**ขั้นตอนการพัฒนาโปรแกรม**

1. ศึกษาโครงสร้างของระบบฐานข้อมูลของระบบบริหารคงคลัง StockDB เพื่อให้ทราบและจัดเตรียมการนำข้อมูลในระบบฐานข้อมูลดังกล่าวมาใช้ในการออกแบบ และพัฒนาโปรแกรม
2. สร้างระบบการขอเบิกเวชภัณฑ์ และระบบการอนุมัติการขอเบิก ของ Server คลังเวชภัณฑ์ส่วนกลาง เพื่อทำการรับรายการขอเบิกจากหน่วยงานอื่น และทำการอนุมัติรายการขอเบิกดังกล่าว
3. สร้างระบบบริหารคงคลังของหน่วยเบิก โดยมีระบบ คงคลังเวชภัณฑ์ ระบบรับ – จ่ายเวชภัณฑ์ และ ระบบขอเบิกเวชภัณฑ์จากส่วนกลาง ผ่านหน้าโปรแกรมที่เป็น web base เพื่อให้ทำงานระยะไกลได้
4. สร้างและออกแบบระบบรายงานต่างๆ ที่เกี่ยวข้อง และระบบติดตามการเคลื่อนไหวของเวชภัณฑ์
5. พัฒนาระบบการสื่อสารและแจ้งเตือนระหว่างผู้ใช้งานจากหน่วยงานอื่น กับ ผู้ใช้งานในส่วนกลาง เพื่อให้ทราบกรณีที่มีการทำรายการขอเบิกเข้ามาในส่วนกลาง

A screenshot of a computer

Description automatically generated

**รูปที่ 1** แผนการดำเนินการ

**โครงสร้างระบบฐานข้อมูล StockDB**

StockDB เป็นโปรแกรมบริหารจัดการคลังเวชภัณฑ์ ที่ทาง รพ.ใช้ในการบริการจัดการระบบจัดซื้อ เบิก จ่ายและบริหารคลังเวชภัณฑ์หลักในปัจจุบัน เนื่องจากเป็นโปรแกรมที่อยู่ภายในเระบบครือข่ายท้องถิ่นของ รพ. (Local area network) ทำให้มีความสามารถจำกัดในเรื่องของการให้บริการขอเบิกได้เฉพาะในหน่วยงานภายใน รพ. เท่านั้น การทำเรื่องขอเบิกให้หน่วยงานภายนอกนั้น ทาง รพ.ต้องเป็นผู้ดำเนินการแทนเอง ไม่สามารถให้ผู้ใช้จากหน่วยเบิกภายนอกเข้ามาดำเนินการเองได้ ดังนั้นทาง รพ.จึงต้องสร้างโปรแกรมสำหรับงานดังกล่าว โดยในขั้นตอนแรกนั้น จำเป็นต้องมีการศึกษาตารางข้อมูลของระบบฐานข้อมูลโปรแกรม StockDB โดยมีตารางฐานข้อมูลที่สำคัญตามรูปที่ 2

