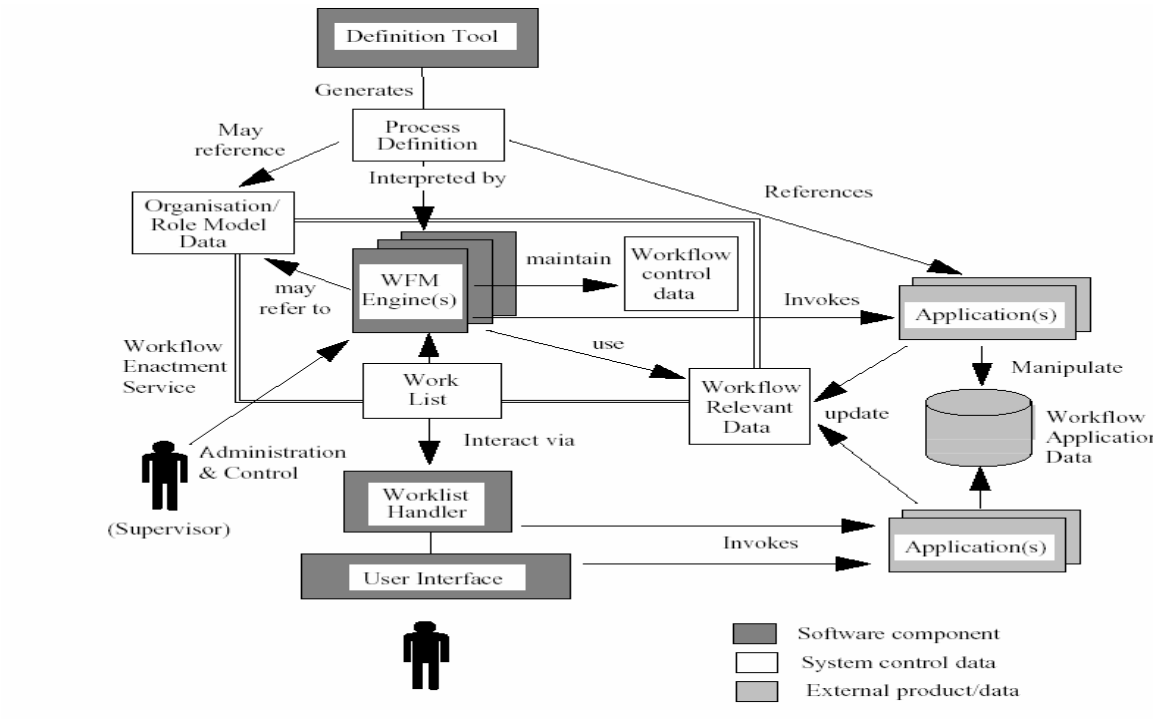
บทที่ 3

การวิเคราะห์และออกแบบ

3.1 ภาพรวมของระบบการไหลของงานเพื่อการจัดการเอกสารในองค์กร

ในการวิเคราะห์ออกแบบระบบเพื่อการบริหารจัดการเอกสารนั้น เนื่องจากระบบทั้งระบบประกอบด้วยส่วนต่าง ๆ หลายส่วนเข้าด้วยกันดังที่กล่าวมาในบทที่ 2 ซึ่งระบบงานในองค์กรการดำเนินงานทางเอกสารเป็นเรื่องที่สำคัญที่ทำให้เอกสารดำเนินไปได้อย่างถูกขั้นตอน ผู้จัดทำจึงเล็งเห็นความสำคัญของการพัฒนาระบบการจัดการเอกสาร ในส่วนของระบบจัดการการไหลของงาน หลังจากที่ผู้จัดทำได้ศึกษาและค้นคว้าข้อมูล และศึกษาผลิตภัณฑ์ในท้องตลาดของระบบจัดการเอกสาร และระบบการไหลของงาน ผู้จัดทำสามารถจึงได้เลือกพัฒนาพัฒนาระบบการจัดการเอกสารในส่วนของระบบจัดการการไหลของงาน ซึ่งมีองค์ประกอบต่างๆ ตามรูปที่ 3-1 ดังนี้



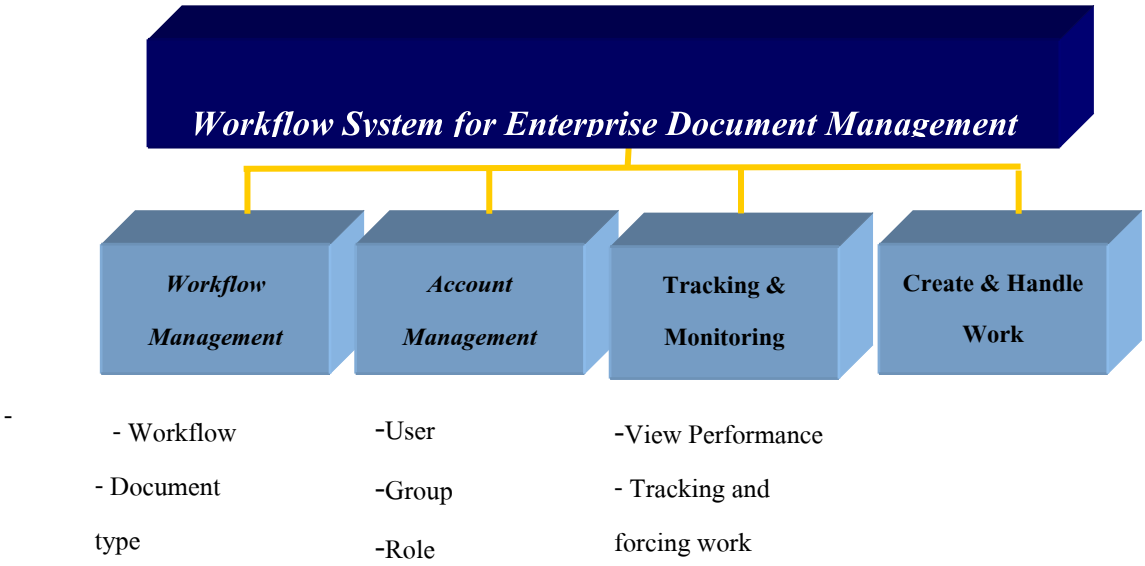
รูปที่ 3-1 ภาพรวมของระบบการไหลของงานเพื่อการจัดการเอกสารในองค์กร

1. ข้อกำหนดของขั้นตอนการไหลของเอกสาร (Process Definition) คือ การนำเอาขั้นตอน (Process) ออกมาวิเคราะห์และทำออกมาให้อยู่ในรูปที่เป็นการไหลของเอกสารที่จำนำไปใช้ได้ โดยใช้เครื่องมือ (Tool) ช่วย เป็นข้อกำหนดของข้อมูล สำคัญของขั้นตอน (process) ว่าจะให้ซอฟท์แวร์ที่ใช้ในการตอบสนองการไหลของงาน (workflow enactment software) ประมวลผลได้หรือไม่ โดยข้อมูลนี้ได้แก่ เงื่อนไขในการเริ่มต้นและจบการไหลของงาน (Workflow), ส่วนประกอบของกิจกรรม (activities) ทั้งหมด และกฎ (rules) เพื่อการค้นหาข้อมูลใน (Process), งานของผู้ใช้งานที่ได้รับ, การอ้างอิงถึงแอปพลิเคชั่น (application) ที่อาจจะมีขึ้น และการกำหนดรายละเอียดของข้อมูล

ที่เกี่ยวข้องอยู่ใน workflow ที่อาจจะมีการถูกอ้างอิงขึ้นมา เป็นต้น ข้อกำหนดของขั้นตอนการไหลของเอกสาร (Process Definition) อาจจะหมายถึง Organization/Role model ที่มีข้อมูล ที่เกี่ยวกับโครงสร้างและบทบาทขององค์กร ซึ่งทำข้อกำหนดของขั้นตอนการไหลของเอกสาร (Process Definition) สามารถเจาะจงลงไปใน organizational entities และ role functions ที่เกี่ยวเนื่องกับกิจกรรมบางอย่าง หรือ จุดมุ่งหมายที่สนใจ มากกว่าที่จะเจาะจงลงไปที่ผู้ใช้ระบบ โดยที่บริการที่ใช้ในการตอบสนองการไหลของงาน (workflow enactment software)จะทำหน้าที่ในการเชื่อมโยงระหว่าง organization entities และบทบาทของผู้ใช้ระบบ เข้ากับ (workflow runtime environment)

