**บทที่ 5**

**การพัฒนาระบบ**

**1. แผนโครงสร้างของกระบวนการทำงานของระบบ (Flowchart)**

เป็นเครื่องมือแสดงขั้นตอน หรือกระบวนการทำงาน โดยใช้สัญลักษณ์ที่เป็นมาตรฐานเดียวกัน ซึ่งในสัญลักษณ์จะมีข้อความสั้น ๆ อธิบายข้อมูลที่ต้องใช้ ผลลัพธ์ หรือคำสั่งประมวลผลของขั้นตอนนั้น ๆ และเชื่อมโยงขั้นตอนเหล่านั้นด้วยเส้นที่มีลูกศรชี้ทิศทางการทำงานตั้งแต่เริ่มต้นจนจบกระบวนการ ซึ่งในกระบวนของระบบโปรแกรมบัญชีของวิทยาลัยจะมีรูปแบบดังนี้

**บันทึกสมุดบัญชีประจำวัน**

**ค้นหารายงานที่ต้องการ**

**เพิ่มประเภทบัญชี**

**ข้อมูลรายการ  
ทางบัญชี   
เดบิต/เครดิต**

**ค้นหา**

* **- เลขที่บัญชี**
* **- ชื่อบัญชี**
* **- หมวดบัญชี**
* **- วัน/เดือน/ปี**
* **- รายงานที่ต้องการดู**

**- เลขที่บัญชี**

**- ชื่อบัญชี**

**- หมวดบัญชี**

**แสดงายะเอียดบัญชีแยกระเภท**

**แสดงรายงาน  
งบทดลอง**

**แสดงรายงาน  
งบกำไรขาดทุน**

**แสดงรายงาน  
งบกระแสเงินสด**

**ฐานข้อมูล**

ภาพที่ 9 : ภาพแสดงแผนโครงสร้างของระบบบัญชี

**2. แผนภาพกระแสของข้อมูล**

แผนภาพกระแสข้อมูลระดับบนสุด **(**Context Diagram) ที่แสดงภาพรวมการทำงานของระบบที่มีความสัมพันธ์กับสภาพแวดล้อมภายนอกระบบ และแสดงให้เห็นถึงรายละเอียดของ Process การทำงานหลักๆ ที่มีอยู่ภายในภาพรวมของระบบ (Context Diagram) ว่ามีขั้นตอนใดบ้าง

**พนักงานบัญชี**

**ผู้บริหาร**

**หัวหน้าพนักงานบัญชี**

**. 0 .**

**ระบบบัญชีแยกประเภทของวิทยานวัตกรรม**

**ดูข้อมูลรายงานทางการเงิน**

* **งบทดลอง**
* **งบดุล**
* **งบกระแสเงินสด**
* **ทำรายการบันทึกสมุดบัญชีรายวัน**
* **เพิ่มประเภทบัญชี**

**ผู้ดูแลระบบ**

* **ดูข้อมูลรายงานทางการเงิน**
* **ดูข้อมูลรายการบัญชีประจำวัน**
* **ดูข้อมูลรายการบัญชีแยกประเภท**
* **แก้ไขรายการสมุดบัญชีรายวัน**
* **ทำรายการบันทึกสมุดบัญชีรายวัน**
* **เพิ่มประเภทบัญชี**
* **ดูข้อมูลรายงานทางการเงิน**
* **ดูข้อมูลรายการบัญชีประจำวัน**
* **ดูข้อมูลรายการบัญชีแยกประเภท**
* **เพิ่มผู้ใช้งานในระบบ**
* **กำหนดสิทธิ์การใช้งานในระบบ**
* **แก้ไขข้อมูลในระบบ**