A screenshot of a computer program

Description automatically generated

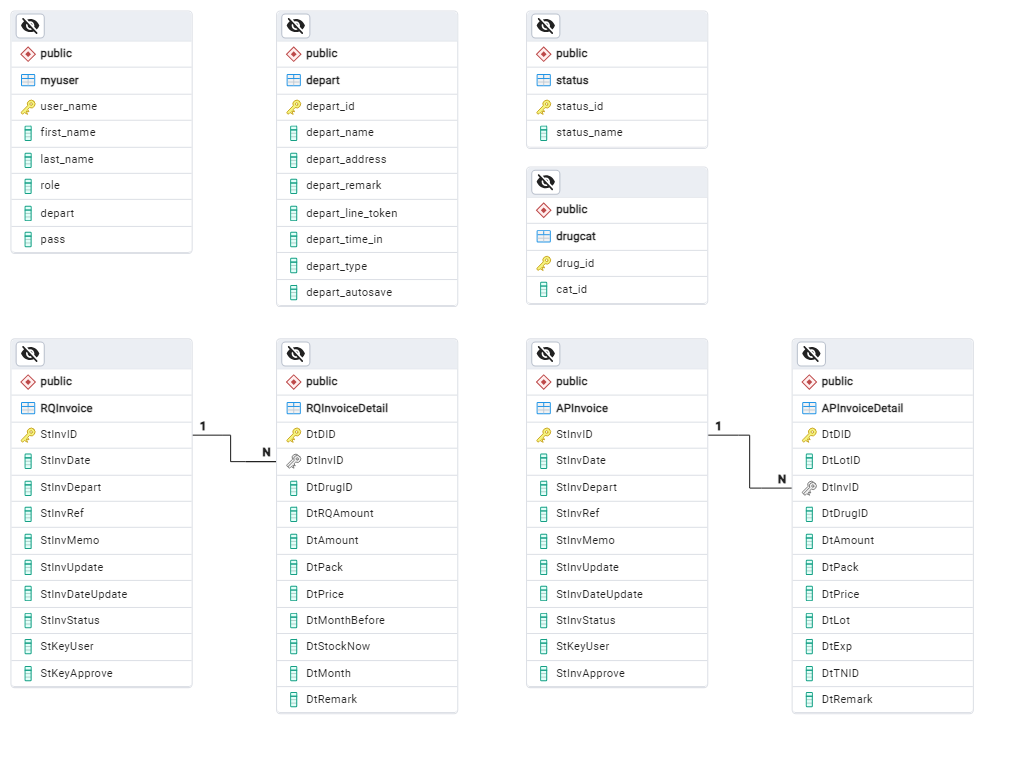
**รูปที่ 2** ตารางฐานข้อมูล StockDB

ตัวอย่างตารางที่เกี่ยวข้องมีดังต่อไปนี้

|  |  |  |
| --- | --- | --- |
| หมวดหมู่ | ตาราง | ข้อมูลที่เก็บ |
| 1. ข้อมูลเวชภัณฑ์ | DrugItem | ข้อมูลพื้นฐานรายการเวชภัณฑ์ รหัส ชื่อ ขนาดบรรจุ หน่วยจ่าย หมวด กลุ่มเวชภัณฑ์ |
| DrugDetail | ข้อมูล Lot เวชภัณฑ์ ปริมาณคงคลัง วันหมดอายุของ Lot และต้นทุน |
| 2. ข้อมูลการขอเบิกเวชภัณฑ์ | RQInvoice | ข้อมูลรายการขอเบิก เลขที่ขอเบิก วันเวลาที่ขอเบิก หน่วยงานที่เบิก |
| RQInvoiceDetail | ข้อมูลรายการเวชภัณฑ์ที่ขอเบิก รหัสเวชภัณฑ์ จำนวนที่เบิก จำนวนที่อนุมัติ |
| 3. ข้อมูลการเบิกเวชภัณฑ์ | OTInvoice | ข้อมูลรายการเบิก เลขที่เบิก หน่วยงานที่เบิก วันเวลาที่ตัดจ่าย |
| OTInvoiceDetail | ข้อมูลรายการเวชภัณฑ์ที่เบิก รหัสเวชภัณฑ์ Lot วันหมดอายุ ต้นทุน |
| 4. ตารางรหัสที่เชื่อมโยง | Supplier | ข้อมูลผู้จำหน่าย |
| Customer | ข้อมูลหน่วยเบิก |
| UnitType | ข้อมูลหน่วยเวชภัณฑ์ ชนาดบรรจุ |
| Category | ข้อมูลหมวดเวชภัณฑ์ |
| DrugGroupItem | ข้อมูลกลุ่มเวชภัณฑ์ |
| TradeNmae | ข้อมูลชื่อการค้า |

**ระบบการขอเบิกเวชภัณฑ์ และระบบการอนุมัติการขอเบิก ของ Server คลังเวชภัณฑ์ส่วนกลาง**

หลังจากที่ได้ข้อมูลระบบฐานข้อมูล StockDB แล้ว ขั้นตอนต่อไปจะเป็นการสร้างระบบ Web API สำหรับใช้ในการจัดการข้อมูลการขอเบิกที่จะส่งมาจากหน่วยงานต่างๆ โดยระบบนี้จะพัฒนาด้วยภาษา JavaScript ที่ทำงานภายใต้ NodeJS runtime environment โดยเป็นทั้ง Web API และ Web Server ในตัว มีตัวฐานข้อมูลเป็น PostgreSQL และเชื่อมโยงกับฐานข้อมูลของระบบบริหารเวชภัณฑ์ StockDB เดิมผ่าน Library node-adodb โดยติดตั้งไว้ที่ Server ของฝ่ายเภสัชกรรมของ รพ. สำหรับโครงสร้างฐานข้อมูล Server คลังเวชภัณฑ์ส่วนกลางแสดงตามรูปที่ 3



