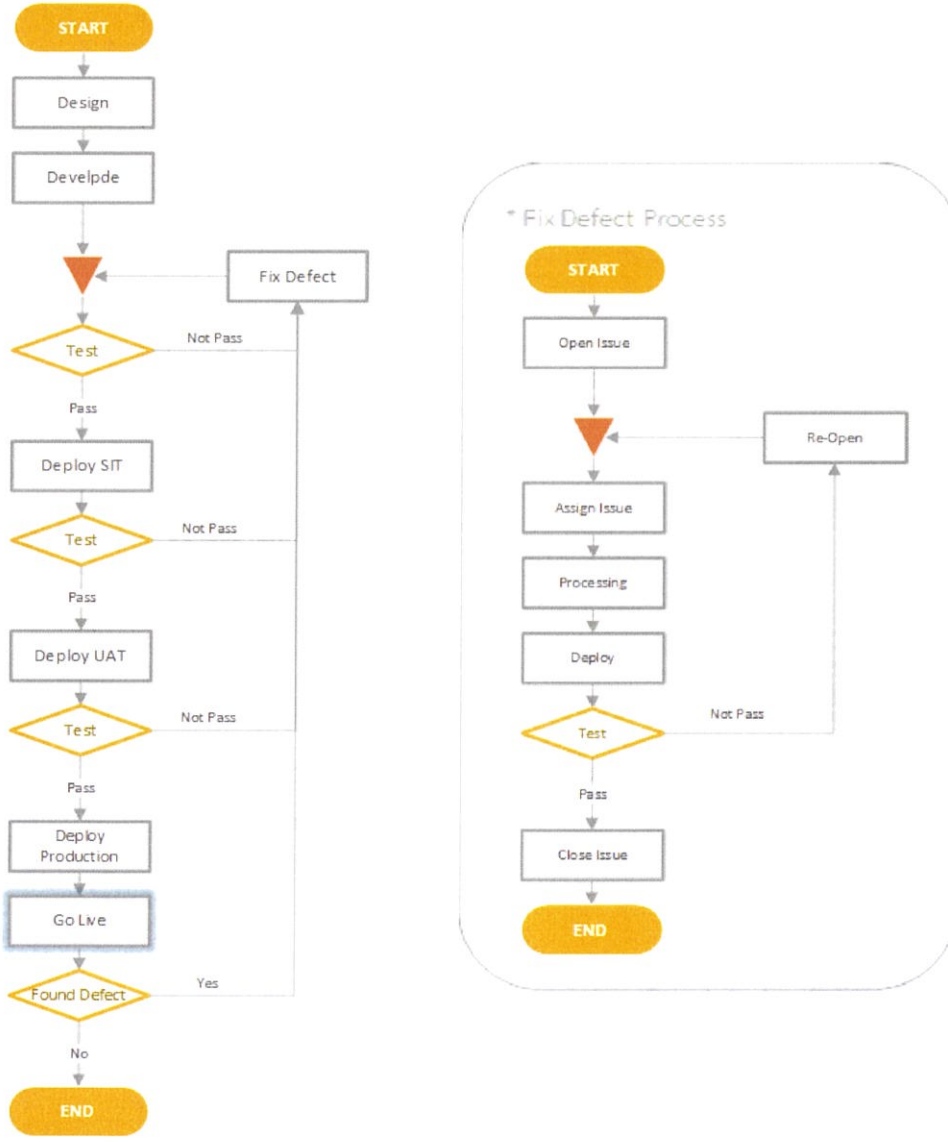
12. ผู้ทดสอบระบบ ทดสอบการใช้งาน หากไม่ผ่านการทดสอบ กลับไปข้อ 6.

13. เมื่อผ่านการทดสอบ ปรับปรุงความเข้ากันไปยังเซิร์ฟเวอร์ Production

14. เริ่มใช้งานระบบจริง

15. ถ้าเจอผู้ใช้ข้อบกพร่อง ผู้ใช้จะแจ้งกลับมาให้แก้ไขปัญหา กลับไปที่ข้อ 6.

16. สิ้นสุดการพัฒนาระบบงาน



ภาพที่ 3.10 ลำดับขั้นตอนการทำงาน

บทที่ 4
ผลการดำเนินงาน

ผลการดำเนินงานสามารถแสดงให้เห็นออกมาในรูปแบบของอินเตอร์เฟส (Interface) ของระบบบริหารจัดการอุบัติเหตุในโรงพยาบาล แต่ละหน้าจะมีส่วนประกอบและฟังก์ชันต่างกันเป็นไปตามเอกสารประกอบการออกแบบ (Customization Specifications Document)

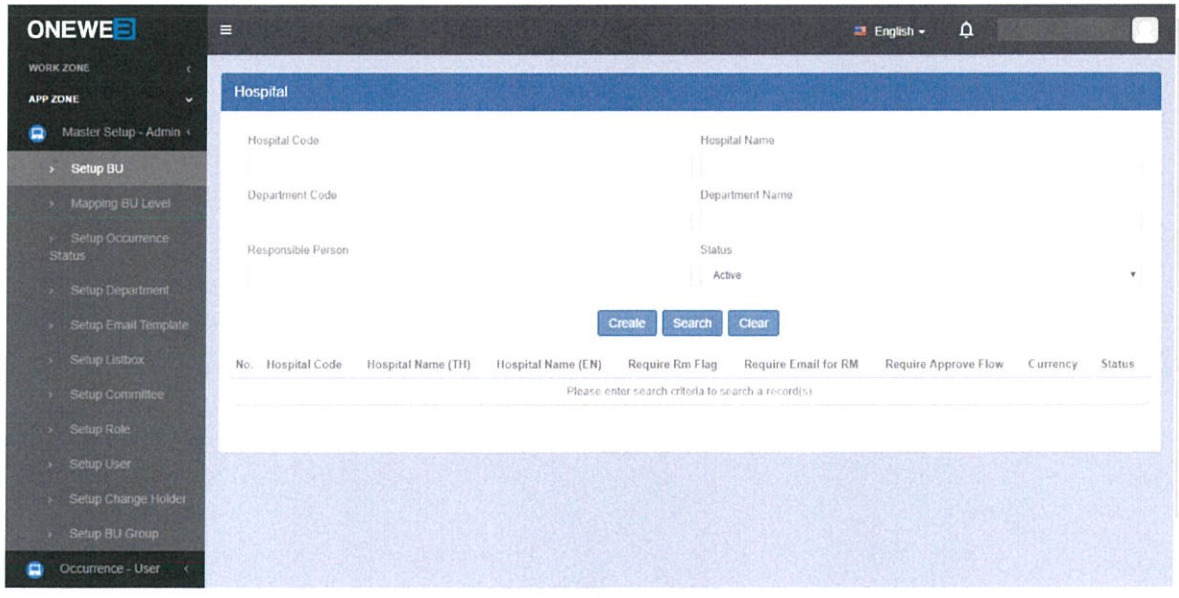
ในหน้าจอระบบหนึ่งหน้าจะประกอบไปด้วยหน้าจอของค้นหา เพิ่มข้อมูล และแก้ไขข้อมูล (Search, Insert and Update) แต่ละหน้าเริ่มต้นจะเป็นหน้าค้นหาข้อมูล ประกอบไปด้วย ฟิลด์ช่องค้นหา (Search Criteria) ปุ่มค้นหา (Search) ปุ่มเพิ่มข้อมูล (Create) ปุ่มล้างการค้นหา (Clear) และรายการข้อมูล (Show Search)

4.1 หน้าจอสำหรับข้อมูลมาสเตอร์ ส่วนผู้ดูแลระบบ (Master Setup - Admin)

หน้าจอสำหรับข้อมูลมาสเตอร์มีไว้สำหรับให้ผู้ดูแลระบบ ทำการเพิ่ม แก้ไข ลบ และค้นหาข้อมูลหลักของระบบในฐานข้อมูลได้

4.1.1 หน้าจอ Setup BU

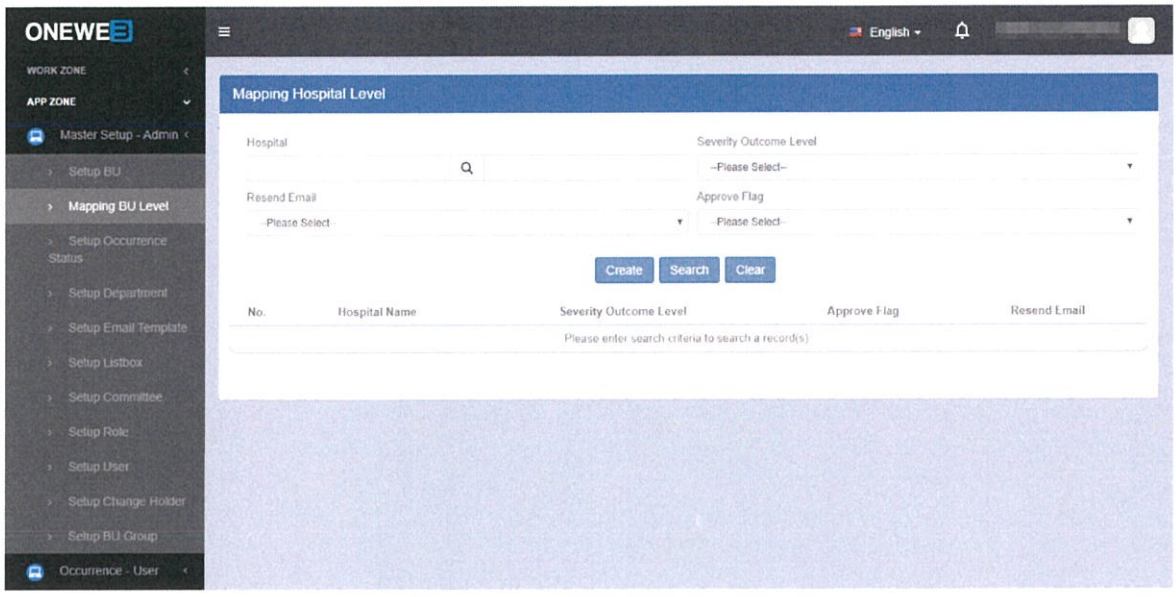
หน้าจอ Setup BU จัดการข้อมูลรายชื่อโรงพยาบาล