2. แอปพลิเคชั่น (Workflow Client Application And Invoked Applications) มีอยู่ 2 ส่วนที่สำคัญ
   a. Workflow systems integrators (WfSi) คือการที่จะ Implement ให้ Front-End Application ที่จะ Manage Workflow หรือถูกเรียกขึ้นมาโดยการไหลของเอกสารที่ให้มันรู้จักกันได้
   b. The workflow API (WAPI) คือตัวที่ทำให้การไหลของเอกสารสามารถติดต่อกับ Application ภายนอกได้
3. เอนจินการไหลของงาน (workflow Engine(s)) คือ อินเตอร์เฟซ (interface) ที่สามารถทำให้ การไหลของเอกสารของทุก Brand ติดต่อกันได้ และสามารถรับสถานะระหว่างการไหลของเอกสาร ได้ว่าสำเร็จกระบวนการ (Complete Process) หรือไม่
4. การติดตามและการดูแลการไหลของเอกสาร (Tracking And Monitoring) คือการติดตามตรวจการทำงานของ การไหลของเอกสารว่าเป็นอย่างไร

3.2 ฟังก์ชันการทำงานของระบบการไหลของงานเพื่อการจัดการเอกสารในองค์กร



รูปที่ 3-2 ฟังก์ชันการทำงานของระบบการไหลของงานเพื่อการจัดการเอกสารในองค์กร

ในการออกแบบระบบจัดการเอกสารนั้นได้นำหลักการในการการไหลของงานใช้ ดังนั้นการกำหนดฟังก์ชันการทำงานของระบบจัดการเอกสารจะแบ่งหน้าที่การทำงานหลัก ๆ ออกเป็นหมวดหมู่ดังแสดงในรูปที่ 3-2 โดยในการกำหนดฟังก์ชันการทำงานได้ทำการกำหนดฟังก์ชันที่จำเป็น โดยคำนึงถึงความเป็นไปได้และ

เลือกฟังก์ชันที่จำเป็นสำหรับการจัดการเอกสารทั่วไป โดยได้ตัดในส่วนของฟังก์ชันที่มีความเฉพาะเจาะจงออกไป โดยมีรายละเอียดดังนี้

1. ระบบการไหลของงาน(Workflow management) เป็นส่วนที่ทำหน้าที่ในการกำหนดขั้นตอนในการไหลของเอกสาร หรือเรียกว่า Admin Definition Tool ซึ่งหลังจากที่ได้มีการวิเคราะห์กระบวนการการไหลของเอกสารแล้ว ผู้ดูแลระบบก็จะนำผลที่ได้จากการวิเคราะห์มาใส่ลงในฐานข้อมูล ซึ่งแบ่งออกเป็น 2 ส่วนดังนี้
    - ชนิดของเอกสาร (Document type) เป็นฟังก์ชันในการกำหนดชนิดของเอกสาร เพราะว่าทุกครั้งที่สร้างการไหลของงาน (workflow) ขึ้นมาแต่ละครั้งการไหลของงาน (workflow) นั้นๆ จะต้องมีชนิดของเอกสารอยู่ด้วย ถ้ามีการลบหรือแก้ไขแต่มีเส้นทางการไหลของงานอ้างอิงอยู่มันก็จะถูกลบไปด้วย
    - การไหลของงาน (workflow) เป็นฟังก์ชันในการกำหนดกระบวนการการไหลของงาน ในส่วนของการกำหนดกระบวนการการไหลของงานนี้จะประกอบไปด้วยข้อมูลหลายชนิด ได้แก่ ชนิดของข้อมูล (Document type), สแตท (State), เงื่อนไขการเปลี่ยนสแตท และสิทธิ์ของผู้ใช้งานในระบบว่าจะสามารถเข้ามาทำงานอะไรในส่วนของสแตท หรือส่วนของการไหลของงาน (workflow) ใดได้บาง และเมื่อมีการเปลี่ยนแก้ไขงานที่ยังคงค้างอยู่ในสถานะใดๆ ที่เราลบงานจะหายไป แต่ถ้าไม่มีงานค้างอยู่ก็จะมีผลอะไร ส่วนการลบก็เช่นกัน
2. การจัดการระบบผู้ใช้งาน (Account management) เป็นส่วนที่ทำหน้าที่ในการจัดการเกี่ยวกับข้อมูลผู้ใช้งานในระบบพร้อมทั้งยังเก็บสิทธิ์ว่าผู้ใช้งานนั้นมีความสามารถที่จะลบงานที่ถูกส่งเข้ามาได้หรือไม่โดยมีส่วนหลักๆ ดังนี้
    - ผู้ใช้งาน(User) ฟังก์ชันนี้มีไว้เพื่อให้ผู้ดูแลระบบสร้าง แก้ไข และลบข้อมูลของผู้ใช้งานในระบบ และนอกจากนี้ผู้ใช้ระบบ ยังสามารถที่จะแก้ข้อมูลของตนเองได้ด้วย
    - กลุ่มผู้ใช้งาน(Group) ฟังก์ชันนี้มีไว้เพื่อให้ผู้ดูแลระบบสร้าง แก้ไข และลบข้อมูลของกลุ่มผู้ใช้งานในระบบ โดยหนึ่งกลุ่มสามารถประกอบไปด้วยผู้ใช้งานหลายคน
    - บทบาทผู้ใช้งาน(Role) ฟังก์ชันนี้มีไว้เพื่อให้ผู้ดูแลระบบสร้าง แก้ไข และลบข้อมูลของบทบาทผู้ใช้งานในระบบ โดยหนึ่งบทบาทสามารถประกอบไปด้วยผู้ใช้งานหลายคน และกลุ่มผู้ใช้งานหลายกลุ่ม
3. ติดตามงาน และตรวจสอบประสิทธิภาพ (Tracking & Monitoring)
    - ติดตามงาน (Tracking) เป็นส่วนที่ทำหน้าที่ในการจัดการเกี่ยวกับการไหลของเอกสารของแต่ละ Workflow ว่าเป็นอย่างไรมันไปอยู่ที่สแตทใด โดยฟังก์ชันพัฒนาไว้ให้ผู้ดูแลระบบใช้ในการบังคับ (force) ให้แต่ละงานสามารถเปลี่ยนสแตทได้
    - การเฝ้าดู (Monitoring) การจัดการควบคุมประสิทธิภาพของระบบ(Performance management) กลุ่มของฟังก์ชันการทำงานเพื่อทำให้ผู้ใช้งานระบบสามารถที่จะดูสถานะของงานที่มีอยู่ในแต่ละการไหลของงาน (workflow) ว่าความเป็นไป ความเปลี่ยนแปลงเป็นเช่นใด ผู้ใช้งานสามารถที่จะดูค่าของงานที่มีอยู่ในแต่ละ Workflow ที่ต้องการได้ โดยสามารถที่จะเลือกวิธีการในการนำเสนอ

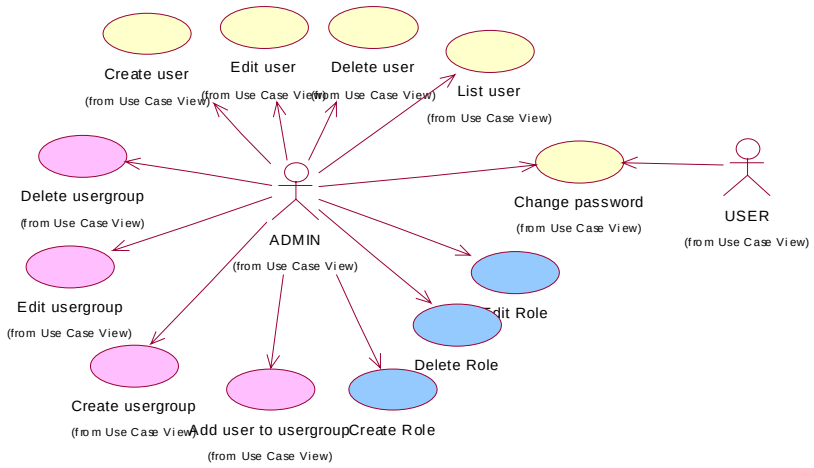
ข้อมูลในรูปแบบที่เหมาะสมกับประเภทของข้อมูลได้ นอกจากนั้นยังสามารถที่จะจัดกลุ่มงานที่มีอยู่ในแต่ละ การไหลของงาน (workflow) เหล่านั้น เข้ามาอยู่ในกลุ่มที่สนใจได้ เพื่อสามารถที่จะนำค่าเหล่านั้นมาเปรียบเทียบเพื่อให้มีความสะดวกในการวิเคราะห์ โดยคำนึงว่าในแต่ละสถานะของการไหลของงาน มีงานค้างหรือเข้าออกตามกำหนดหรือไม่