ภาพที่ 10 : ภาพแสดงกระแสข้อมูลของระบบบัญชี

**3. การออกแบบฐานข้อมูล (E-R Diagram)**



ภาพที่ 11 : ภาพแสดงแผนผังการออกแบบฐานข้อมูล

รายละเอียดตาราง

ตาราง : journals  
คำอธิบาย : เป็นตารางที่ใช้เก็บข้อมูลรายการของสมุดบัญชีรายวัน

|  |  |  |
| --- | --- | --- |
| ชื่อฟิลด์ | ประเภทข้อมูล | คำอธิบาย |
| journal\_id | Long Integer | หมายเลขรายการ |
| journal\_no | Long Integer | ลำดับในรายการ |
| ref\_no | Text | เลขที่เอกสารอ้างอิง |
| coa\_detail | Text | รายละเอียดของรายการ |
| coa\_id | Text | เลขที่บัญชี(แยกประเภท) |
| drcr | Text | debit หรือ credit |
| amount | Currency | จำนวนเงิน |
| date\_time | Date/Time | วันเดือนปี |
| campus\_id | Long Integer | เลขที่วิทยาเขต(ระบุว่ารายการนี้เป็นของวิทยาเขตไหน) |
| account\_id | Long Integer | เลขที่บัญชี |
| user\_created | Text | คนที่สร้างรายการนี้ |
| date\_created | Date/Time | วันที่สร้างรายการนี้ |
| user\_updated | Text | คนที่แก้ไขรายการนี้ |
| date\_updated | Date/Time | วันที่แก้ไขรายการนี้ |

ตารางที่ 21 : ตารางการออกแบบฐานข้อมูลที่ใช้เก็บข้อมูลรายการของสมุดบัญชีรายวัน

ตาราง : accounts  
คำอธิบาย : เป็นตารางที่ใช้เก็บข้อมูลบัญชี

|  |  |  |
| --- | --- | --- |
| ชื่อฟิลด์ | ประเภทข้อมูล | คำอธิบาย |
| account\_id | Text | เลขที่บัญชี |
| account\_name | Text | ชื่อบัญชี |

ตารางที่ 22 : ตารางการออกแบบฐานข้อมูลที่ใช้เก็บข้อมูลบัญชี

ตาราง : coa  
คำอธิบาย : เป็นตารางที่ใช้เก็บข้อมูลเกี่ยวกับหมวดผังบัญชี

|  |  |  |
| --- | --- | --- |
| ชื่อฟิลด์ | ประเภทข้อมูล | คำอธิบาย |
| coa\_id | Integer | รหัสหมวดผังบัญชี |
| coa\_detail | Text | ชื่อหมวดผังบัญชี |

ตารางที่ 23 : ตารางการออกแบบฐานข้อมูลที่ใช้เก็บข้อมูลเกี่ยวกับหมวดผังบัญชี

ตาราง : campuses  
คำอธิบาย : เป็นตารางที่ใช้เก็บข้อมูลวิทยาเขต

|  |  |  |
| --- | --- | --- |
| ชื่อฟิลด์ | ประเภทข้อมูล | คำอธิบาย |
| campus\_id | Integer | เลขที่วิทยาเขต |
| campus\_name | Text | ชื่อวิทยาเขต |

ตารางที่ 24 : ตารางการออกแบบฐานข้อมูลที่ใช้เก็บข้อมูลวิทยาเขต

ตาราง : users  
คำอธิบาย : เป็นตารางที่ใช้เก็บข้อมูลผู้ใช้งานในระบบ

|  |  |  |
| --- | --- | --- |
| ชื่อฟิลด์ | ประเภทข้อมูล | คำอธิบาย |
| user\_id | Text | ไอดีของผู้ใช้งาน |
| user\_name | Text | ชื่อของผู้ใช้งาน |
| password | Text | รหัสของผู้ใช้งาน |
| salt | Text | ข้อมูลสุ่มที่ใช้ในการเข้ารหัสของผู้ใช้งาน |
| email | Text | อีเมล์ของผู้ใช้งาน |
| tel\_no | Text | เบอร์โทรของผู้ใช้งาน |
| address | Text | ที่อยู่ของผู้ใช้งาน |
| campus\_id | Integer | รหัสวิทยาเขต |
| role\_id | Integer | รหัสตำแหน่งงาน |

ตารางที่ 25 : ตารางการออกแบบฐานข้อมูลที่ใช้เก็บข้อมูลผู้ใช้งานระบบ

ตาราง : roles  
คำอธิบาย : เป็นตารางที่ใช้เก็บข้อมูลเกี่ยวกับตำแหน่งงาน

|  |  |  |
| --- | --- | --- |
| ชื่อฟิลด์ | ประเภทข้อมูล | คำอธิบาย |
| role\_id | Integer | รหัสตำแหน่ง |
| role\_name | Text | ชื่อตำแหน่ง |

ตารางที่ 26 : ตารางการออกแบบฐานข้อมูลที่ใช้เก็บข้อมูลตำแหน่งงาน

**4. การออกแบบหน้าจอการทำงาน**

ในการออกแบบ User interface จะใช้ Baxster Admin Panel เป็นรูปแบบ ( Template) หลักของ Application ซึ่งมีความสามารถในการทำ Responsive และมีรูปแบบที่สวยงาม ส่วนฟอนต์ภาษาไทยจะใช้ Cloud สร้างโดย Typomancer