**รูปที่ 3** ตารางฐานข้อมูลระบบขอเบิก และอนุมัติเบิกของส่วนกลาง

ข้อมูลของตารางต่างๆ

|  |  |  |
| --- | --- | --- |
| หมวดหมู่ | ตาราง | ข้อมูลที่เก็บ |
| 1. ข้อมูลผู้ใช้งาน | myuser | ข้อมูลผู้ใช้งานทั้งหมด รหัสผู้ใช้ รวมถึง Hash ของรหัสผ่าน |
| 2. ข้อมูลหน่วยงาน | depart | ข้อมูลหน่วยงานทั้งหมด หน่วยส่วนกลาง และหน่วยเบิก |
| 3. ข้อมูลรายการขอเบิก | RQInvoice | ข้อมูลเลขที่รายการขอเบิก รหัสหน่วยงาน วันเวลาที่ขอเบิก |
| RQInvoiceDetail | ข้อมูลรายการขอเบิก รหัสเวชภัณฑ์ จำนวนขอเบิก อัตราการเบิกจ่ายย้อนหลัง |
| 4. ข้อมูลรายการนำส่ง | APInvoice | ข้อมูลเลขที่นำส่ง รหัสหน่วยงาน วันเวลาอนุมัติ และนำส่ง |
| APInvoiceDetail | ข้อมูลรายการนำส่ง รหัสเวชภัณฑ์ จำนวนที่ให้เบิก ต้นทุน Lot และวันหมดอายุ |
| 5. ตารางสถานะ | status | ตารางเก็บรหัสและชื่อสถานะ สำหรับใช้ในข้อ 3 และ 4 |

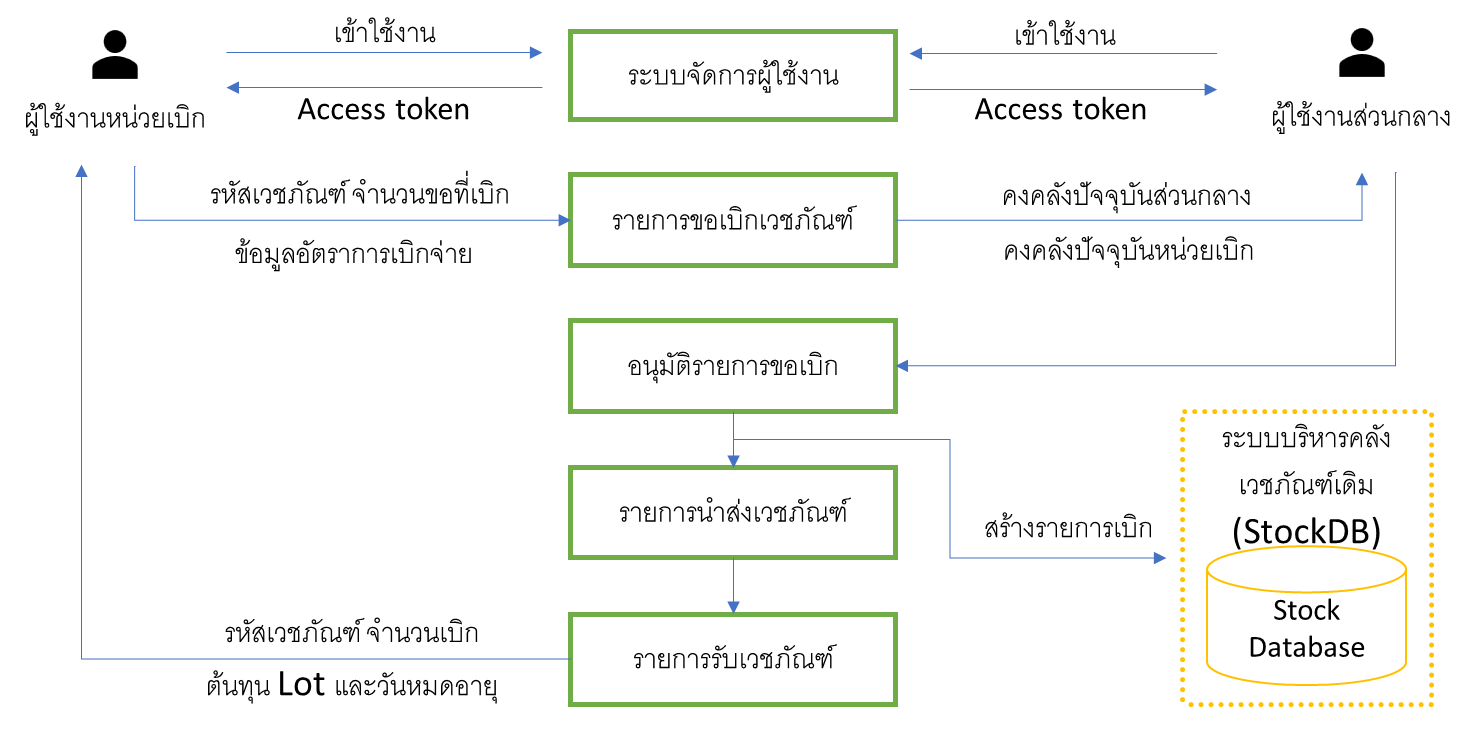
โดยระบบการขอเบิกเวชภัณฑ์ และระบบการอนุมัติการขอเบิก คลังเวชภัณฑ์ส่วนกลาง จะมีระบบต่างๆดังต่อไปนี้