4. การสร้างงานและวิธีจัดการกับงาน(Create and Handle work) เป็นส่วนที่ทำหน้าที่ในการจัดการเกี่ยวกับงานที่เกิดขึ้นโดยส่วนนี้เป็นส่วนสำหรับผู้ใช้งานโดยผู้ใช้งาน แบ่งเป็น 2 ส่วน ดังนี้
   - การสร้างงาน (Create work) ส่วนสำหรับการสร้างงาน ด้วยการแนบเอกสารเข้าไปในระบบการไหลของเอกสาร เพื่อให้เอกสารนั้นไหลไปตามเงื่อนไขที่ได้กำหนดไว้ในระบบ
   - วิธีจัดการกับงาน (Handle work) ที่มีสิทธิ์เข้ามาทำงานของในแต่สแตทนั้นมีฟังก์ชันดังนี้ Submit รับงานเข้ามา, ลบงานทิ้งออกจากระบบ, ส่งงานกลับไปยังสแตทก่อนหน้านี้

3.3 การออกแบบระบบการไหลของงานเพื่อการจัดการเอกสารในองค์กร

3.3.1 Use case diagram

เป็นไดอะแกรมที่แสดงฟังก์ชันการทำงานที่สำคัญ ๆ ของระบบ ดังแสดงในรูป 3-4 ซึ่งจะมีรายละเอียดแสดงไว้ต่อไป



รูปที่ 3-3 Use Case Diagram ของระบบจัดการผู้ใช้งาน (Account Management)

ระดับผู้ดูแลระบบ (Administration Level)

Use case	User management (Admin)
ผู้ใช้	Administrator
เงื่อนไข	ใช้สร้าง, แก้ไข, ลบ user ในระบบ
ข้อมูลที่เกี่ยวข้อง	Username, password

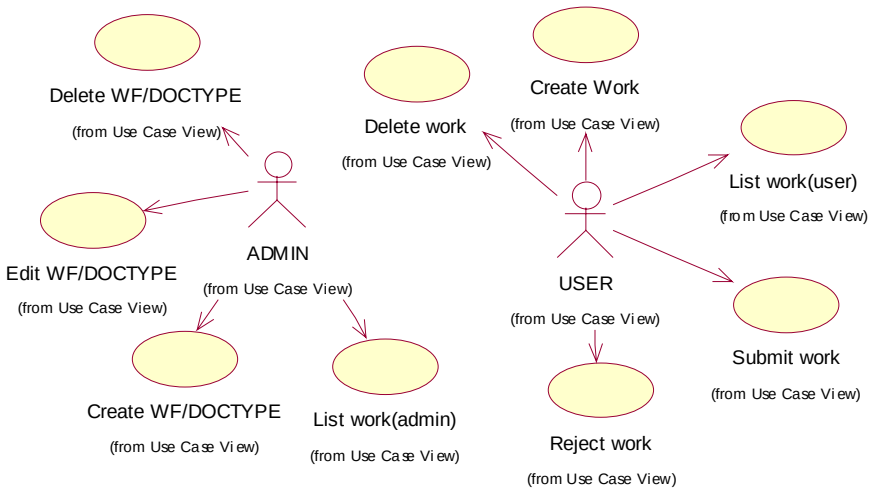
Use case	User group management (Admin)
ผู้ใช้	Administrator
เงื่อนไข	ใช้สร้าง, แก้ไข, ลบ usergroup ในระบบ และ map user เข้ากับ usergroup เพราะ 1 usergroup มีได้หลาย user
ข้อมูลที่เกี่ยวข้อง	Username, user group name, password

Use case	Role management (Admin)
ผู้ใช้	Administrator
เงื่อนไข	ใช้สร้าง, แก้ไข, ลบ role ในระบบ และ map user, usergroup เข้ากับ role เพราะ 1 role มีได้หลาย user, usergroup
ข้อมูลที่เกี่ยวข้อง	Username, user group name, role, password

Use case	Change password(Admin)
ผู้ใช้	Administrator
เงื่อนไข	ใช้แก้ไข password ของ admin
ข้อมูลที่เกี่ยวข้อง	Username, password

ระดับผู้ใช้งาน (User level)

Use case	Change password(User)
ผู้ใช้	Administrator
เงื่อนไข	ใช้แก้ไข password ของ user
ข้อมูลที่เกี่ยวข้อง	Username, password



รูปที่ 3-4 Use Case Diagram ของระบบจัดการการไหลของเอกสาร (Workflow Management)

ระดับผู้ดูแลระบบ (Administration Level)

Use case	Document type(Admin)
ผู้ใช้	Administrator
เงื่อนไข	ใช้สร้าง, แก้ไข, ลบ ชนิดของเอกสาร (Document type)
ข้อมูลที่เกี่ยวข้อง	Document type, Attribute, value of attribute

Use case	Workflow (Admin)
ผู้ใช้	Administrator
เงื่อนไข	ใช้สร้าง, แก้ไข, ลบเส้นทางการไหลของงาน (workflow)
ข้อมูลที่เกี่ยวข้อง	Document type, state of workflow, condition to change state

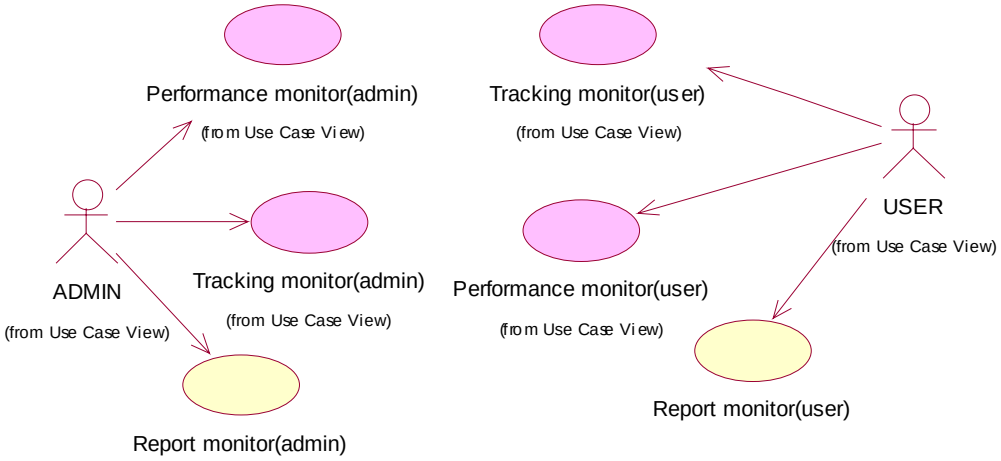
Use case	List work (Admin)
ผู้ใช้	Administrator
เงื่อนไข	ดูงานทั้งหมดในระบบ
ข้อมูลที่เกี่ยวข้อง	Work, document, username

ระดับผู้ใช้งาน (User level)

Use case	List work (Users)
ผู้ใช้	Users
เงื่อนไข	ดูงานทั้งหมดในระบบ
ข้อมูลที่เกี่ยวข้อง	Work, document, username
Use case	Submit work (Users)
ผู้ใช้	Users
เงื่อนไข	เมื่องานเข้ามาอยู่ใน Work list ของตนแล้ว user ก็สามารถที่จะรับงานเข้ามาทำด้วยการ download มาทำเมื่อทำเสร็จก็ส่งต่อไปให้คนอื่นด้วยการ submit
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร, username