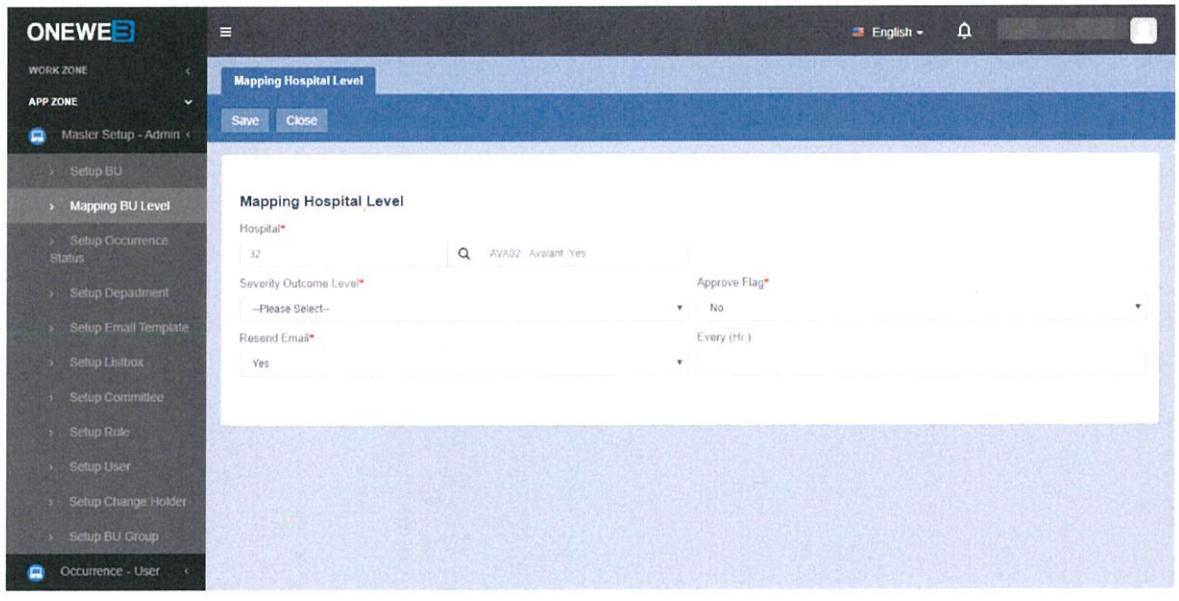
ภาพที่ 4.1 หน้า Search Setup BU

4.1.2 หน้าจอ Mapping BU Level

หน้าจอ Mapping BU Level จัดการข้อมูลระดับความรุนแรงของเหตุการณ์



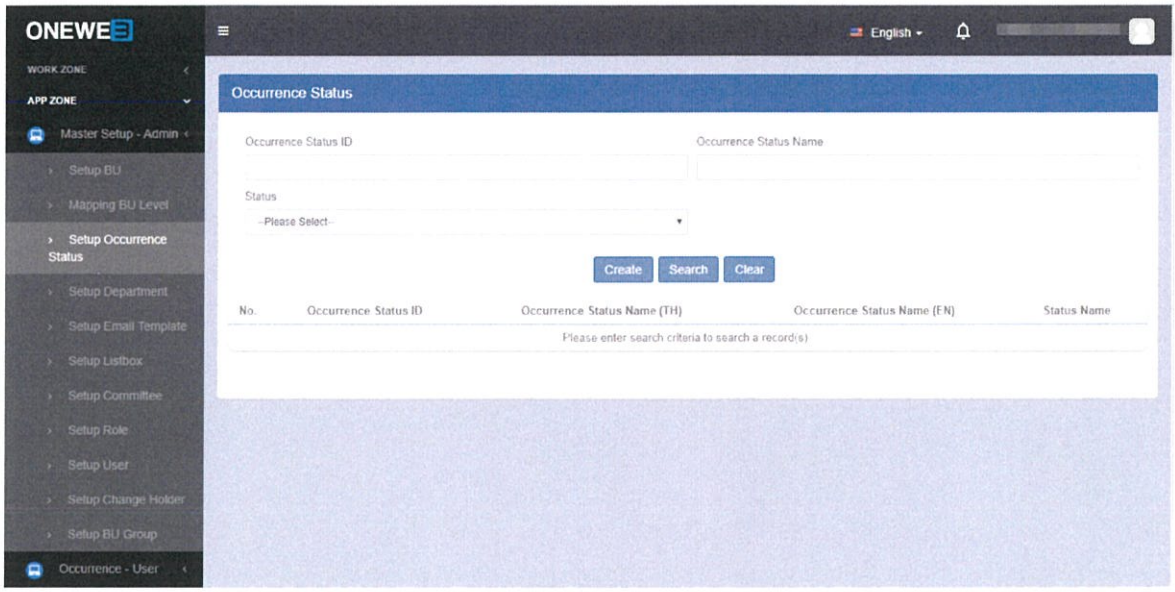
ภาพที่ 4.2 หน้าจอ Search Mapping BU Level



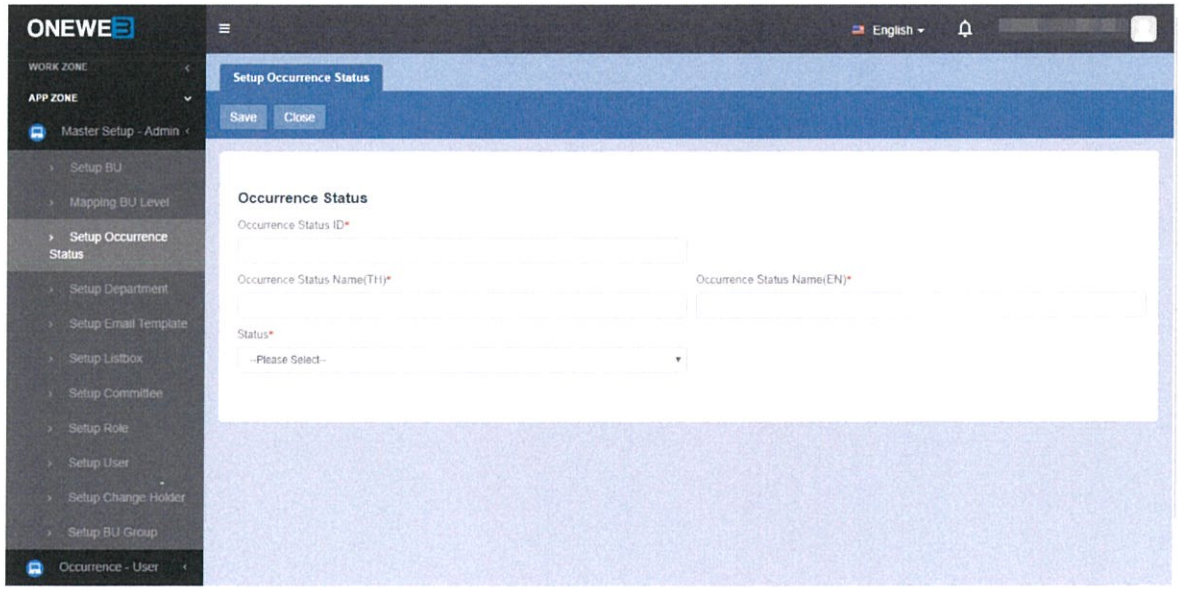
ภาพที่ 4.3 หน้าจอ Insert/Update Mapping BU Level

4.1.3 หน้าจอ Setup Occurrence Status

หน้าจอ Setup Occurrence Status จัดการข้อมูลสถานะของเหตุการณ์



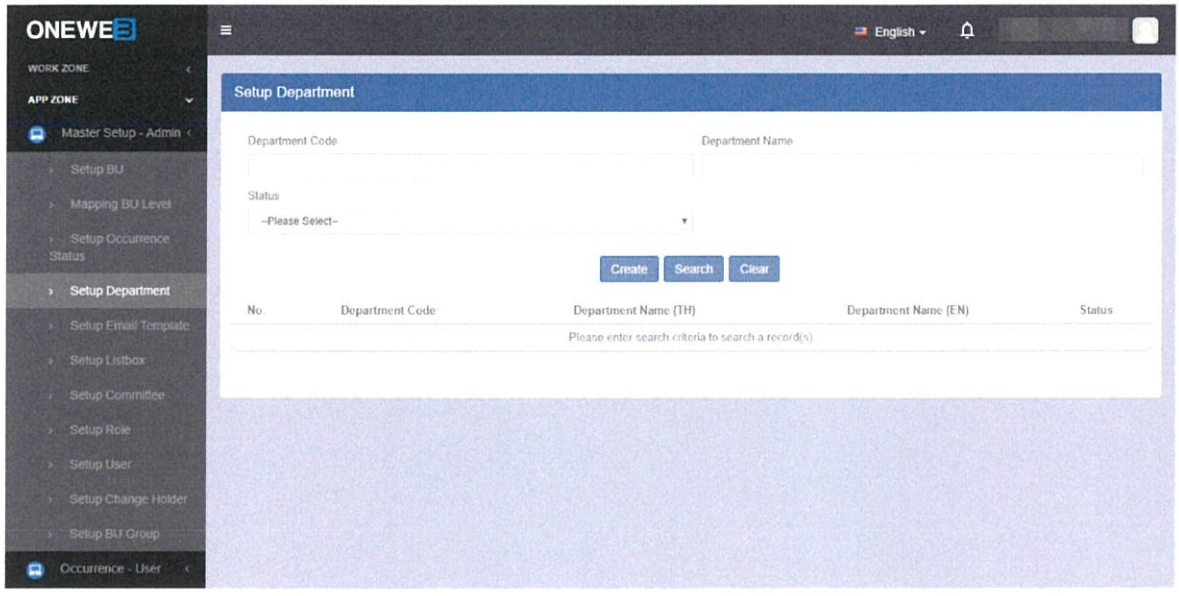
ภาพที่ 4.4 หน้าจอ Search Setup Occurrence Status



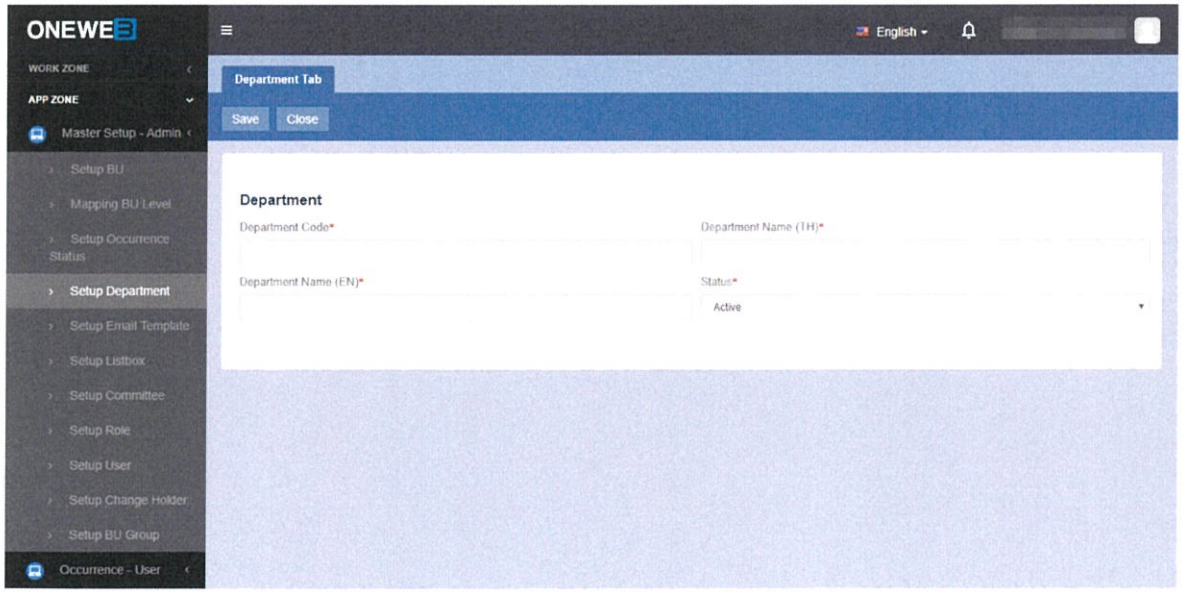
ภาพที่ 4.5 หน้าจอ Insert/Update Setup Occurrence Status