หน้าจอ Login



ภาพที่ 12 : ภาพแสดงหน้าจอเข้าสู่ระบบของโปรแกรม

หน้าจอเมนู



ภาพที่ 13 : ภาพแสดงหน้าจอเมนูของระบบ

โดยในระบบจะมีการแบ่งเมนูการทำงานดังนี้

*หน้าแรก  
 |\_ สมุดบัญชีรายวันทั่วไป  
 |\_ ค้นหารายการรายวัน  
 |\_ ทำรายการบัญชีรายวัน  
 |\_ แก้ไขรายการบัญชีรายวัน*

*|\_ สมุดบัญชีแยกประเภท  
 |\_ งบทดลอง  
 |\_ งบดุล  
 |\_ งบกำไรขาดทุน  
 |\_ จัดการข้อมูลระบบ  
 |\_ จัดการข้อมูลผู้ใช้งาน  
 |\_ จัดการผังบัญชี  
 |\_ จัดการบัญชี  
 |\_ ตั้งค่ายอดยกมา  
 |\_ ออกจากระบบ*

**5. เทคโนโลยีที่ใช้พัฒนาระบบ**

**Heroku**  
 Heroku เป็นบริการ Cloud Platform ที่มีความสามารถทำหน้าที่เป็น Web Server ได้ ใช้งานง่ายเหมาะสำหรับผู้ที่ไม่ต้องการจัดการ Server ด้วยตนเอง ทำให้มีความสะดวกในการพัฒนาโปรแกรม เนื่องจากสามารถมุ่งเน้นไปที่การพัฒนาโปรแกรมอย่างเดียวได้เลย

**ExpressJS**  
 ExpressJS เป็น Web Application Framework ที่ได้รับความนิยม ทำงานบน NodeJS ซึ่งใช้เป็น Back-End Web Framework ของฝั่งเซิร์ฟเวอร์ สามารถนำมาพัฒนาแอพพลิเคชั่นต่อไปได้ โดย ExpressJS นั้นจะใช้การ Routing (การกำหนดเส้นทางของระบบ) และ Middleware (การรับส่งข้อมูลของระบบ) สามารถเขียนได้ในรูปแบบ MVC ( Model View Controller ) และใช้ในการเชื่อมต่อกับฐานข้อมูล โดยในการพัฒนานั้นจะใช้ภาษา Javascript ในการเขียนโปรแกรม

**NodeJS**  
 NodeJS คือแพลตฟอร์มที่สร้างโดย Javascript ทำให้สามารถใช้ภาษา Javascript ในการทำงานได้นอกเหนือจากการทำงานบนเว็บเบราเซอร์ สาเหตุที่เลือกใช้ NodeJS เนื่องจากต้องการใช้ภาษา Javascript ในการพัฒนาแอพพิลเคชั่น

**AngularJS** AngularJS เป็น Front-End Web Framework ที่พัฒนาโดย Google ซึ่งจะใช้ภาษา Javascript เป็นหลัก ทำให้ Web Developer สามารถเรียนรู้ เข้าถึงและจัดการส่วน Front End ได้ง่ายขึ้นทั้งในเรื่อง ส่วนจัดการรูปแบบของเว็บและการแสดงข้อมูล

**Twitter Bootstrap** Bootstrap เป็น Responsive Web Framwork ที่ช่วยในการให้เว็บไซต์รองรับการแสดงผลเว็บไซต์สำหรับหน้าจอที่มีขนาดแตกต่างกันให้อยู่ในรูปแบบที่เหมาะสม เช่น มือถือ สมาร์ทโฟน แท็บเล็ท

**MySQL** MySQL (มายเอสคิวแอล) เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System) โดยใช้ภาษา SQL เป็นฐานข้อมูลที่มีความนิยมสูง โดยในการพัฒนานั้นจะใช้ MySQL ในการเก็บข้อมูลต่างๆ และนำมาประมวลผลโดยใช้คำสั่ง SQL ในการ Query ข้อมูล เพื่อให้ได้ข้อมูลที่ต้องการ