Use case	Turn back work (Users)
ผู้ใช้	Users
เงื่อนไข	เมื่องานเข้ามาอยู่ใน Work list ของตนแล้ว user ก็สามารถที่จะส่งกลับงานไปสแตทก่อนหน้าได้ถ้างานที่เข้าไม่ถูกต้อง
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร, username

Use case	Delete work (Users)
ผู้ใช้	Users
เงื่อนไข	เมื่องานเข้ามาอยู่ใน Work list ของตนแล้ว user ก็สามารถที่จะส่งลบงานได้ ถ้ามีสิทธิ์
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร, username
Use case	Create work (Users)
ผู้ใช้	Users
เงื่อนไข	สร้างงานขึ้นมาได้
ข้อมูลที่เกี่ยวข้อง	Work, document , username



รูปที่ 3-5 Use Case Diagram ของระบบ (Performance and Tracking)

Adminระดับผู้ดูแลระบบ (Administration Level)

Use case	Performance Monitor (Admin)
ผู้ใช้	Administrator
เงื่อนไข	ดูประสิทธิภาพของระบบจัดการเอกสารว่าผู้ใช้งานได้ปฏิบัติงานลุล่วงไปตามกำหนดเวลาที่ได้ตั้งไว้ของแต่ละงานหรือไม่ และเป็นจำนวนเท่าไหร่
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร

Use case	Tracking Monitor (Admin)
ผู้ใช้	Administrator
เงื่อนไข	สำหรับติดตามงาน เพื่อดูว่างานนั้นไปถึงไหนแล้ว admin สามารถบังคับให้งานเปลี่ยนไปยังสเตทอื่นได้
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร

Use case	Report Monitor (Admin)
ผู้ใช้	Administrator
เงื่อนไข	ดูงานแต่ละชิ้นกำลังอยู่ที่สแตทใดและใครเป็นผู้รับผิดชอบ
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร

ระดับผู้ใช้งาน (User level)

Use case	Tracking Monitor (Users)
ผู้ใช้	Users
เงื่อนไข	สำหรับติดตามงาน เพื่อดูว่างานนั้นไปถึงไหนแล้ว Users สามารถบังคับให้งานเปลี่ยนไปยังสแตทอื่นได้
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร

Use case	Performance Monitor (User)
ผู้ใช้	User
เงื่อนไข	ดูประสิทธิภาพของระบบจัดการเอกสารว่าผู้ใช้งานได้ปฏิบัติงานลุล่วงไปตามกำหนดเวลาที่ได้ตั้งไว้ของแต่ละงานหรือไม่ และเป็นจำนวนเท่าไหร่ โดย user แต่ละคนจะมีสิทธิในการดูงานไม่เท่ากัน user ต้องได้สิทธิ์จึงจะดูระบบได้
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร

Use case	Report Monitor (Users)
ผู้ใช้	Users
เงื่อนไข	ดูงานแต่ละชิ้นกำลังอยู่ที่สแตทใดและใครเป็นผู้รับผิดชอบ
ข้อมูลที่เกี่ยวข้อง	Work information, เอกสาร

3.3.2 หลักการออกแบบ

หลังจากที่ได้ทำการวิเคราะห์แล้วผู้จัดทำจึงออกแบบระบบแบ่งออกเป็น 4 ส่วนดังนี้

1. User Interface ใช้สำหรับให้ user ส่ง message ไปติดต่อกับระบบผู้จัดทำพัฒนาด้วยภาษา JSP และ HTMP ซึ่งมันจะไปทำการติดต่อกับ Javabean ที่พัฒนาแบบ object-oriented ด้วยภาษาจาวา
2. Access control มีหน้าที่ในการกำหนดสิทธิ์ว่าผู้ใช้งาน หลังจากที่ user ได้ส่ง message เข้ามาทาง user interface แล้วส่วนของ access control จะทำการประมวลผลว่า user นี้จะมีสิทธิ์ใน method ของ class ที่ ทำหน้าที่ในการประมวลuser ต้องการ access เข้าไปใดได้บ้าง และได้พัฒนาด้วยเทคโนโลยี Javabean เพื่อในอนาคตผู้ที่ต้องการนำไปพัฒนาต่อสามารถนำส่วนนี้ไปเป็น API ได้ ซึ่งทางผู้จัดทำออกแบบ class ดังนี้

- AdminPower class	- ReadUser class	- ReadUserState class	- ReadGroup class
- UserPower class	- WriteUser class	- WriteUserState class	-WriteGroup class
- ReadGroupState class	-ReadUserGroup class	- ReadRoleGroup class	- Readworkflow class
- WriteGroupState class	-WriteUserGroup class	- WriteRoleGroup class	- Writeworkflow class
- ReadRole class	- ReadRoleUser class	- ReadDocumenttype class	- ReadState class
- WriteRole class	- WriteRoleUser class	- WriteDocumenttype class	- WriteState class
- ReadRoleState class	-ReadUserGroup class	- ReadAttribute class	- Readcondition class
- WriteRoleState class	-WriteUserGroup class	- WriteAttribute class	- Writecondition class
- ReadWorkAttribute class	-WriteWorkAttribute class	- ReadWork class	- WriteWork class

สมมติว่า User ที่เป็น Admin ส่ง message ไปเรียกใช้ method ใน Work class (ซึ่งอยู่ในส่วนของการประมวลผล) ส่วน access control admin ก็จะถูกจัดในประเภทของ AdminPower class จะทำการตรวจสอบสิทธิ์ว่า admin นั้นมีสิทธิ์ในการ Read Write หรือไม่ถ้าใช่ และมันก็จะ access ข้อมูลผ่านทาง method ของ WriteWork และ ReadWork

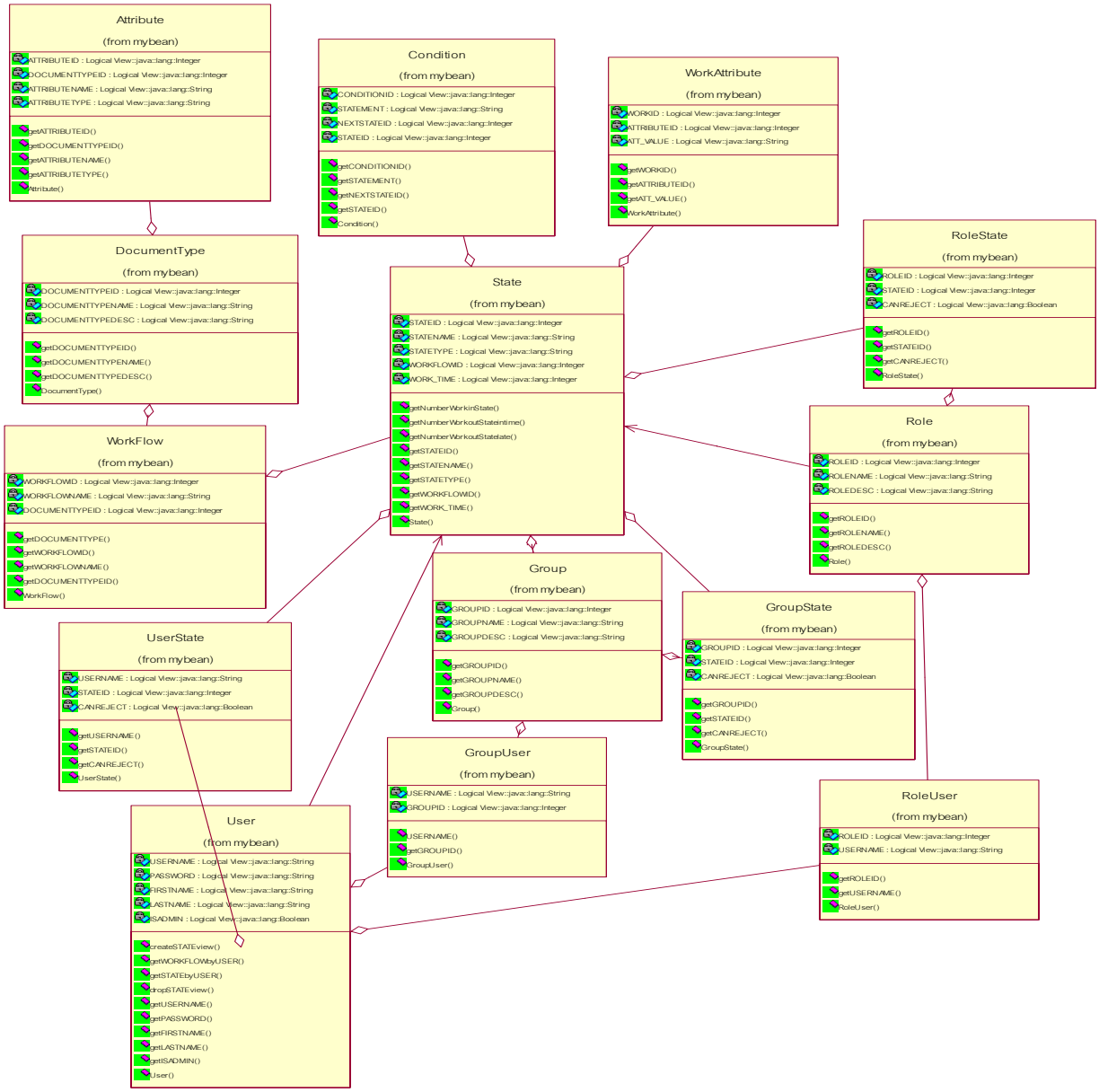
3. ประมวลผล มีเพื่อประมวลผล message ที่ user ส่งเข้ามาผ่านทางส่วน access control โดยส่วนนี้เป็นส่วนที่ทำการขับเคลื่อนระบบให้ถูกต้องเป็นไปตามระบบการไหลของงาน ซึ่งประกอบไปด้วย

- GroupState class	- User class	- UserState class	- Group class
- Role class	- UserGroup class	- RoleGroup class	- workflow class
- RoleState class	- RoleUser class	- Documenttype class	- State class
- WorkAttribute class	- UserGroup class	- Attribute class	- condition class
- WorkAttribute class	- Work class	- Work class	

4. ติดต่อ Database มีไว้เพื่อเป็นตัวกลางในการติดต่อระหว่างระบบฐานข้อมูลกับส่วน access control และส่วนประมวลผล ที่ผู้จัดทำพัฒนาไปในแนวทางนี้เพื่อในอนาคต ผู้ที่ต้องการนำระบบไปพัฒนาต่อนั้นจะได้สามารถเปลี่ยนไปใช้งานระบบฐานข้อมูลอื่นๆ ได้ที่ไม่ใช่ SQL2000 ที่ใช้ในตอนนี้ ซึ่งทางผู้จัดทำได้ออกแบบ class ไว้ใช้ในการติดต่อฐานข้อมูล คือ Database class

- ความสัมพันธ์ของคลาสต่างๆ ในระบบ

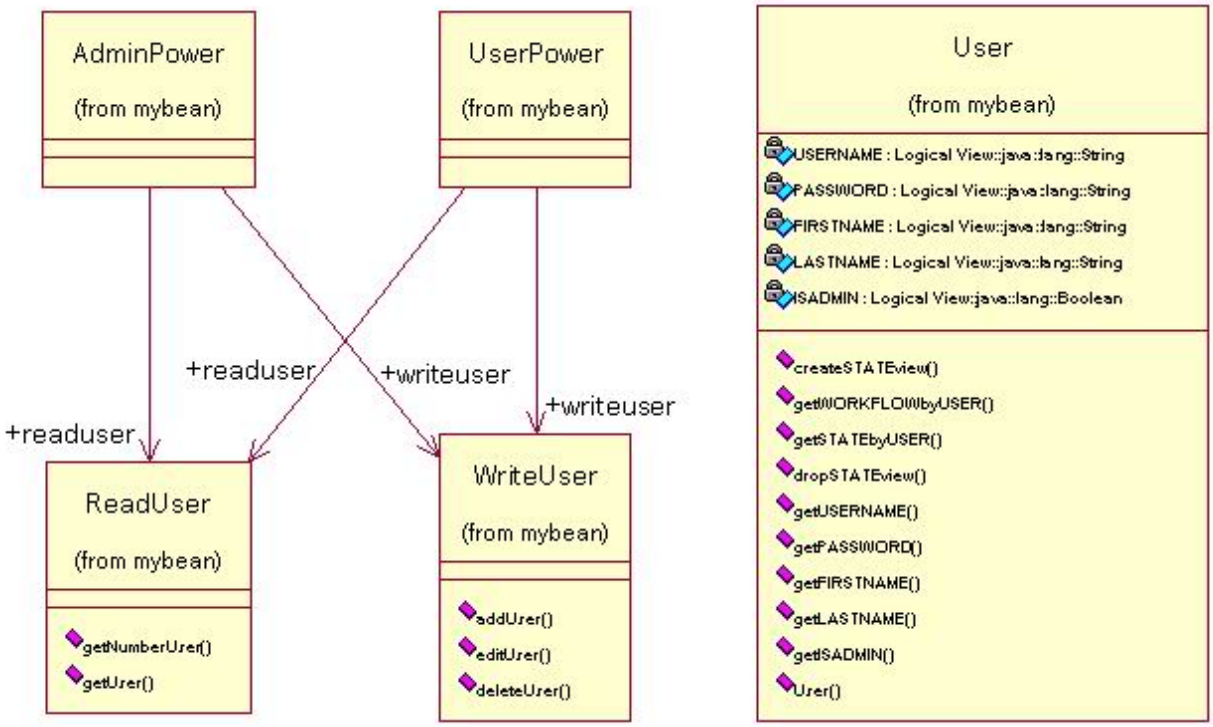
1. ประมวลผล ในรูปที่ 3-6 หมายถึง เอกสารหนึ่งชนิด (Document Type) นั้นมีได้หลายแอททริบิวท์ (Attribute) และเป็นของหนึ่งการไหลของงาน(Workflow)หนึ่งการไหลของงาน(Workflow) นั้นมีหลายสถานะ (State) หนึ่งสถานะ (State)นั้นมีสิทธิ์ของผู้ใช้งานที่เข้าถึงชิ้นงานในแต่ละสถานะ (State)ที่ต่างกันไปซึ่งประกอบไปด้วยผู้ใช้งาน (User), กลุ่มผู้ใช้งาน (Group) ที่มีหลายผู้ใช้งาน (User)ในหนึ่งกลุ่มผู้ใช้งาน (Group) และ บทบาทผู้ใช้งาน (Role)ที่มีหลายผู้ใช้งาน (User) และกลุ่มผู้ใช้งาน (Group) ในหนึ่งบทบาทผู้ใช้งาน (Role) นอกจากนี้ยังประกอบไปด้วยหลายเงื่อนไข (Condition) เพื่อการเปลี่ยนแปลงสถานะ (State) สำหรับตัวเงื่อนไข (Condition)นั้นมันต้องใช้ ค่าในแอททริบิวท์ของงาน (Work Attribute) ของแต่ละงาน (Work)ที่เกาะอยู่กับสถานะ (State) มาตัดสินใจว่าควรจะดำเนินไปในสถานะ (State) ใด