4.1.4 หน้าจอ Setup Department

หน้าจอ Setup Department จัดการข้อมูลแผนกในโรงพยาบาล



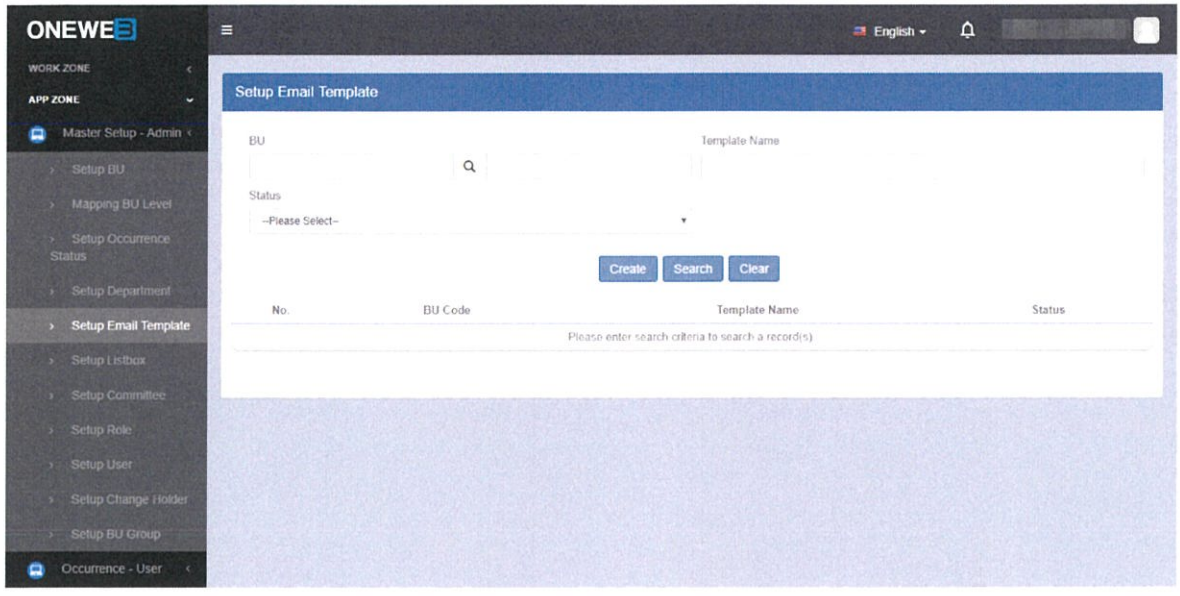
ภาพที่ 4.6 หน้าจอ Search Setup Department



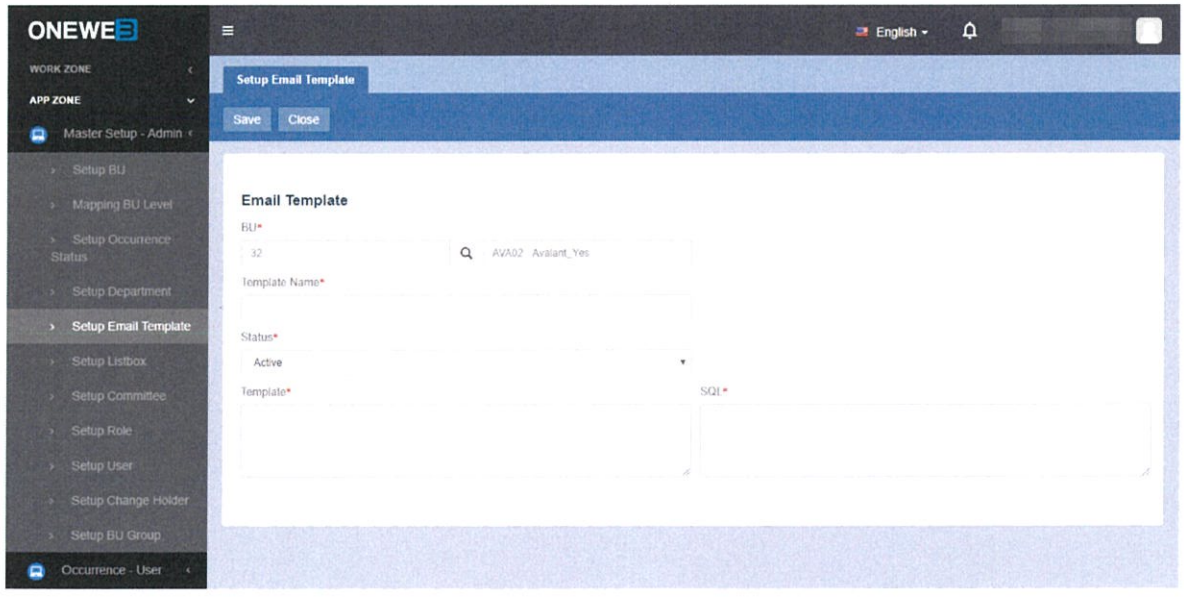
ภาพที่ 4.7 หน้าจอ Insert/Update Setup Department

4.1.5 หน้าจอ Setup Email Template

หน้าจอ Setup Email Template จัดการข้อมูล Template ของ Email



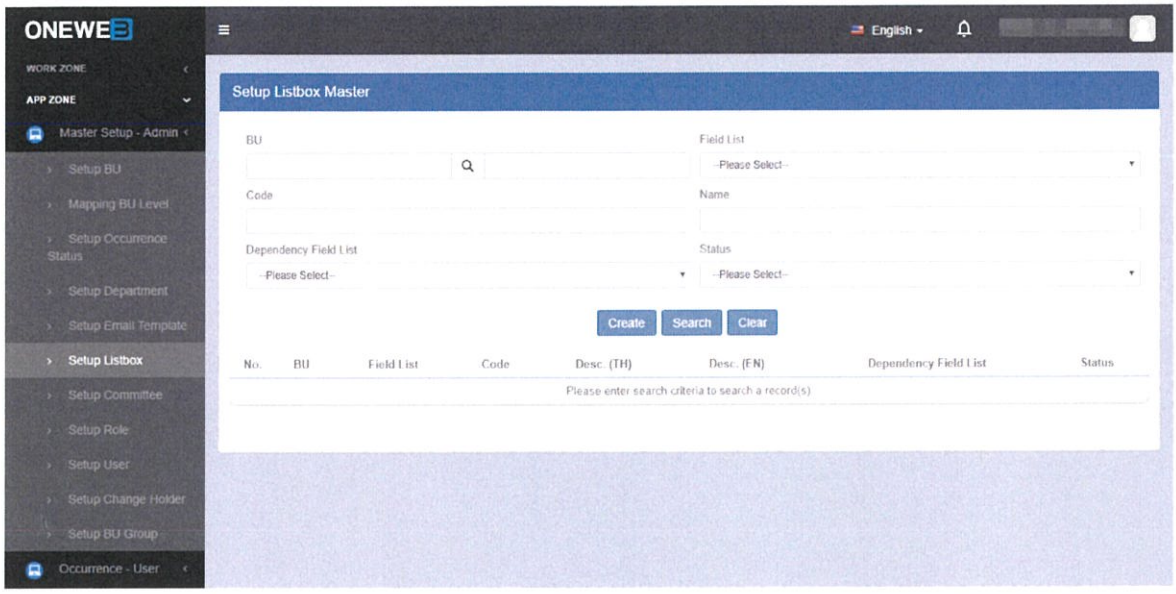
ภาพที่ 4.8 หน้าจอ Search Setup Email Template



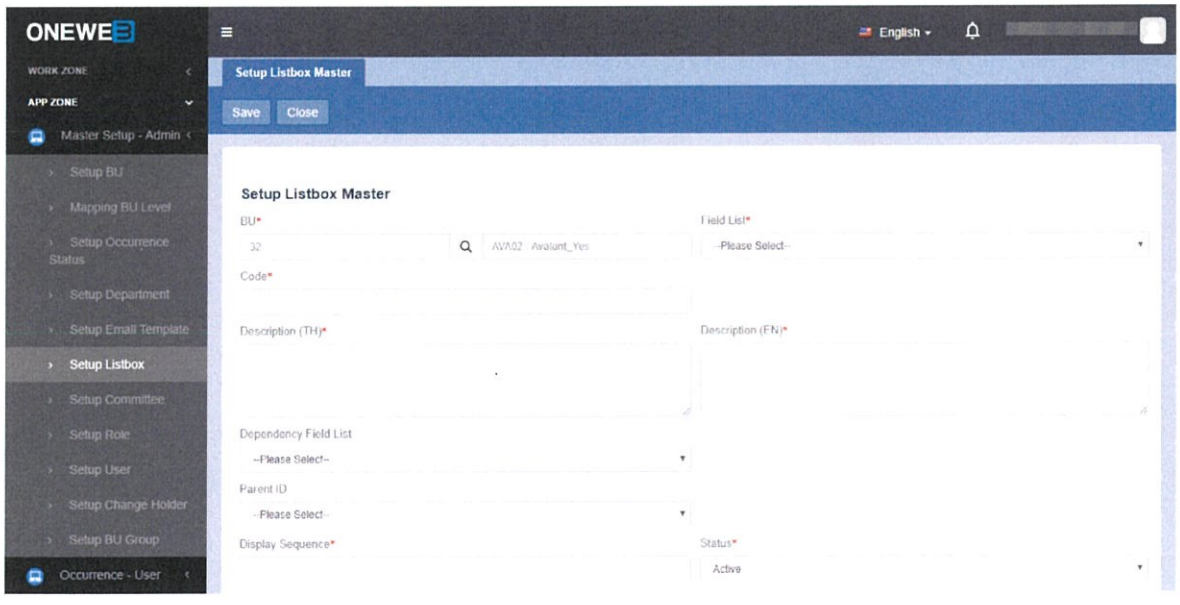
ภาพที่ 4.9 หน้าจอ Insert/Update Setup Email Template