1. ระบบการสร้างและกำหนดผู้ใช้งาน รวมถึงการควบคุมสิทธิการเข้าถึงของผู้ใช้งานจากหน่วยงานต่าง โดยระบบจะแยกผู้ใช้งานตามระดับการเข้าถึง ได้แก่ ผู้ดูแลระบบ, ผู้ใช้งานส่วนกลาง, ผู้ใช้งานจากหน่วยเบิก และผู้สังเกตการณ์ ซึ่งจะมีสิทธิในการใช้งานที่แตกต่างกัน

* ผู้ดูแลระบบ สามารถเข้าถึงได้ทั้งหมด
* ผู้ใช้งานส่วนกลาง สามารถเข้าถึงระบบการขอเบิก และอนุมัติการขอเบิกจากหน่วยเบิกได้
* ผู้ใช้งานจากหน่วยเบิก สามารถทำรายการขอเบิกของหน่วยตัวเอง รวมถึงการรับและตัดจ่ายรายการเวชภัณฑ์ของหน่วยเบิกตนเอง
* ผู้สังเกตการณ์ สามารถเข้าดูระบบต่างๆได้ แต่ไม่สามารถทำรายการใดๆได้

สำหรับการเข้าใช้งานของผู้ใช้งานนั้น จะกระทำผ่านการ login ด้วย username และ password ในครั้งแรก หลังจากนั้นจะเชื่อมต่อกับ API ผ่านทาง Access token แบบ Customized JWT token และจะ logout ออกจากระบบอัตโนมัติ เมื่อไม่ได้ใช้งานเกิน 15 นาที

1. ระบบการรับรายการขอเบิกจากหน่วยเบิก เมื่อผู้ใช้งานหน่วยเบิกทำรายการขอเบิกจากหน่วยเบิกเข้ามาในส่วนกลาง ทางส่วนกลางจะทำการตรวจสอบความจำเป็น จำนวนที่ขอเบิก โดยระบบจะแสดงอัตราการเบิกจ่ายย้อนหลัง จำนวนคงคลังปัจจุบันของหน่วยเบิก และคงคลังปัจจุบันของเวชภัณฑ์ส่วนกลาง เพื่อทำการเปรียบเทียบและช่วยในการตัดสินใจอนุมัติจำนวนที่จะให้เบิก ทั้งนี้ผู้ใช้ส่วนกลางสามารถเข้าไปดูรายละเอียดและการเคลื่อนไหวของเวชภัณฑ์ทั้งหมดของหน่วยเบิกที่ทำรายการขอเบิกมาได้
2. ระบบอนุมัติรายการขอเบิก เมื่อผู้ใช้งานส่วนกลางทำการตรวจสอบรายการขอเบิกจากหน่วยเบิกแล้ว ผู้ใช้งานส่วนกลางสามารถทำการอนุมัติรายการขอเบิกจากหน่วยเบิกนี้ได้ ซึ่งภายหลังจากทำการอนุมัติแล้ว ระบบจะต้องทำการสร้างรายการเบิกไปยังระบบบริหารคลังเวชภัณฑ์ StockDB เดิมของ รพ. พร้อมกับสร้างรายการนำส่งเวชภัณฑ์ เพื่อให้ทางหน่วยเบิกทำการโอนเข้ารายการรับเวชภัณฑ์ของหน่วยเบิกต่อไป