**6. ความปลอดภัยของระบบ**

* SSL  
  เป็นโปรโตคอลในการสื่อสารหรือส่งข้อมูลอย่างปลอดภัยบนเครือข่ายคอมพิวเตอร์และอินเตอร์เน็ต ที่ใช้กันส่วนใหญ่ได้แก่ การท่องเว็บไซต์ เช็คอีเมล์ ทำธุรกรรมออนไลน์ เป็นต้น ผู้ใช้ทั่วไปเข้าเว็บไซต์ที่มีการเข้ารหัสความปลอดภัยและมีใบรับรอง SSL Certificates จะสังเกตเห็น url ขึ้นต้นด้วย https:// และมีสัญลักษณ์รูปแม่กุญแจอยู่ ซึ่งจำเป็นจะต้องมีค่าใช้จ่าในการขอ Certification Authority (CA) จากผู้ให้บริการ แต่ในขณะนี้มีโครงการ Let’s Encrypt ที่เป็นโครงการที่ทำให้เราสามารถออก Certification Authority (CA) ได้ฟรีโดยไม่มีค่าใช้จ่าย
* IPS  
  Intrusion Prevention System (IPS) คือ ระบบที่คอยตรวจจับการบุกรุกของผู้ที่ไม่ประสงค์ดี โดยจะทำการวิเคราะห์ข้อมูลทั้งหมดที่ผ่านเข้าออกภายในเครือข่ายว่า มีลักษณะการทำงานที่เป็นความเสี่ยงที่ก่อให้เกิดความ เสียหายต่อระบบเครือข่ายหรือไม่ แล้วจึงจะทำการแจ้งเตือนให้ผู้ดูแลระบบทราบ เมื่อใดที่พบข้อมูลที่มีลักษณะการทำงานที่เป็นความเสี่ยงต่อระบบเครือข่ายก็จะทำการป้องกันข้อมูลดังกล่าวนั้น ไม่ให้เข้ามาภายในเครือข่ายได้
* Web Application Firewall

Web Application Firewall (WAF) เป็นอาจจะเป็นได้ทั้งซอฟต์แวร์ที่เอามาลงไว้ในเซิร์ฟเวอร์หรือเป็นพวกอุปกรณ์ ซึ่งจะทำหน้าทีตรวจจับ HTTP Traffic ถูกออกแบบมาเฉพาะเจาะจงในการป้องกัน การโจมตีที่เกิดขึ้นกับ Web Application และยังสามารถตรวจพบเหตุการณ์หรือสิ่งผิดปกติได้ในปริมาณและรายละเอียดที่มากกว่า IPS

* Server Hardening

การทำ Hardening คือการลดความเสี่ยงที่จะเกิดด้านความปลอดภัยให้ได้มากที่สุด เช่น แก้ใขจุดบกพร่องของระบบ แอพพลิเคชั่น หรือ ฐานข้อมูล กระบวนการเสริมความแข็งแรงด้านความปลอดภัยให้กับเครื่องเซิร์ฟเวอร์

* การ Backup ข้อมูล

การ Backup หรือการสำรองข้อมูลนั้นเป็นการคัดลอกข้อมูลเพื่อทำสำเนาเก็บไว้ เพื่อหลีกเลี่ยงความเสียหายที่จะเกิดขึ้น หากข้อมูลเกิดการเสียหายหรือสูญหาย โดยสามารถนำข้อมูลที่สำรองไว้มาใช้งานได้ทันที

**7. การทดสอบระบบ**

* Unit test - เป็นการทดสอบในระดับฟังค์ชั่นการทำงาน เพื่อเป็นการยืนยันการทำงานระดับย่อยที่สุดว่าทำงานได้ถูกต้อง เป็นการทดสอบโดยโปรแกรมเมอร์ผู้เขียนโค๊ด
* Acceptant test - เป็นการทดสอบโดยผู้ใช้ (End user) หรือโดยลูกค้า เพื่อดูว่าซอฟต์แวร์ทำงานถูกต้องตามต้องการหรือไม่

โดยการทดสอบการใช้งานมีวิธีการดังนี้

1. เริ่มทำบัญชีรายวันโดยใส่ข้อมูลดังนี้

ภาพที่ 14 : ภาพแสดงหน้าจอบัญชีรายวัน

1. ข้อมูลบัญชีรายวันที่ได้บันทึกไปจะปรากฎลงในบัญชีแยกประเภท





ภาพที่ 15 : ภาพแสดงหน้าจอบัญชีแยกประเภท

1. งบทดลองของวันที่ 24/04/2017 จะปรากฎข้อมูลที่มาจากรายการ 201704240001





ภาพที่ 16 : ภาพแสดงหน้าจองบทดลอง

1. งบดุลจะพบว่ายอดคงเหลือของ *บัญชีเงินสดและรายการเทียบเท่าเงินสด* (1101000000) ลดลงจากเดิม 10000 เช่นเดียวกับในบัญชีแยกประเภทและงบทดลอง