4.1.6 หน้าจอ Setup Listbox

หน้าจอ Setup Listbox จัดการข้อมูลทั่วไปที่จะนำมาแสดงใน Dropdown List



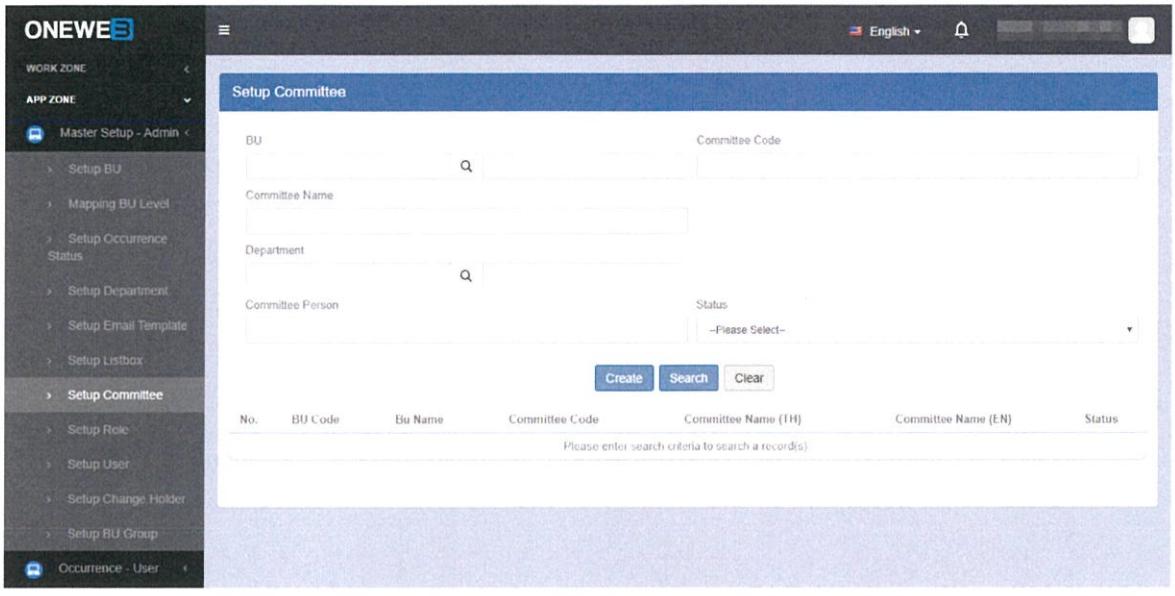
ภาพที่ 4.10 หน้าจอ Search Setup Listbox



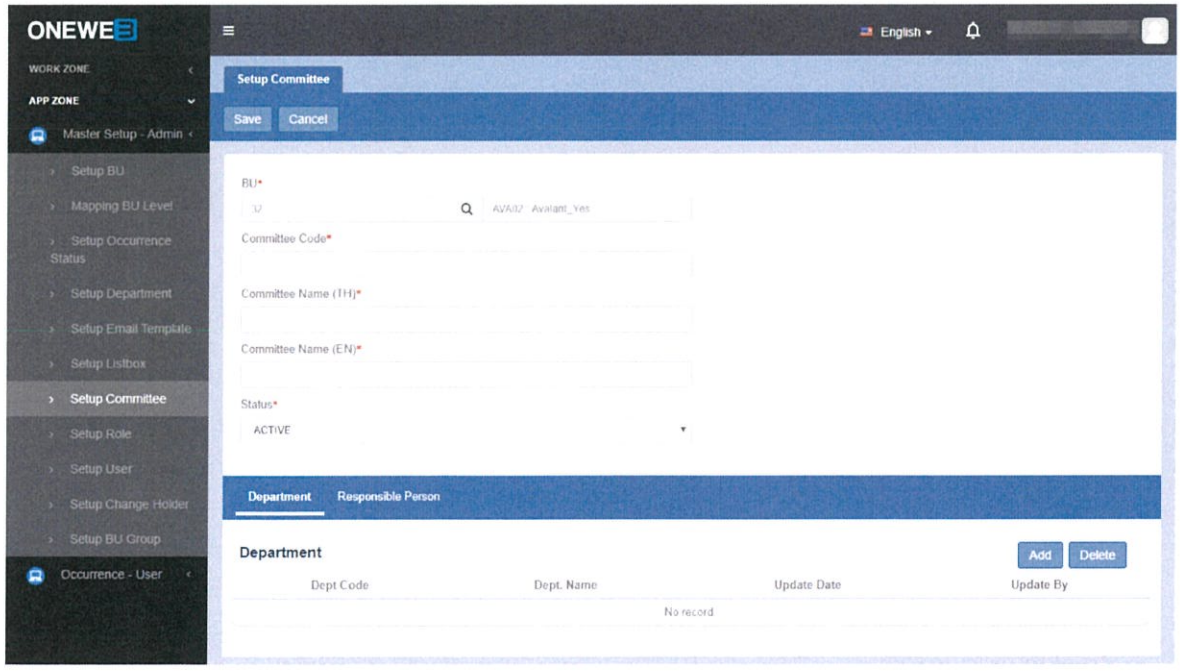
ภาพที่ 4.11 หน้าจอ Insert/Update Setup Listbox

4.1.7 หน้าจอ Setup Committee

หน้าจอ Setup Committee จัดการข้อมูล Committee



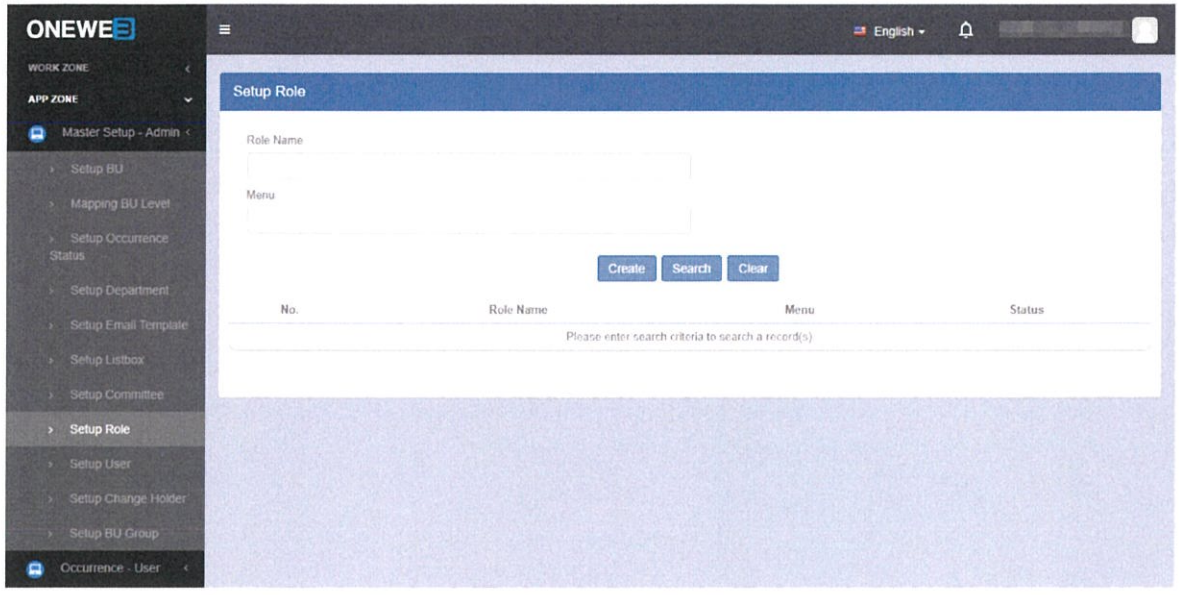
ภาพที่ 4.12 หน้าจอ Search Setup Committee



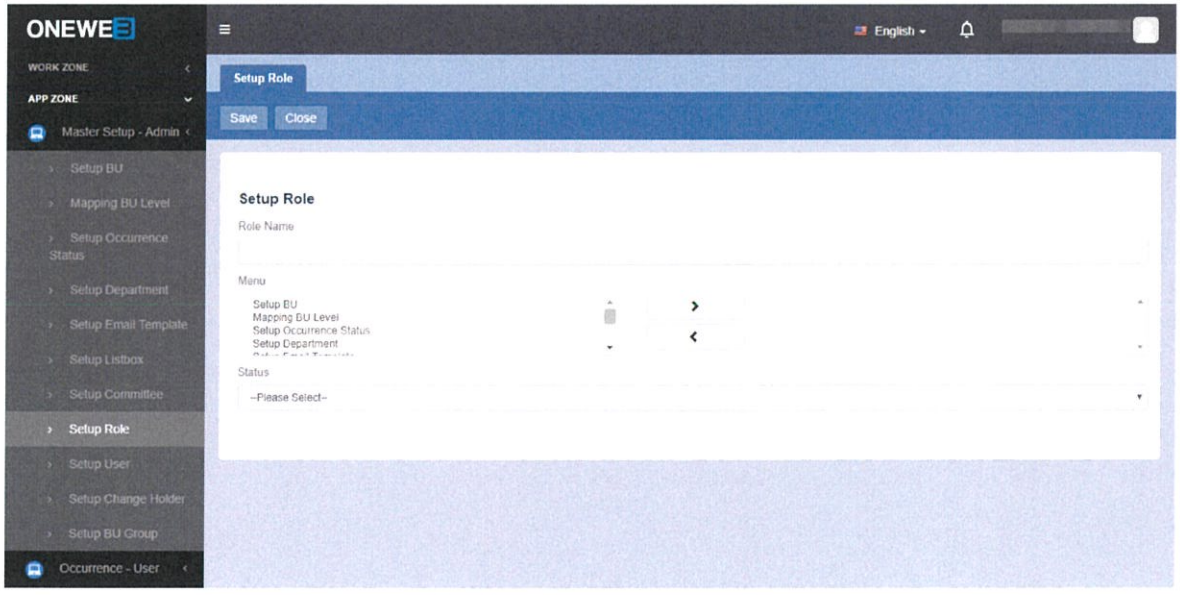
ภาพที่ 4.13 หน้าจอ Insert/Update Setup Committee

4.1.8 หน้าจอ Setup Role

หน้าจอ Setup Role จัดการข้อมูล Role หรือบทบาทของผู้ที่ใช้งาน



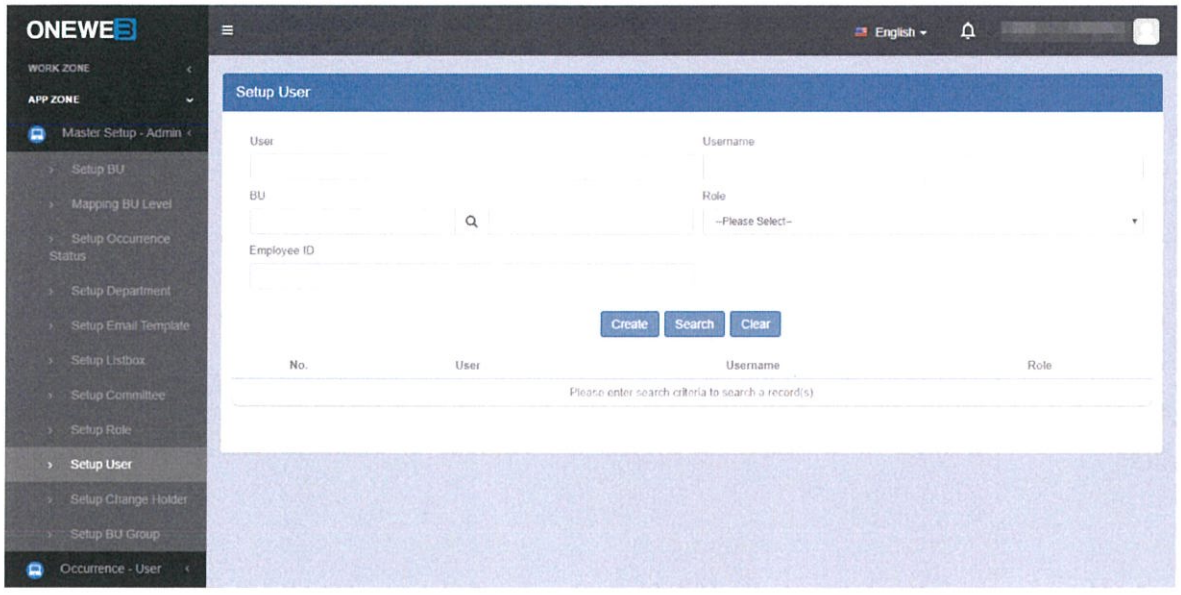
ภาพที่ 4.14 หน้าจอ Search Setup Role



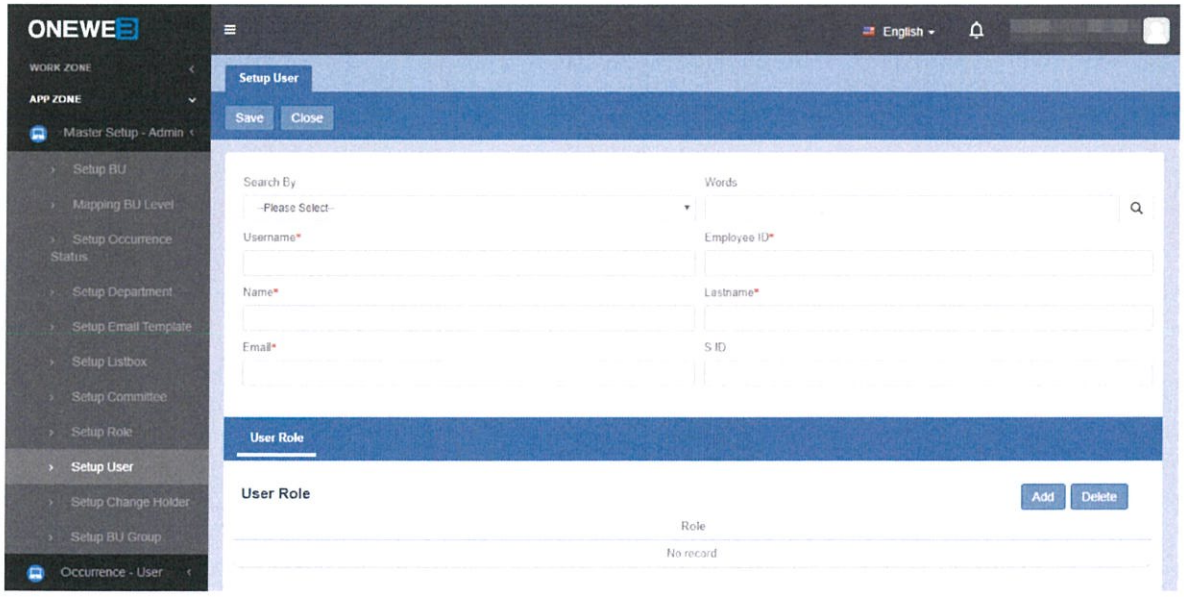
ภาพที่ 4.15 หน้าจอ Insert/Update Setup Role