****

**รูปที่ 4** แผนภูมิการเคลื่อนไหวของข้อมูล สำหรับหน่วยงานส่วนกลาง

**ระบบคงคลังเวชภัณฑ์ของหน่วยเบิก**

หน่วยเบิกแต่ละหน่วยจะต้องมีระบบบริหารคลังเวชภัณฑ์ของตนเอง โดยระบบนี้สามารถเพิ่มหน่วยเบิกเข้าไปได้ตามต้องการ และระบบจะทำการสร้างฐานข้อมูลสำหรับจัดการคงคลังของแต่ละหน่วยเบิกให้ รูปแบบฐานข้อมูลของหน่วยเบิกเป็นดังต่อไปนี้

A screenshot of a computer

Description automatically generated

**รูปที่ 5** ตารางฐานข้อมูลหน่วยเบิก

รายละเอียดของแต่ละตาราง จะยึดแบบข้อมูลตามตารางเดิมที่มีอยู่ในตัวระบบบริหารคลังเวชภัณฑ์ StockDB เดิมของ รพ. โดยหน่วยเบิกแต่ละหน่วยเบิกจะมีระบบคงคลังเป็นของตัวเอง รูปแบบการทำงานจะคล้ายคลึงกับระบบบริหารคลังเวชภัณฑ์ StockDB เดิมเกือบทั้งหมด โดยสามารถกำหนดรูปแบบการจัดการปริมาณคงคลังเองได้เช่นเดียวกับระบบของส่วนกลาง เช่นการกำหนดจำนวนคงคลังสูงสุด ต่ำสุด การรับเบิกจ่ายเวชภัณฑ์ภายในหน่วยงาน การดูอัตราการเคลื่อนไหวเบิกจ่าย รายงานเวชภัณฑ์หมดอายุ และใบคุมคลังกำกับเวชภัณฑ์แต่ละตัว เป็นต้น โดยระบบจะ sync ข้อมูลรายการเวชภัณฑ์จากส่วนกลางกรณีที่มีการเพิ่มรายการเวชภัณฑ์ใหม่ๆเข้าไปในระบบ

ระบบคงคลังของหน่วยเบิก จะทำงานผ่านโปรแกรมแบบ web base ซึ่งพัฒนาขึ้นมาจากภาษา TypeScript โดยใช้ Angular framework เป็นตัวช่วยกำหนดกรอบในการพัฒนา ทำให้ผู้ใช้งานสามารถทำงานระยะไกลผ่าน Internet ได้ ซี่งเป็นการลดข้อจำกัดในการใช้งานของหน่วยงานภายนอก เช่น รพ.สต. หรือหน่วยงานในกำกับ ที่ไม่ได้อยู่ในเครือข่ายท้องถิ่น (Local area network) ของทาง รพ. ได้ การทำรายการขอเบิกจะสามารถทำได้เฉกเช่นเดียวกับหน่วยงานภายใน เช่น หอผู้ป่วย ทำให้ระบบบริหารจัดการคลังเวชภัณฑ์ของหน่วยงานทั้งหมด เป็นไปในทิศทางเดียวกันทั้งระบบ การทำรายการขอเบิกเวชภัณฑ์จากส่วนกลางสามารถกระทำผ่านหน้าโปรแกรม โดยเมื่อทำรายการแล้ว ตัวรายการจะไปปรากฏที่ระบบส่วนกลางโดยอัตโนมัติ เพื่อให้ทางคลังเวชภัณฑ์ส่วนกลางทำการอนุมัติ และดำเนินการนำส่ง เพื่อให้ทางหน่วยเบิกได้ทำการรับเวชภัณฑ์ต่อไปได้

**รายงานและระบบติดตามความเคลื่อนไหว**

ระบบรายงาน โปรแกรมได้มีการจัดเตรียมรายงานต่างๆ ดังนี้

* สำหรับหน่วยเบิก ได้แก่ รายงานคงคลัง, อัตราการเบิก, รายการรับเข้า, รายการเบิกจ่าย และใบคุมคลัง
* สำหรับส่วนกลาง ได้แก่ รายงานคงคลัง, ใบคุมคลัง, สรุปรับตามหมวด, สรุปเบิกตามหมวด, สรุปรวมรับวัสดุ, สรุปรวมเบิกวัสดุ, สรุปประจำเดือน และ แผนจัดซื้อ