ภาพที่ 17 : ภาพแสดงหน้าจองบดุล

1. จะพบว่า *บัญชีค่าใช้จ่ายบุคลากร* (5101000000) มีค่าเปลี่ยนแปลงไปเช่นเดียวกับในงบทดลอง แต่เนื่องจากว่า *จ่ายค่าเงินเดือนพนักงาน* (5101010000) เป็นบัญชีย่อยอยู่ภายใต้บัญชีหลัก *บัญชีค่าใช้จ่ายบุคลากร* (5101000000) จึงได้ถูกรวบรวมให้แสดงเฉพาะบัญชีหลัก



ภาพที่ 18 : ภาพแสดงหน้าจองบกำไรขาดทุน



ภาพที่ 19 : ภาพแสดงหน้าจองบกำไรขาดทุน(แบบแสดงฐานะทางการเงิน)

* Usability test - เป็นการทดสอบการใช้งานซอฟต์แวร์โดยผู้เชียวชาญส่วนติดต่อผู้ใช้ ว่าใช้งานได้ง่าย เข้าใจได้ง่าย หรือไม่

**8. ความต้องการของระบบ**

**Server**

* Network Bandwidth : 2TB/month
* Harddisk : 200 MB
* Memory : 512 MB
* CPU : 1 Core

**Client**

* Web browser ( Google Chrome , Firefox , Safari , Internet Explorer 11 )
* Internet

**9. คุณสมบัติที่แตกต่างจากโปรแกรมบัญชีทั่วไป**

* เป็นระบบบัญชีออนไลน์สามารถเข้าใช้งานได้พร้อมกันทั้ง 3 วิทยาเขต
* ออกแบบโปรแกรมให้ใช้งานง่ายสำหรับนักบัญชี
* มีการจัดการสิทธิ์ในการใช้งานที่เหมาะสมกับระบบของวิทยาลัยนวัตกรรมการจัดการ
* รูปแบบของผังบัญชีออกแบบให้คล้ายกับผังบัญชีของหน่วยงานกองคลังของมหาลัยเพื่อให้ง่ายต่อการเข้าใจ
* แบ่งบัญชีออกตามสาขาวิชาที่เปิดสอน และออกรายงานต่างๆแยกตามบัญชีได้

**10. สิทธิ์ในการเข้าใช้งานระบบ**

โปรแกรมระบบบัญชีแบ่งออกเป็น 6 ฟังค์ชั่นการทำงานหลัก ได้แก่ บัญชีรายวัน , บัญชีแยกประเภท , งบทดลอง , งบดุล , งบกำไรขาดทุน และจัดการข้อมูลระบบ โดยจะมีการกำหนดสิทธิ์ตามหน้าที่ของผู้ใช้งาน ซึ่งในการใช้งานระบบได้นั้นจำเป็นต้องมี username และ password เพื่อยืนยันตัวต้นในการเข้าใช้งานระบบก่อน สิทธิ์ในการทำงานต่างๆจะถูกแบ่งตามตำแหน่งของผู้ใช้งานต่างๆ ดังนี้

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ฟังค์ชั่นการทำงาน \ ตำแหน่ง | พนักงานบัญชี | หัวหน้าพนักงาน | ผู้บริหาร | ผู้ดูแลระบบ |
| 1.สมุดบัญชีรายวัน | **✓** | **✓** | **✓** | **✓** |
| 1.1 ค้นหารายการประจำวัน | **✓** | **✓** | **✓** | **✓** |
| 1.2 ทำรายการประจำวัน | **✓** |  |  | **✓** |
| 1.3 แก้ไขรายการประจำวัน |  | **✓** |  | **✓** |
| 2. สมุดบัญชีแยกประเภท | **✓** | **✓** | **✓** | **✓** |
| 3. งบทดลอง | **✓** | **✓** | **✓** | **✓** |
| 4. งบดุล | **✓** | **✓** | **✓** | **✓** |
| 5. งบกำไรขาดทุน | **✓** | **✓** | **✓** | **✓** |
| 6. จัดการข้อมูลระบบ |  | **✓** |  | **✓** |
| 6.1 จัดการผู้ใช้งาน |  |  |  | **✓** |
| 6.2 จัดการผังบัญชี |  | **✓** |  | **✓** |
| 6.3 จัดการสมุดบัญชี |  | **✓** |  | **✓** |
| 6.4 ตั้งค่ายอดยกมา |  | **✓** |  | **✓** |

ตารางที่ 27 : ตารางแสดงสิทธิ์ในการใช้งานระบบของผู้ใช้งาน