4.1.9 หน้าจอ Setup User

หน้าจอ Setup User จัดการข้อมูล User หรือรายชื่อผู้ที่ใช้งาน



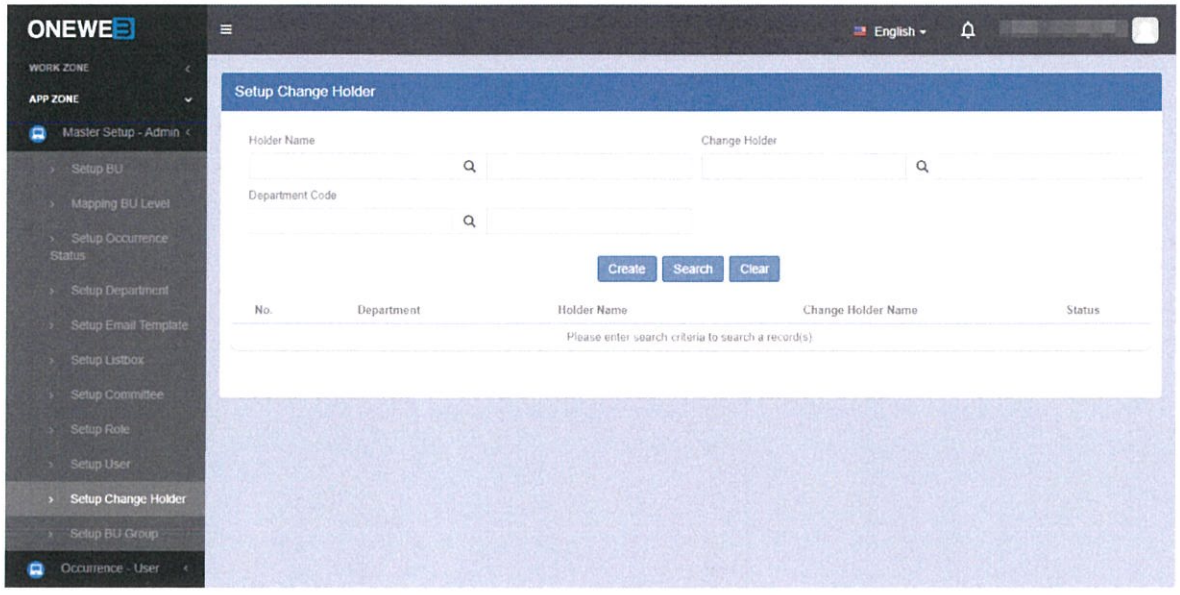
ภาพที่ 4.16 หน้าจอ Search Setup User



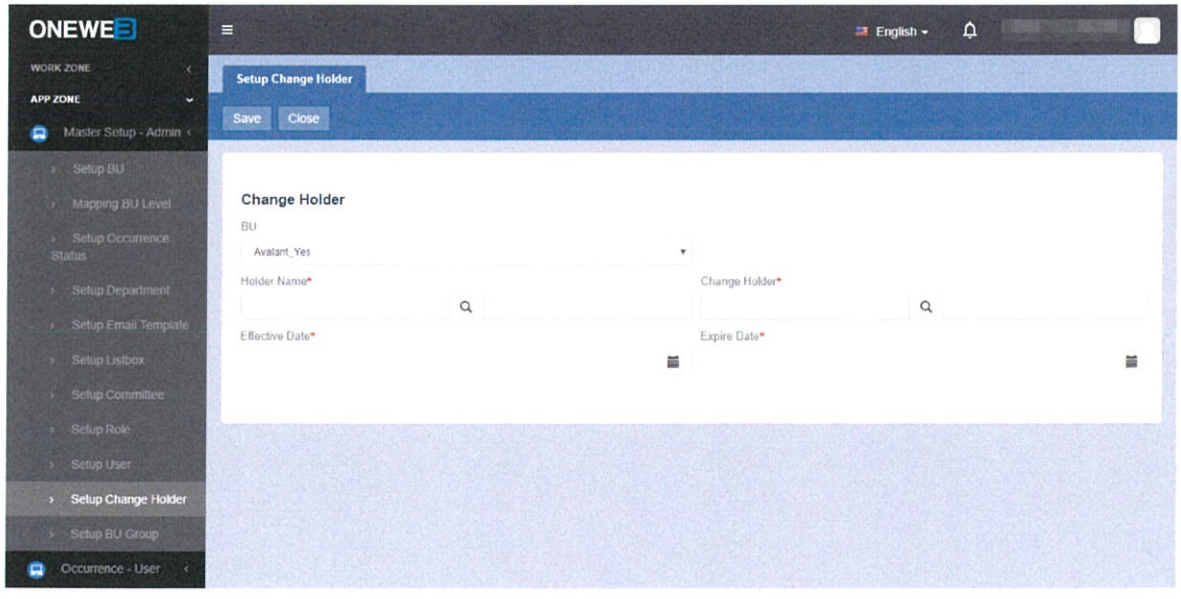
ภาพที่ 4.17 หน้าจอ Insert/Update Setup User

4.1.10 หน้าจอ Setup Change Holder

หน้าจอ Setup Change Holder จัดการข้อมูลการเปลี่ยนผู้รับผิดชอบงาน



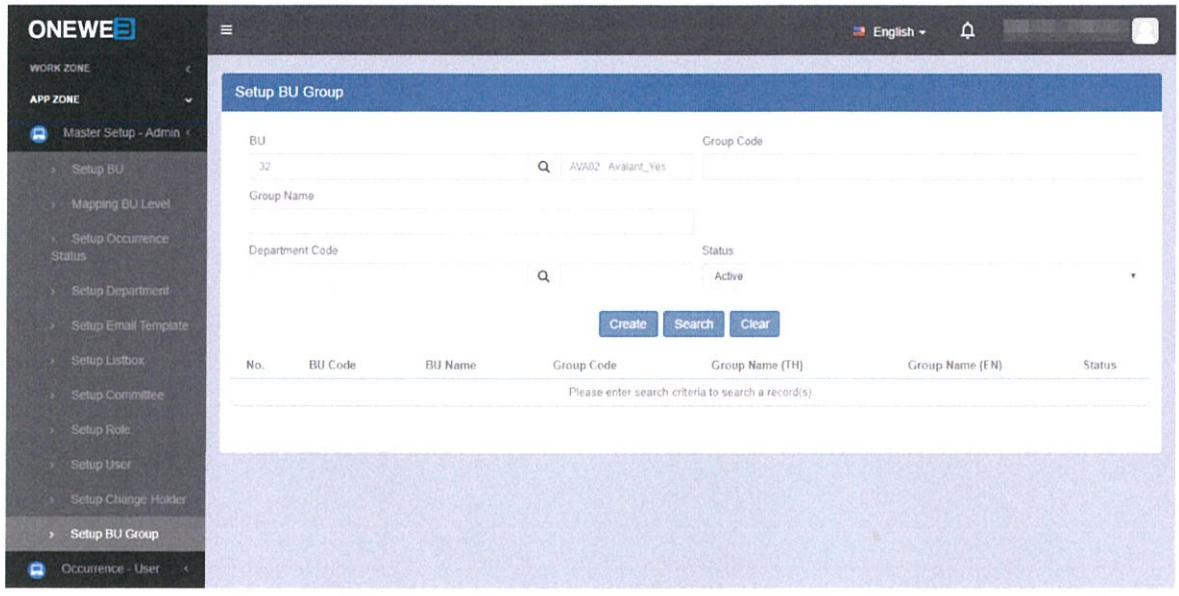
ภาพที่ 4.18 หน้าจอ Search Setup Change Holder



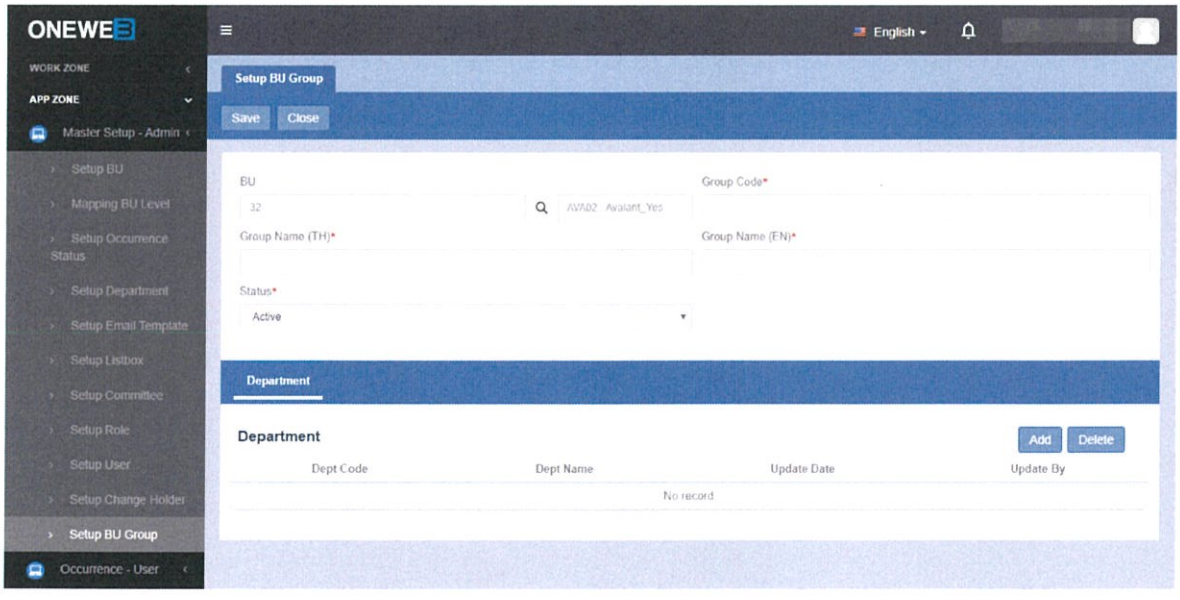
ภาพที่ 4.19 หน้าจอ Insert/Update Setup Change Holder