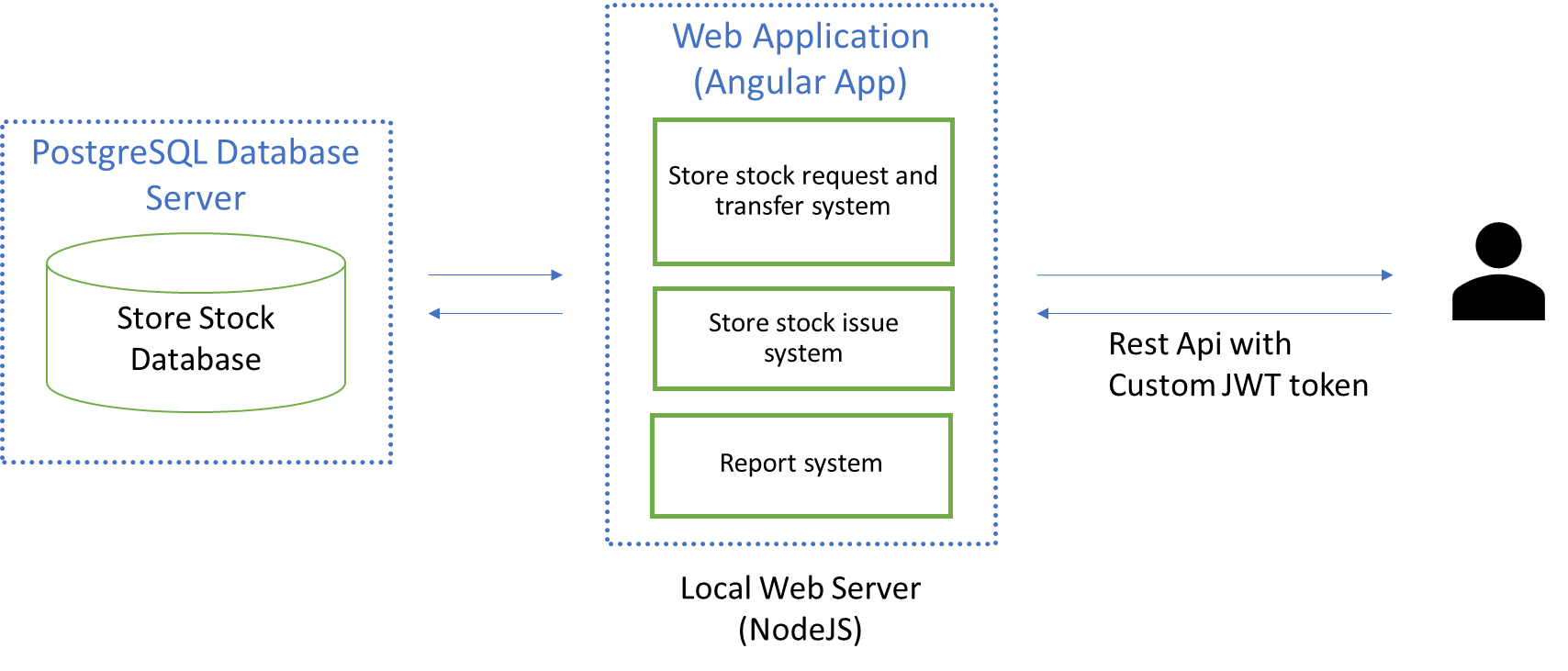
นอกจากนี้ ระบบยังมีผู้ใช้งานที่เป็นผู้สังเกตการณ์ได้ โดยสามารถเข้าไปใช้งานได้โดยที่ไม่สามารถปรับเปลี่ยนแก้ไขข้อมูลได้ เหมาะสำหรับผู้บริหาร หรือหัวหน้าหน่วยงาน ที่ต้องการทราบความเคลื่อนไหว หรือเฝ้าติดตามการดำเนินการต่างๆ ภายในระบบได้

**ระบบสื่อสารและแจ้งเตือนระหว่างหน่วยเบิกและส่วนกลาง**

ตัวโปรแกรมได้มีการพัฒนาในส่วนของการแจ้งเตือนผ่านทาง Line application โดยส่ง message ผ่านทาง Line notification ทุกครั้งที่มีการทำรายการขอเบิก และ ทุกครั้งที่มีการทำรายการนำส่ง เพื่อแจ้งเตือนให้กับหน่วยงานที่เกี่ยวข้อง ซึ่งสามารถตั้งค่าการแจ้งเตือนให้กับหลายหน่วยงานพร้อมกัน หรือแยกตามหน่วยงานได้