รูปที่ 3-6 ความสัมพันธ์ของคลาสหลักในระบบ

2. Access control ซึ่งแสดงถึงความสัมพันธ์ของส่วนประมวลผลในแต่ละ class กับส่วนของ access control

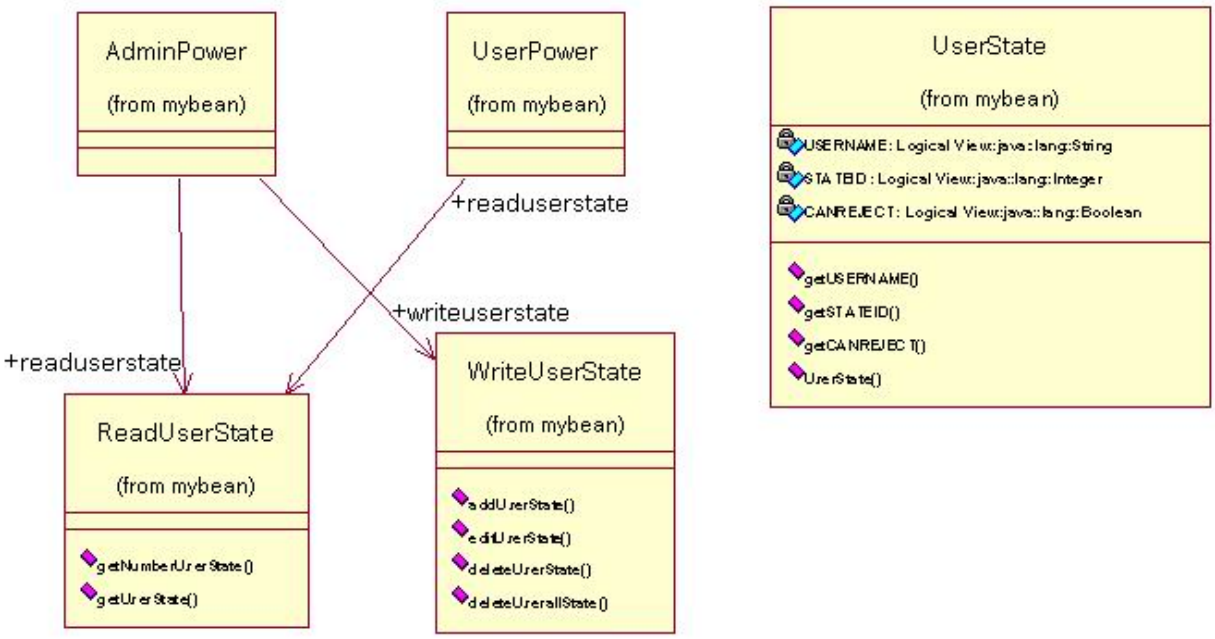
- คลาสที่เกี่ยวกับผู้ใช้ระบบ (User)



รูปที่ 3-7 ใช้ในข้อมูลเกี่ยวกับผู้ใช้ระบบ (User)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ ผู้ใช้ระบบทั้งอ่านเขียนข้อมูลได้เพราะว่า ในส่วนผู้ใช้ระบบนั้นสามารถเปลี่ยนแปลงรหัสผ่านของตนได้
อ่านข้อมูล (Read User)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลผู้ใช้ระบบ(User class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์ผู้ใช้ระบบ (User instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write User)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลผู้ใช้ระบบ(User class)
ผู้ใช้ระบบ (User)	อ๊อปเจคต์ข้อมูลของผู้ใช้ระบบทั้งหมด

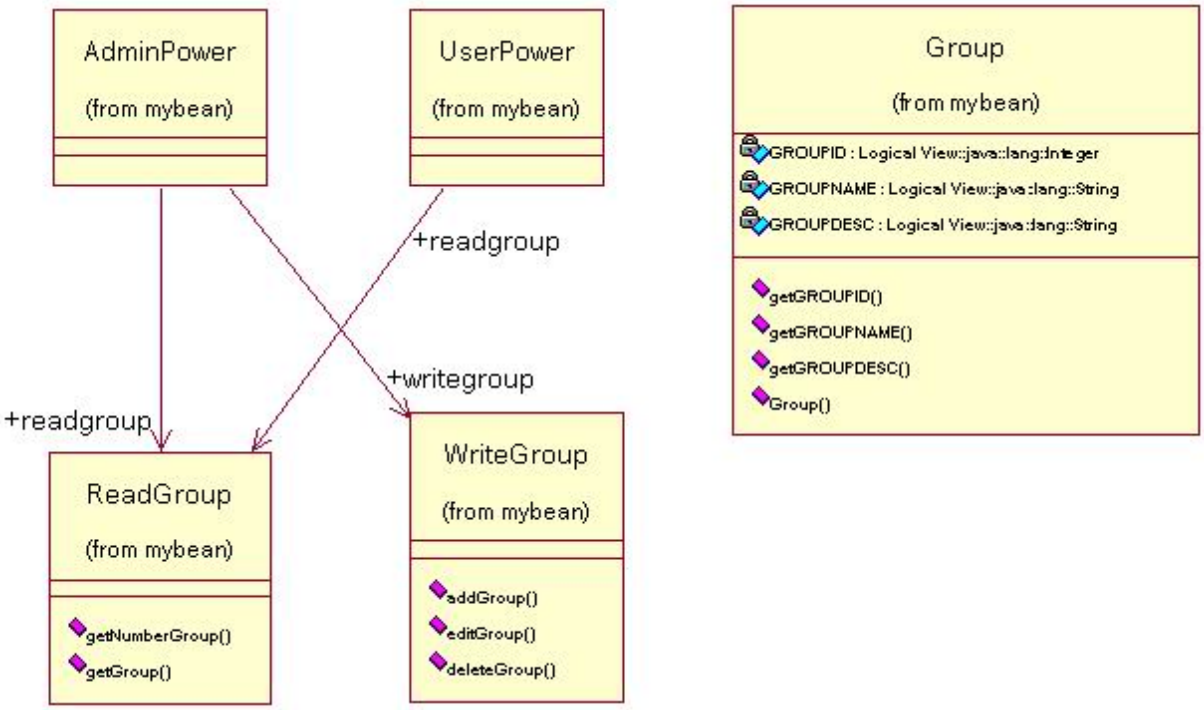
- คลาสที่เกี่ยวกับกลุ่มผู้ใช้ระบบ (User state)



รูปที่ 3-8 ใช้ในข้อมูลเกี่ยวกับบทบาทผู้ใช้ระบบ (UserState)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านได้อย่างเดียว
อ่านข้อมูล (Read UserState)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลสถานะผู้ใช้ระบบ (UserState class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์สถานะผู้ใช้ระบบ (UserState instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write UserState)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลบทบาทผู้ใช้ระบบ(UserState class)
สถานะกลุ่มผู้ใช้ระบบ (UserState)	อ๊อปเจคต์ข้อมูลของผู้ใช้ระบบทั้งหมด ว่ามีสิทธิ์ในการลบงานที่มีอยู่ในระบบได้หรือไม่

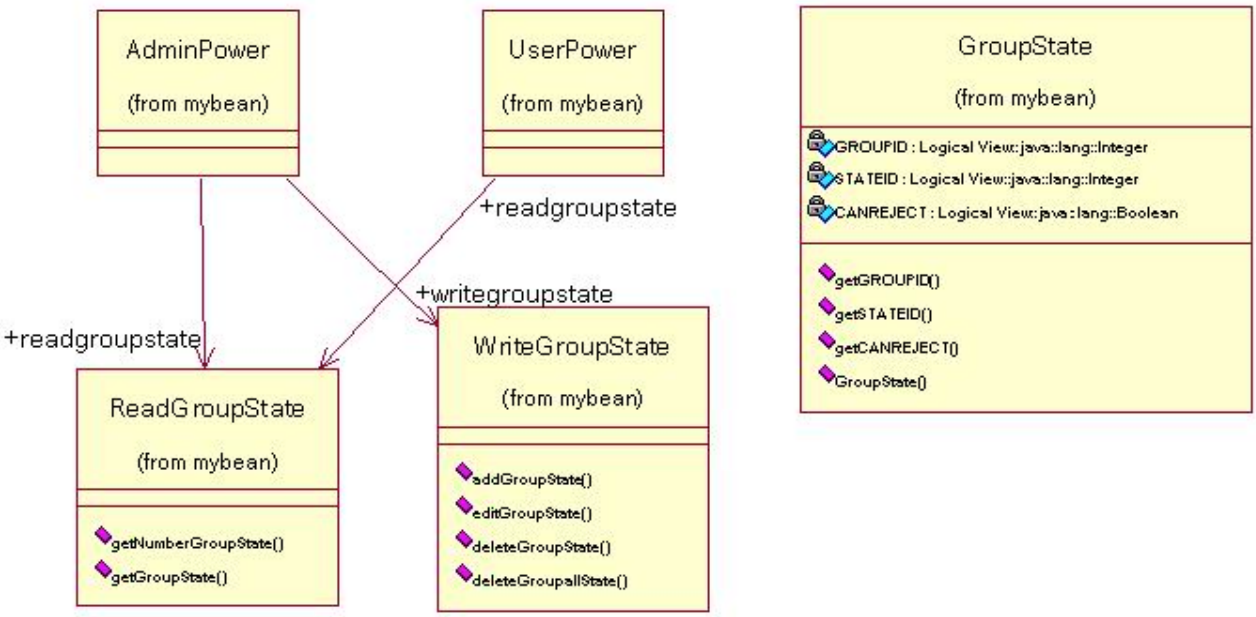
- คลาสที่เกี่ยวกับกลุ่มผู้ใช้ระบบ (User group)