4.1.11 หน้าจอ Setup BU Group

หน้าจอ Setup BU Group จัดการข้อมูลการจัดกลุ่มโรงพยาบาล



ภาพที่ 4.20 หน้าจอ Search Setup BU Group



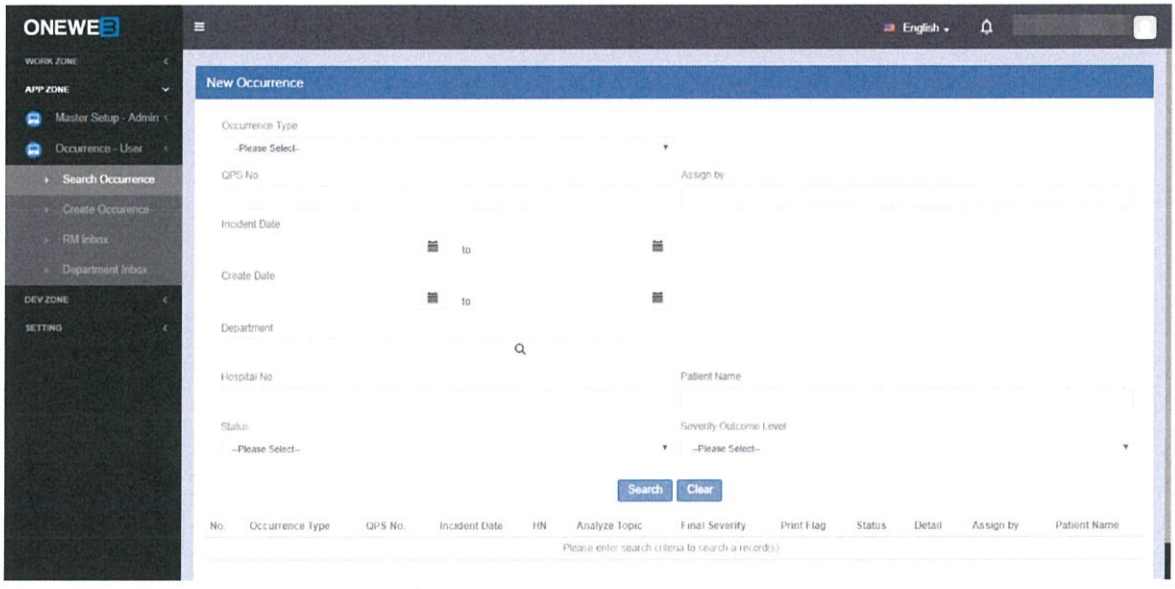
ภาพที่ 4.21 หน้าจอ Insert/Update Setup BU Group

4.2 หน้าจอสำหรับใช้งานระบบ ส่วนผู้ใช้งาน (Occurrence - User)

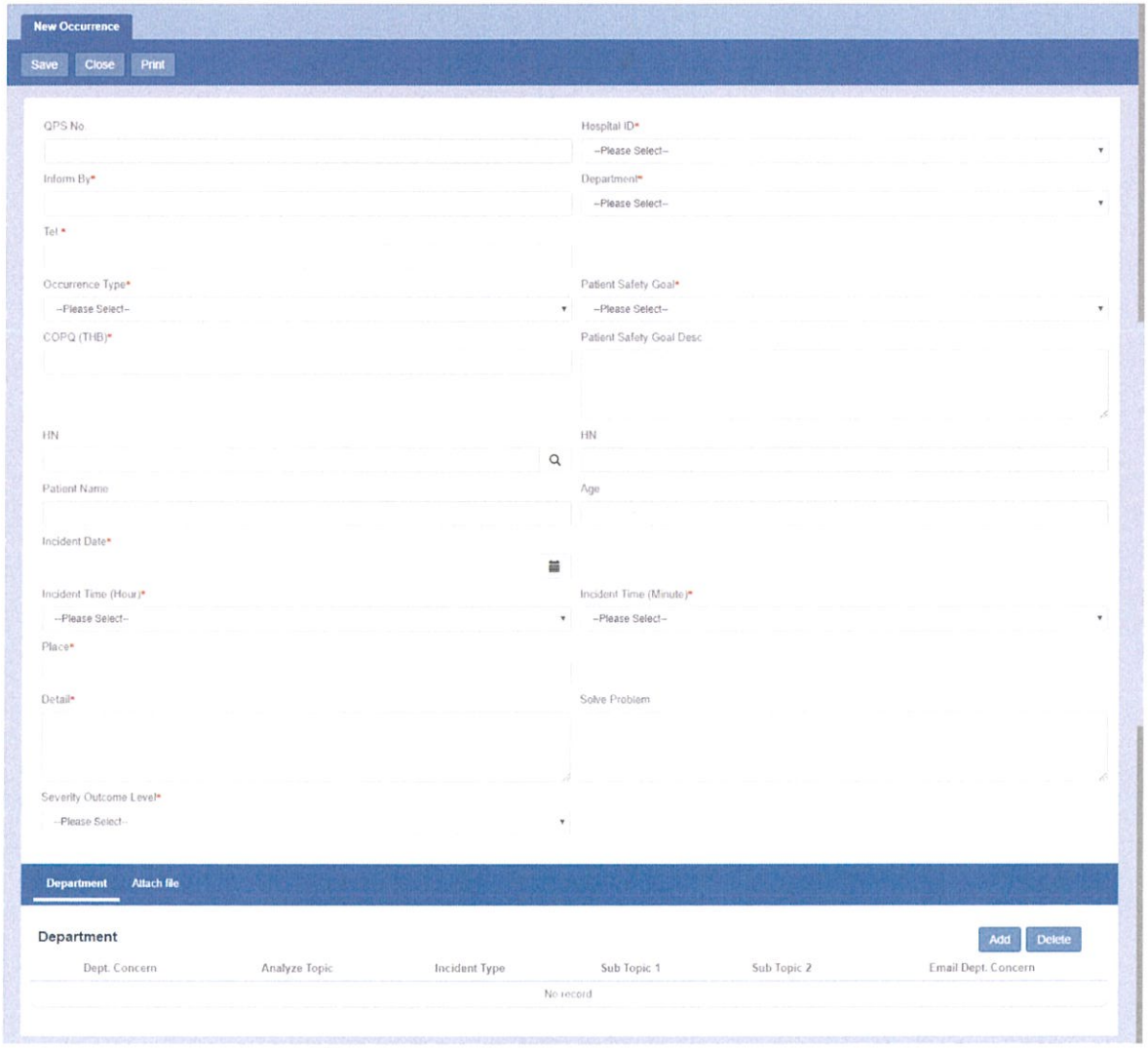
หน้าจอสำหรับใช้งานระบบมีไว้สำหรับให้ผู้ใช้งานทำการเพิ่ม แก้ไข ลบ และค้นหาข้อมูลการแจ้งเหตุการณ์ต่าง ๆ

4.2.1 หน้าจอ New Occurrence

หน้าจอ New Occurrence สำหรับค้นหาและจัดการข้อมูลเหตุการณ์ต่าง ๆ ที่ถูกเพิ่มหรือรายงานเข้ามาในระบบ โดยจะแสดงข้อมูลแค่ส่วนที่ผู้ใช้งานคนนั้นรับผิดชอบ หรือเป็นไปตามที่ลงชื่อเข้าใช้ (Login) มา