**Technology Architect**

ระบบพัฒนาเป็น web base application โดยจัดการวางไว้ที่ Local Server ชองกลุ่มงานเภสัชกรรม โดยมีสถาปัตยกรรมดังต่อไปนี้

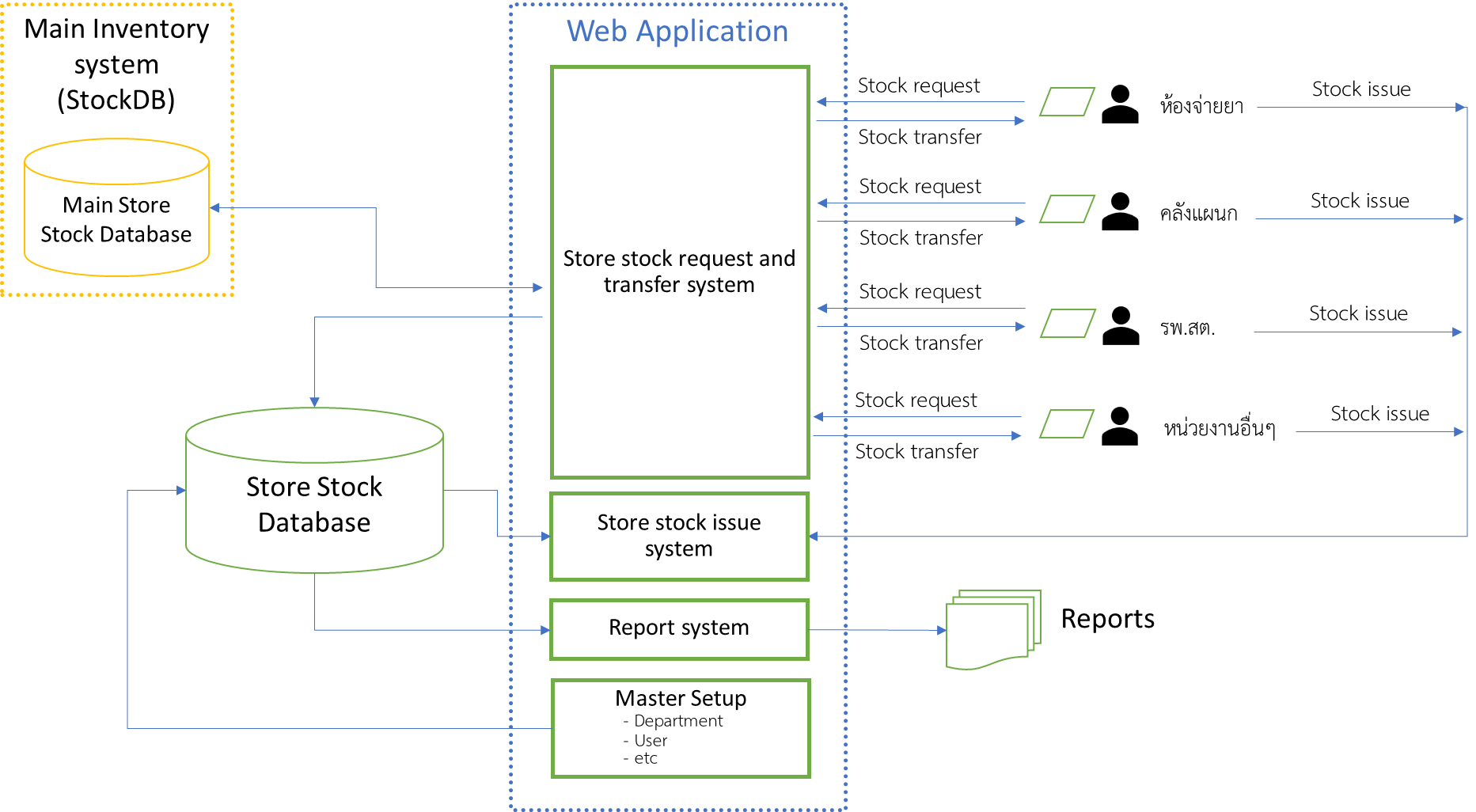


1. Database server ใช้สำหรับเก็บข้อมูลทั้งหมดของโครงการ เลือกใช้เทคโนโลยี่ของ PostgreSQL ที่เป็น RDMS Database เพื่อความรวดเร็วในการเข้าถึงและจัดเก็บข้อมูล โดยข้อมูลส่วนใหญ่เป็นข้อมูลรายการรับเบิกจ่ายวัสดุจากหน่วยงานต่างๆ รวมถึงข้อมูลผู้ใช้งาน ข้อมูลหน่วยงาน และข้อมูลอื่นๆ ข้อมูลทั้งหมดได้แบ่งการจัดเก็บพร้อมกับทำดัชนีเพื่อช่วยให้การสืบค้นได้อย่างรวดเร็ว ตัวฐานข้อมูลติดตั้งอยู่บนพื้นที่จัดเก็บเดียวกับตัว Web Server และเชื่ออมต่อกับ Web Server ในลักษณะ local connection เท่านั้น ไม่มีการ expose connection ของฐานข้อมูลออกสู่ภายนอกแต่อย่างใด การบันทึกข้อมูลต้องกระทำผ่านทาง Web Server ได้แต่เพียงอย่างเดียว เพื่อเป็นการรับประกันความปลอดภัยของตัวฐานข้อมูล และการจัดเก็บรหัสผ่านจะเก็บไว้ในรูปแบบของ bcrypt encryption ซึ่งเป็นรูปแบบการเข้ารหัสที่เป็นที่ยอมรับว่ายากต่อการ brute force หรือการใช้ rainbow table ในการถอดรหัสผ่าน

2. Local Web Server เป็นส่วนที่ให้บริการหน้า web และจัดเก็บ API backend และมีส่วนติดต่อกับ Database โดยที่เป็นส่วนที่ทำงานเกี่ยวกับ working logic ต่างๆในการนำข้อมูลมาวิเคราะห์และสังเคราะห์ก่อนนำไปเก็บหรือส่งให้กับผู้ใช้ผ่านทาWeb Frontend ต่อไป เทคโนโลยี่ในส่วนนี้พัฒนาด้วย NodeJS โดยใช้ Express framework เป็น web server ในการจัดการ route และ end point ต่างๆ ในการเชื่อมต่อกับ Frontend ผ่าน protocol HTTP ด้วย REST API โดยส่งข้อมูลในรูปแบบของ JSON เพื่อให้ง่ายในการจัดการและเป็นมาตรฐานที่ใช้ในงาน web application ในปัจจุบัน โดยมีการรับการเชื่อมต่อกับทาง Frontend ซึ่งยอมรับเฉพาะการ request ที่ต้องผ่านการตรวจสอบสิทธิ์การใช้งานผ่านทางเทคโนโลยี่ของ JSON Web Token (jwt) ร่วมกับ custom encryption ที่พัฒนามาใช้งานในโครงการนี้เป็นการเฉพาะ

ในส่วนที่เป็น Frontend เป็นส่วนของ user interface ที่ผู้ใช้งานจะกระทำการใช้งานต่อตัวโปรแกรมโดยตรง ในส่วนนี้ถูกพัฒนาโดย Angular Framework ซึ่งถือว่าเป็น JavaScript Framework ที่จะถูกเลือกใช้ในอันดับต้นๆ ในการพัฒนาระบบ web ui ในระดับ Enterprise ตัว Frontend พัฒนาในรูปแบบ single page application ทำงานบน web browser ตอบสนองต่อการใช้งานที่รวดเร็วและทำงานได้ด้วยตัวเองจากทางฝั่งของผู้ใช้งาน โดยลดความจำเป็นในการติดต่อกับตัว web server ให้น้อยลง ทำให้มีประสิทธิภาพสูงและตอบสนองได้เร็ว application ได้ถูกออกแบบให้เป็น stateless application ที่จะไม่เก็บข้อมูล tracking ใดๆไว้ที่ browser ทำให้ไม่เหลือร่องรอยข้อมูลใดๆหลังจากเลิกใช้งานแล้ว เพื่อเป็นการป้องกันข้อมูลที่มีความอ่อนไหวที่จะถูกโจรกรรมได้ ระบบจะเข้าใช้งานได้เฉพาะผู้ใช้งานที่มี username และ password ตามที่กำหนดไว้เท่านั้น ไม่ได้เปิดให้บุคคลภายนอกได้เข้าใช้งาน และตัว web application เปิดให้ใช้งานแบบจำกัดเวลาในช่วงของ Office Hour ในวันทำการเท่านั้น

**Data Flow Diagram**

**…**