รูปที่ 3-9 ใช้ในข้อมูลเกี่ยวกับกลุ่มผู้ใช้ระบบ (User group)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านและเขียนได้ ที่ผู้ใช้งานสามารถเปลี่ยนแปลงข้อมูลได้เพราะว่า เมื่อผู้ใช้งานลอคอินเข้ามาระบบแล้ว ถ้าหากว่าผู้ใช้มีสิทธิ์ในการเข้าถึงกลุ่มผู้ใช้งานที่ผู้ใช้ต้องการเปลี่ยนแปลงค่าผู้ใช้ก็สามารถเข้าไปทำได้
อ่านข้อมูล (Read group)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลกลุ่มผู้ใช้ระบบ (Group class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์กลุ่มผู้ใช้ระบบ (Group instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write Group)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลกลุ่มผู้ใช้ระบบ(Group class)
กลุ่มผู้ใช้ระบบ (Group)	อ๊อปเจคต์ข้อมูลของกลุ่มผู้ใช้ระบบทั้งหมด

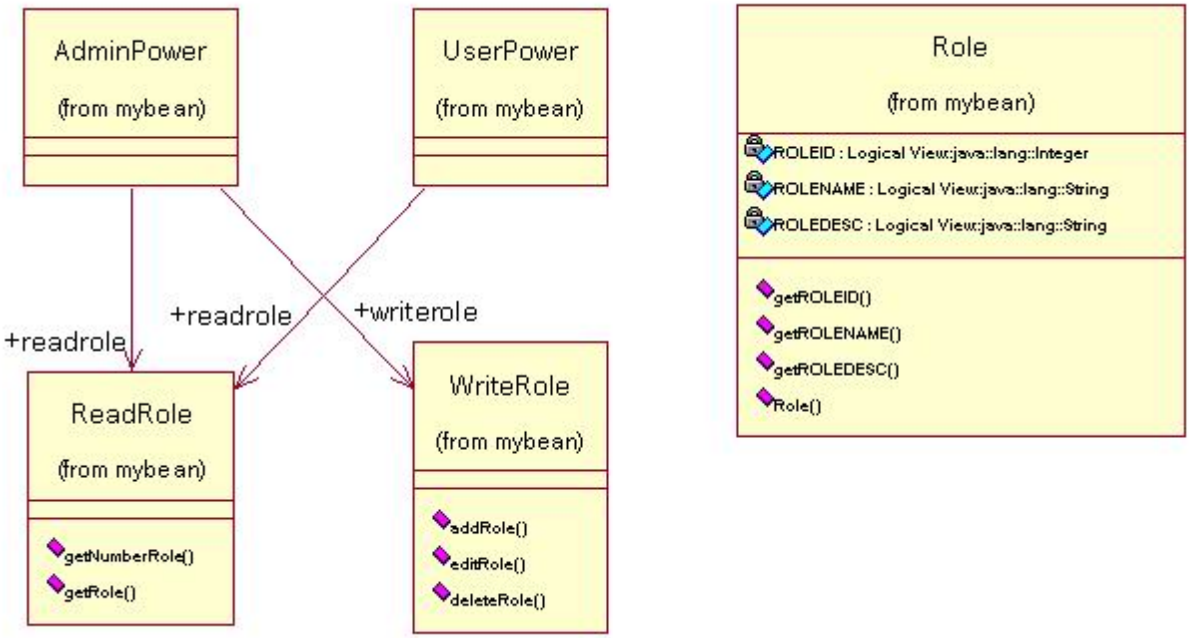
- คลาสที่เกี่ยวกับสถานะภาพกลุ่มผู้ใช้ระบบ (Group State)



รูปที่ 3-10 ใช้ในข้อมูลเกี่ยวกับบทบาทกลุ่มผู้ใช้ระบบ (Group state)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านและเขียนได้ ที่ผู้ใช้งานสามารถเปลี่ยนแปลงข้อมูลได้เพราะว่า เมื่อผู้ใช้งานลอคอินเข้ามาระบบแล้ว ถ้าหากว่าผู้ใช้มีสิทธิ์ในการเข้าถึงกลุ่มผู้ใช้งานที่ผู้ใช้ต้องการเปลี่ยนแปลงค่าผู้ใช้ก็สามารถเข้าไปทำได้
อ่านข้อมูล (Read GroupState)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลสถานะกลุ่มผู้ใช้ระบบ (GroupState class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์สถานะกลุ่มผู้ใช้ระบบ (GroupState instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write GroupState)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลบทบาทกลุ่มผู้ใช้ระบบ(GroupState class)
สถานะกลุ่มผู้ใช้ระบบ (GroupState)	อ๊อปเจคต์ข้อมูลของกลุ่มผู้ใช้ระบบทั้งหมด ว่ามีสิทธิ์ในการลบงานที่มีอยู่ในระบบได้หรือไม่

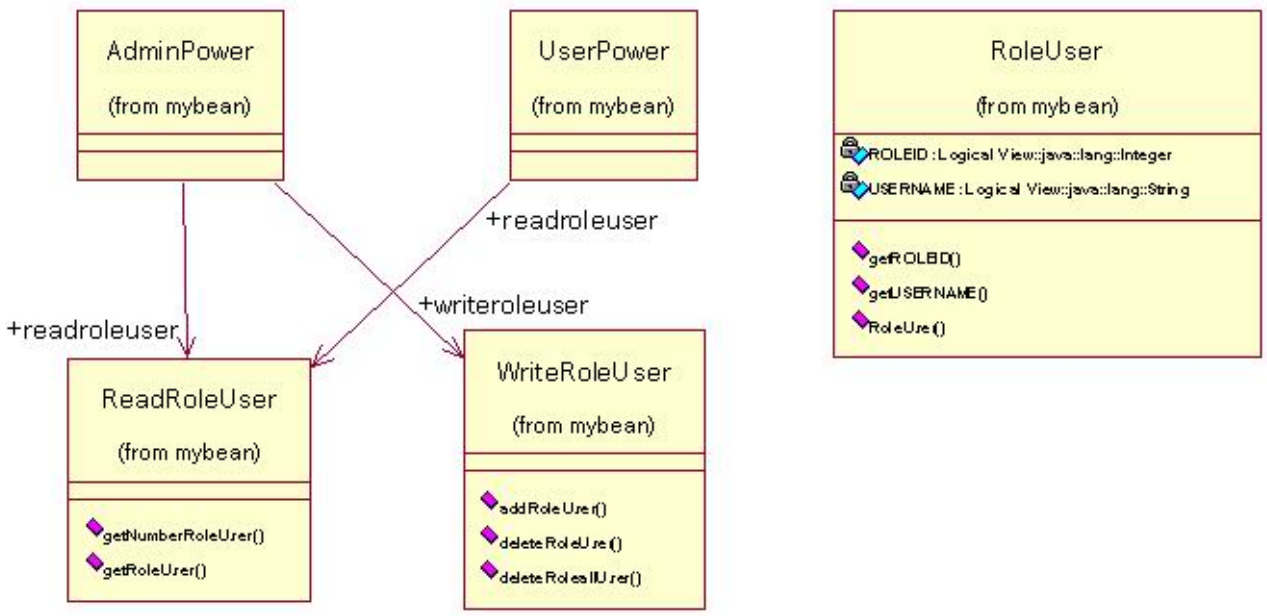
- คลาสที่เกี่ยวกับบทบาทผู้ใช้ระบบ (Role)