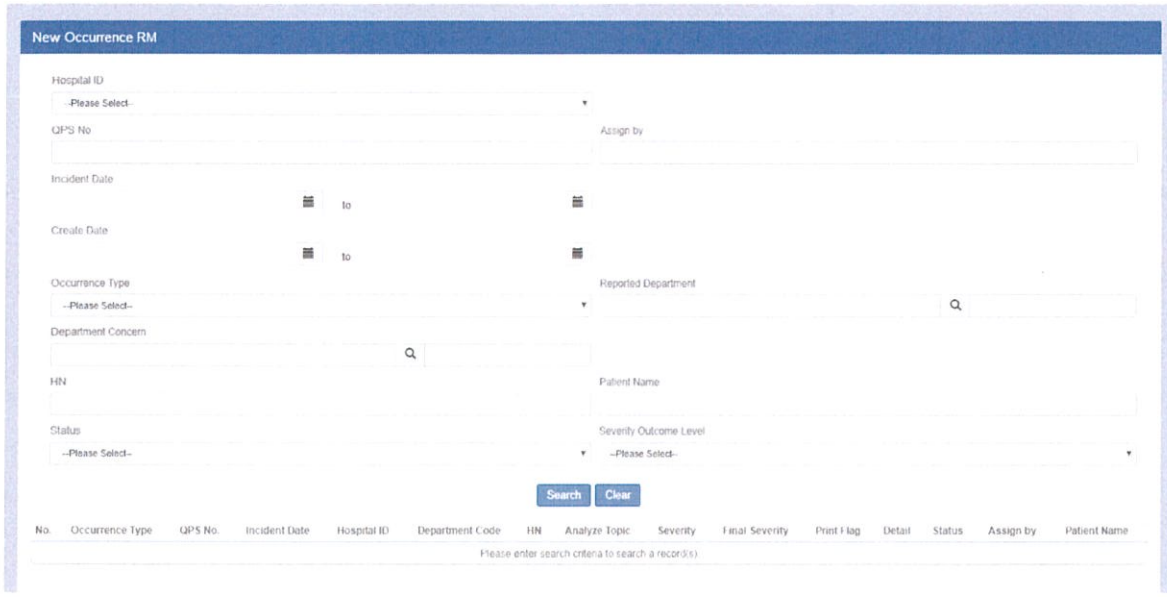
ภาพที่ 4.22 หน้าจอ Search Occurrence



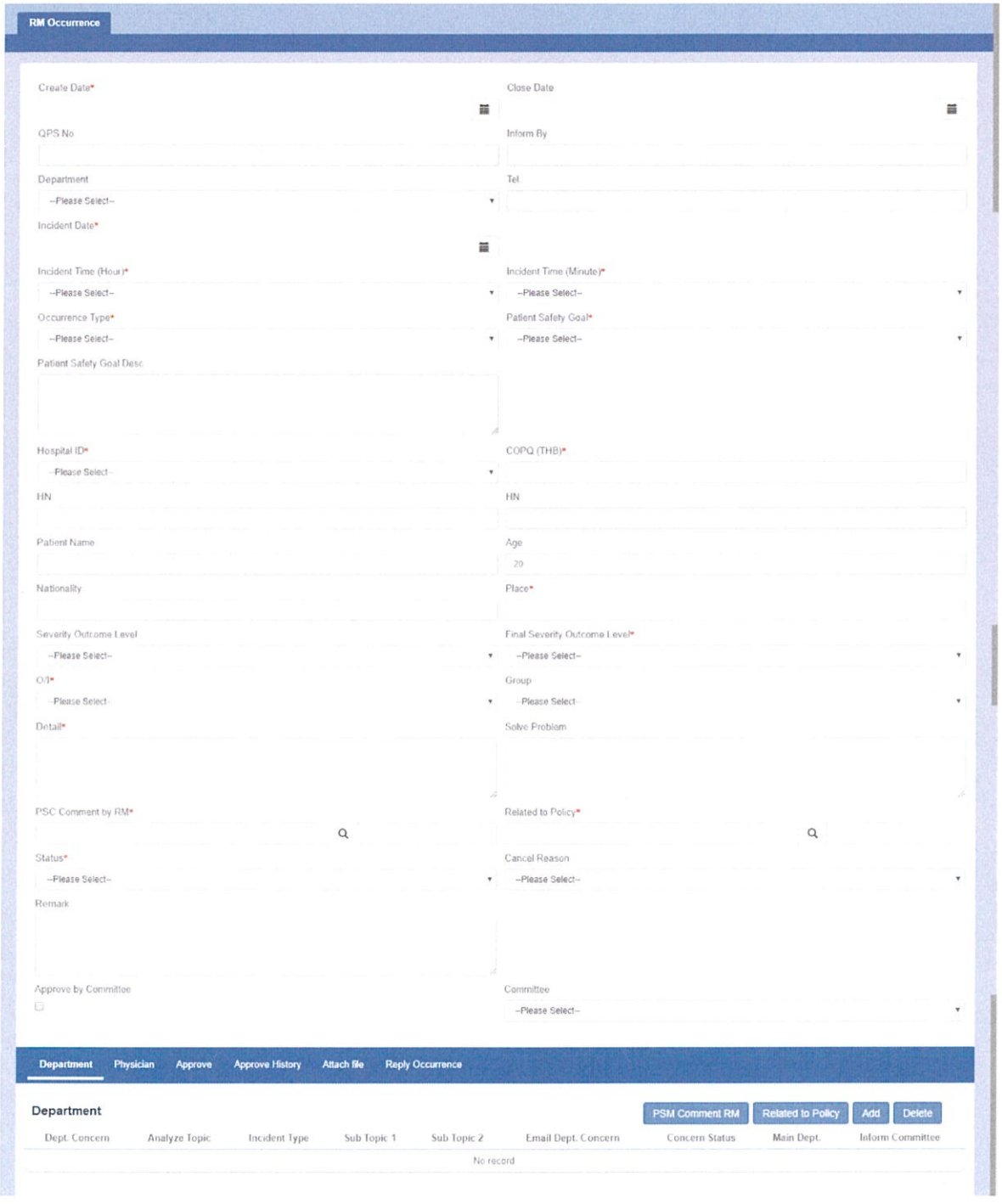
ภาพที่ 4.23 หน้าจอ Insert/Update New Occurrence

4.2.2 หน้าจอ New Occurrence RM

หน้าจอ New Occurrence RM สำหรับค้นหาและจัดการข้อมูลเหตุการณ์ต่าง ๆ ที่ถูกเพิ่มหรือรายงานเข้ามาในระบบ โดยจะแสดงข้อมูลทั้งหมดในระบบ



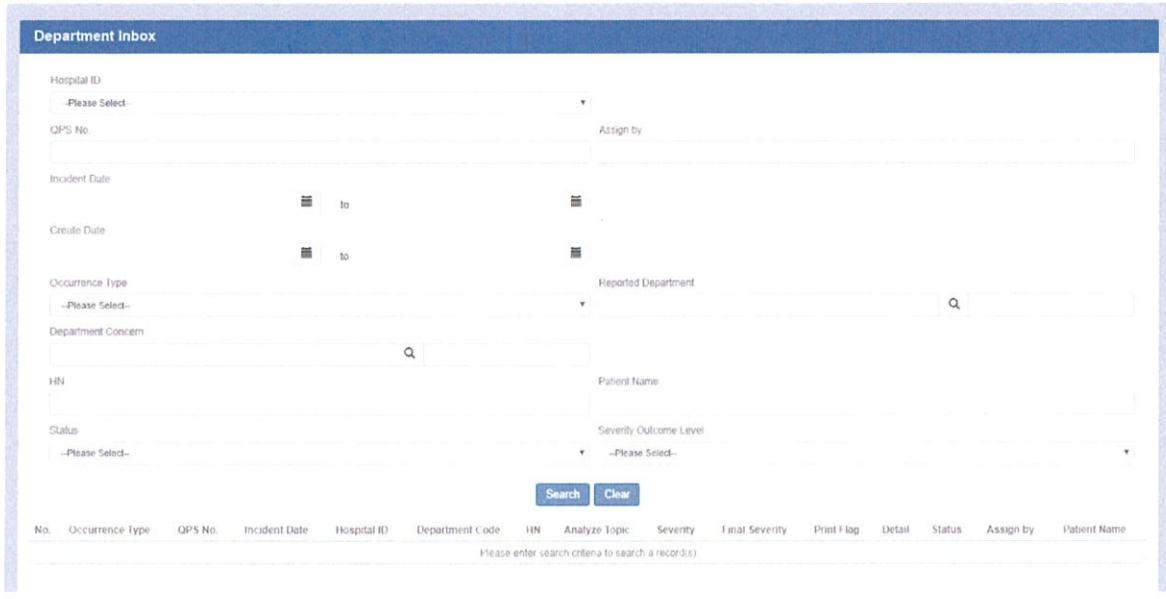
ภาพที่ 4.24 หน้าจอ Search RM Occurrence



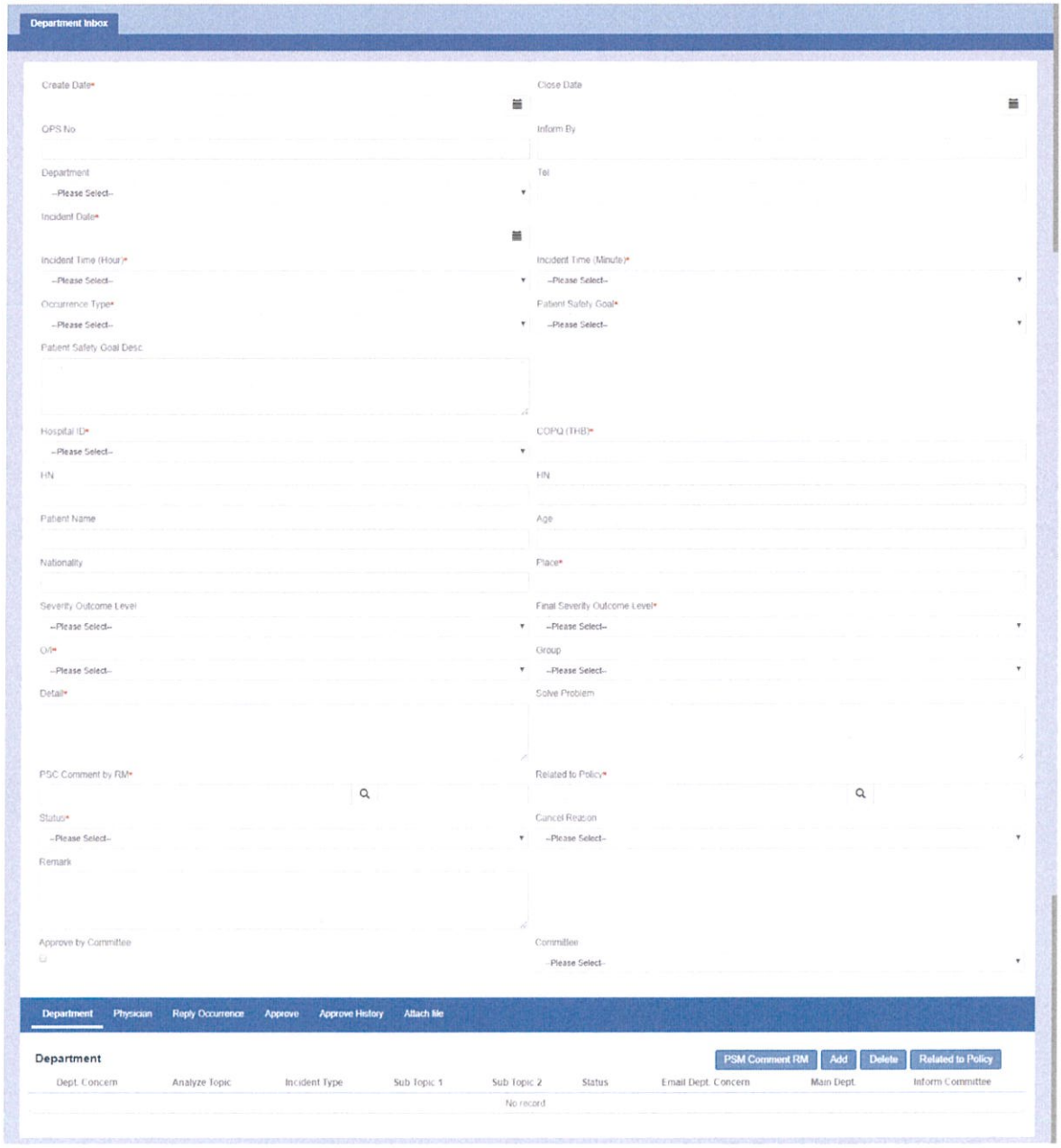
ภาพที่ 4.25 หน้าจอ Update RM Occurrence

4.2.3 หน้าจอ Department Inbox

หน้าจอ Department Inbox สำหรับค้นหาและจัดการข้อมูลเหตุการณ์ต่าง ๆ ที่ถูกเพิ่มหรือรายงานเข้ามาในระบบ โดยจะแสดงข้อมูลตามแผนก (Department) ของผู้ใช้งาน



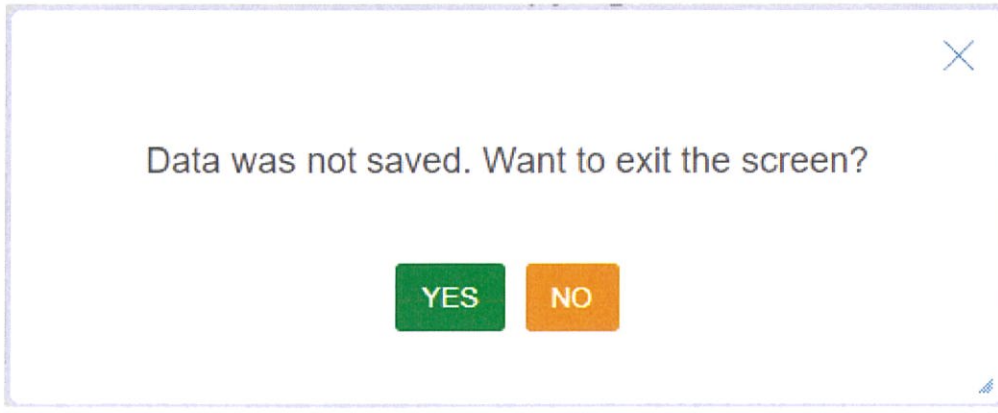
ภาพที่ 2.26 หน้าจอ Search Department Inbox



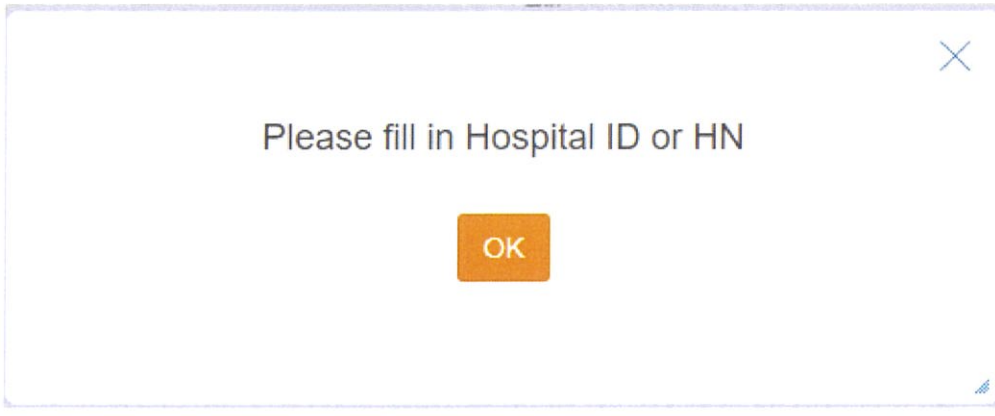
ภาพที่ 4.27 หน้าจอ Update Department Inbox

4.3 ส่วนขยายเพิ่มเติมในหน้าจอ

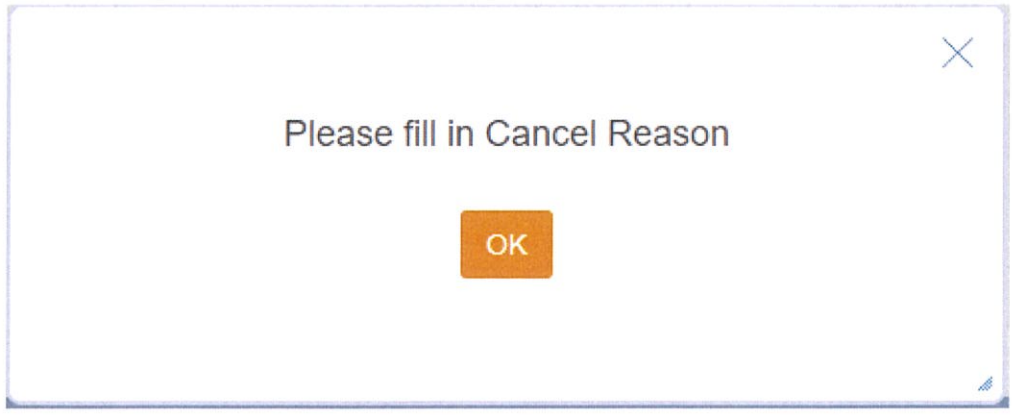
ส่วนขยายเพิ่มเติมในหน้าจอแสดงให้เห็นการโต้ตอบของระบบ หรือเป็นส่วนเพิ่มเติมที่ซอฟต์แวร์ ONEWEB 4.0 ไม่สามารถทำได้ เช่น ป๊อปอัพ (Pop up) แจ้งเตือนต่าง ๆ ไอคอนสำหรับค้นหาข้อมูล และ หน้าจอล็อกอินระบบ



ภาพที่ 4.28 Pop up แจ้งเตือนเมื่อกดปุ่ม Back

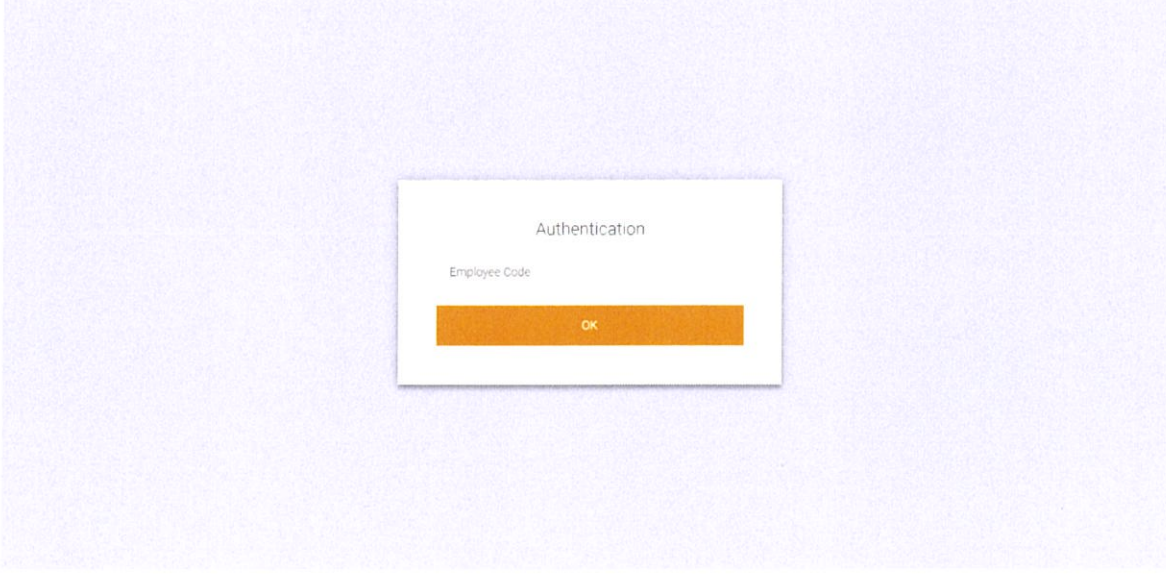


ภาพที่ 4.29 Pop up แจ้งเตือนให้ระบุข้อมูล HN



ภาพที่ 4.30 Pop up แจ้งเตือนให้ระบุข้อมูล Cancel Reason

ภาพที่ 4.31 ไอคอนสำหรับค้นหาข้อมูล Physician



ภาพที่ 4.32 หน้าจอล็อกอินเข้าใช้งานระบบ

บทที่ 5
สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 บทสรุป

การปฏิบัติงานที่บริษัท อัฟวาแลนท์ จำกัด ในระยะเวลา 4 เดือน ตั้งแต่วันที่ 7 สิงหาคม ถึง 30 พฤศจิกายน 2560 ในตำแหน่งงาน Developer ทำให้ได้รับประสบการณ์มากมายจากการได้ทำระบบบริหารจัดการอุบัติเหตุในโรงพยาบาลนี้ ซึ่งจัดทำขึ้นเพื่อรับการแจ้งเหตุการณ์ต่าง ๆ ที่เกิดขึ้นในโรงพยาบาล ไม่ว่าจะเป็นอุบัติเหตุเกี่ยวกับคนไข้หรือบุคลากรในโรงพยาบาล รวมไปถึงอุปกรณ์ทางการแพทย์ หรืออุปกรณ์อำนวยความสะดวกทั้งหลาย ฯลฯ และนำเหตุการณ์นั้น ๆ มาวิเคราะห์โดยผู้เชี่ยวชาญที่รับผิดชอบ หาทางแก้ปัญหาและวิธีป้องกันในอนาคต เพื่อพัฒนางานบริการของโรงพยาบาลให้ดีขึ้น รวมทั้งประสบการณ์ด้านการเขียนโปรแกรม ความรู้เกี่ยวกับซอฟต์แวร์ ONEWEB และความรู้เกี่ยวกับระบบจัดการความเสี่ยงในโรงพยาบาล

ในการปฏิบัติงานสหกิจศึกษาได้ใช้ความรู้ความสามารถในการทำงานเป็นทีม โดยใช้ความรู้ที่ได้ศึกษาบางส่วนมาใช้ปฏิบัติงานจริงและทำหน้าที่ของตัวเองที่ได้รับมอบหมายงานอย่างเต็มที่ การทำงานกับผู้ที่มีประสบการณ์ทำให้เราได้คำแนะนำต่าง ๆ ที่สามารถนำมาใช้ได้จริง งานจึงมีประสิทธิภาพมากขึ้น

5.2 ปัญหาที่พบในระหว่างการดำเนินงาน

- ขาดความรู้ความเข้าใจถึงซอฟต์แวร์ ONEWEB และ Eclipse ทำให้เกิดความล่าช้าของการทำงาน และเกิดข้อผิดพลาดในการทำงาน
- เกิดความขัดข้องของระบบเซิร์ฟเวอร์และซอฟต์แวร์ที่ใช้ในการทำงาน ทำให้ไม่สามารถทำงานได้
- ระยะเวลาที่จำกัดของในการทำงาน

5.3 แนวทางแก้ปัญหา

- ศึกษาเรียนรู้และทำความเข้าใจซอฟต์แวร์จากพนักงานผู้เชี่ยวชาญเพื่อความเข้าใจ
- แจ้งเจ้าหน้าที่ที่รับผิดชอบให้รีบทำการแก้ไขให้ระบบกลับมาใช้งานได้ หรือขอคำแนะนำจากพี่เลี้ยง หรือพนักงานในการแก้ปัญหาดังกล่าว
- วางแผนในการดำเนินงานให้เรียบร้อย และทำงานให้เป็นไปตามแผนที่วางไว้

5.4 แนวทางพัฒนาต่อ และการนำไปใช้

- สามารถนำความรู้ต่าง ๆ เช่น การเขียนภาษาเอสคิวแอล ไปเป็นพื้นฐานในการทำงานในด้านอื่นต่อไปได้

- การนำแนวคิด หลักการ หรือกระบวนการคิดไปช่วยในการดำเนินงานอย่างมีประสิทธิภาพ

- การนำประสบการณ์ที่ได้รับเป็นตัวช่วยในการตัดสินใจเลือกในสิ่งที่สนใจ ช่วยในการทำงานในภายหน้า หรือเป็นการทดสอบเตรียมตัวเพื่อการทำงานจริง

เอกสารอ้างอิง

[1] ชนวัฒน์ ศรีสอ้าน, **การออกแบบและพัฒนาฐานข้อมูล**, มหาวิทยาลัยเทคโนโลยีสุรนารี : นครราชสีมา, 2542.

[2] วราภรณ์ โกวิทวรางกูร, **ระบบฐานข้อมูลและการออกแบบ**, ศูนย์หนังสือจุฬาลงกรณ์มหาวิทยาลัย : กรุงเทพฯ, 2543.

[3] ศิริลักษณ์ โรจนกิจอำนวย, **ระบบฐานข้อมูล**, พิมพ์ครั้งที่ 3, ดวงกลมสมัย : กรุงเทพฯ, 2542.

[4] ศุภกฤษฎิ์ นิวัฒนากูล, **การออกแบบและพัฒนาฐานข้อมูล**, มหาวิทยาลัยเทคโนโลยีสุรนารี : นครราชสีมา, 2545.

[5] **การเขียนโปรแกรมภาษา JavaScript** [ออนไลน์] เข้าถึงได้จาก: https://sites.google.com/site/ntaweera/javascript

[6] **โครงสร้างของภาษา HTML** [ออนไลน์] เข้าถึงได้จาก: https://pirun.ku.ac.th/~agrtnk/web/units/unit1/unit1-3.htm

[7] **ภาษา Java** [ออนไลน์] เข้าถึงได้จาก: http://marcuscode.com/lang/java/introduction

[8] **เรียนรู้และใช้งาน Chrome Developer Tools** [ออนไลน์] เข้าถึงได้จาก: http://pstudiodev.blogspot.com/2012/09/chrome-developer-tools.html

[9] **Avalant OneWeb** [ออนไลน์] เข้าถึงได้จาก: https://www.techtalkthai.com/introduce-avalant-oneweb-configuration-based-enterprise-application-development/

[10] **Avalant เปิดตัว OneWeb 4** [ออนไลน์] เข้าถึงได้จาก: https://www.techtalkthai.com/avalant-launches-oneweb-4-erp-that-supports-ai-and-chatbot-in-thai-language/

[11] **Chrome developer tools** [ออนไลน์] เข้าถึงได้จาก: http://www.siamhttp.com/site/article/working-with-chrome-developer-tools.html

[12] **CSS** [ออนไลน์] เข้าถึงได้จาก: http://www.enjoyday.net/webtutorial/css/css_chapter01.html

[13] **Eclipse** [ออนไลน์] เข้าถึงได้จาก: http://javaitfun.blogspot.com/2013/09/eclipse.html

[14] **Git** [ออนไลน์] เข้าถึงได้จาก: https://devahoy.com/posts/introduction-to-git-and-github/#step1

[15] **JavaScript Overview** [ออนไลน์] เข้าถึงได้จาก: https://developer.mozilla.org/th/docs/Web/JavaScript/Guide/JavaScript_Overview