รูปที่ 3-11 ใช้ในข้อมูลเกี่ยวกับบทบาทผู้ใช้ระบบ (Role)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านและเขียนได้ ที่ผู้ใช้งานสามารถเปลี่ยนแปลงข้อมูลได้เพราะว่า เมื่อผู้ใช้งานลอคอินเข้ามาระบบแล้ว ถ้าหากว่าผู้ใช้มีสิทธิ์ในการเข้าถึงบทบาทผู้ใช้งานที่ผู้ใช้ต้องการเปลี่ยนแปลงค่าผู้ใช้ก็สามารถเข้าไปทำได้
อ่านข้อมูล (Read Role)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลบทบาทผู้ใช้ระบบ (Role class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์บทบาทผู้ใช้ระบบ (Role instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write Role)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลบทบาทผู้ใช้ระบบ(Role class)
กลุ่มผู้ใช้ระบบ (Role)	อ๊อปเจคต์ข้อมูลของบทบาทผู้ใช้ระบบทั้งหมด

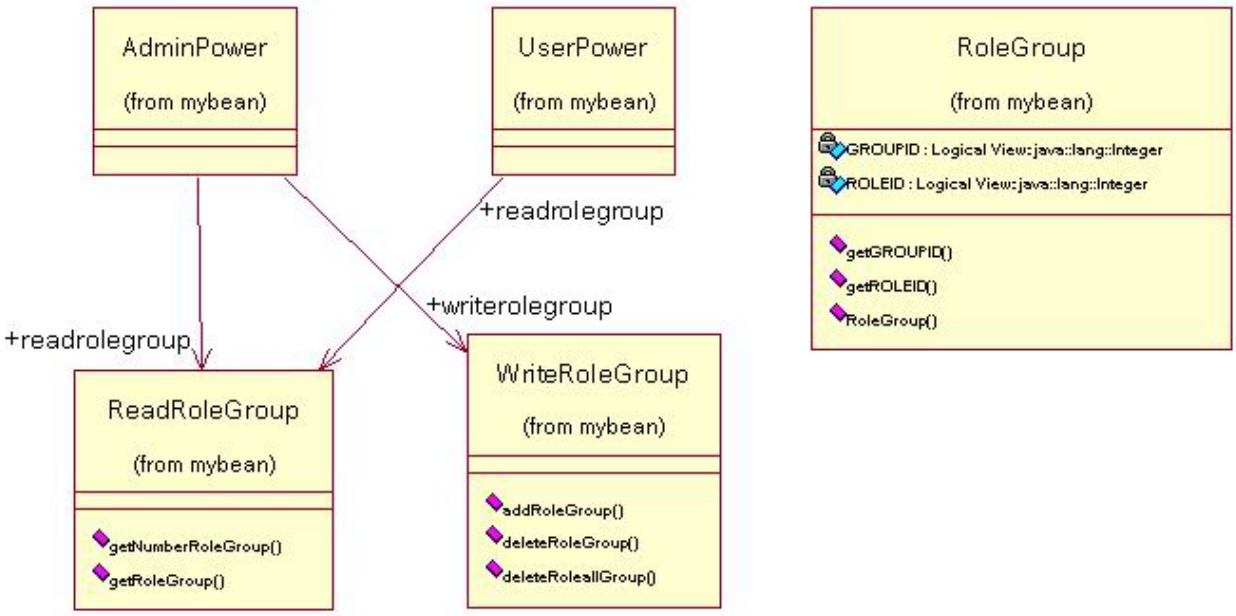
- คลาสที่เกี่ยวกับบทบาทและผู้ใช้ระบบ (RoleUser)



รูปที่ 3-12 ใช้ในข้อมูลเกี่ยวกับบทบาทและผู้ใช้ระบบ (RoleUser)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านและเขียนได้ ที่ผู้ใช้งานสามารถเปลี่ยนแปลงข้อมูลได้เพราะว่า เมื่อผู้ใช้งานลอคอินเข้ามาระบบแล้ว ถ้าหากว่าผู้ใช้มีสิทธิ์ในการเข้าถึงบทบาทผู้ใช้งานที่ผู้ใช้ต้องการเปลี่ยนแปลงค่าผู้ใช้ก็สามารถเข้าไปทำได้
อ่านข้อมูล (Read RoleUser)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลบทบาทและผู้ใช้ระบบ (RoleUser class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์บทบาทและผู้ใช้ระบบ (RoleUser instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write RoleUser)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลบทบาทและผู้ใช้ระบบ(RoleUser class)
กลุ่มผู้ใช้ระบบ (RoleUser)	อ๊อปเจคต์ข้อมูลความสัมพันธ์ระหว่างผู้ใช้ระบบและบทบาทของผู้ใช้งาน

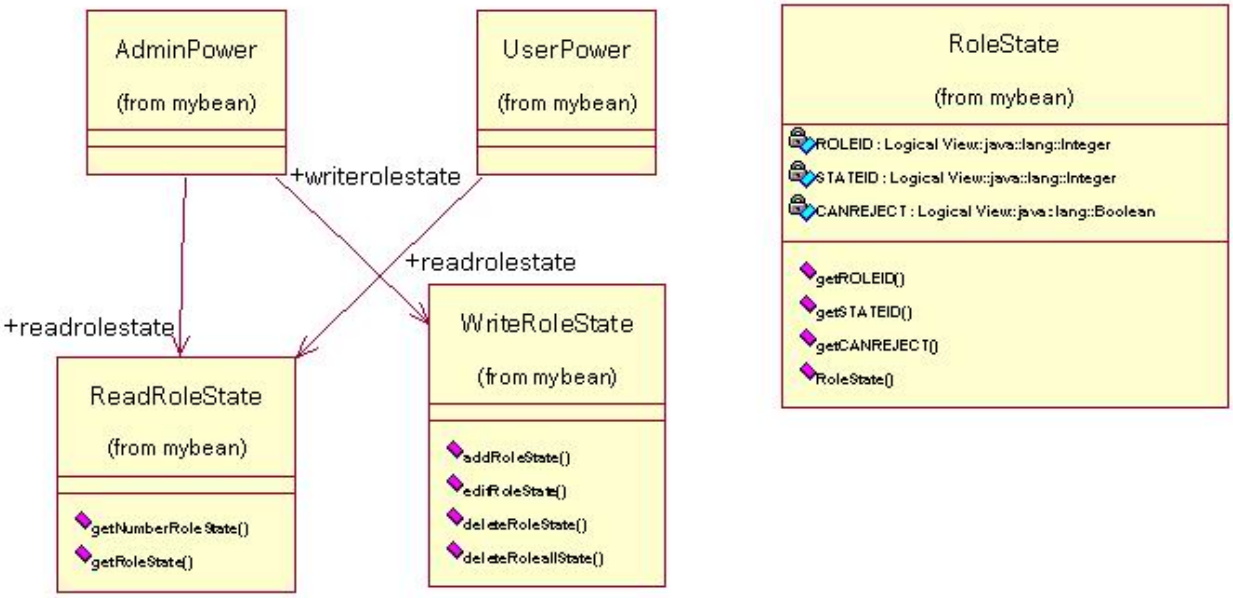
- คลาสที่เกี่ยวกับบทบาทและกลุ่มผู้ใช้ระบบ (RoleGroup)



รูปที่ 3-13 ใช้ในข้อมูลเกี่ยวกับบทบาทและกลุ่มผู้ใช้ระบบ (RoleGroup)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านและเขียนได้ ที่ผู้ใช้งานสามารถเปลี่ยนแปลงข้อมูลได้เพราะว่า เมื่อผู้ใช้งานลอคอินเข้ามาระบบแล้ว ถ้าหากว่าผู้ใช้มีสิทธิ์ในการเข้าถึงบทบาทผู้ใช้งานที่ผู้ใช้ต้องการเปลี่ยนแปลงค่าผู้ใช้ก็สามารถเข้าไปทำได้
อ่านข้อมูล (Read RoleGroup)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลบทบาทและกลุ่มผู้ใช้ระบบ (RoleGroup class) และเพื่อรับค่า (Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์บทบาทและกลุ่มผู้ใช้ระบบ (RoleGroup instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write RoleGroup)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลบทบาทและกลุ่มผู้ใช้ระบบ(RoleGroup class)
กลุ่มผู้ใช้ระบบ (RoleGroup)	อ๊อปเจคต์ข้อมูลของบทบาทผู้ใช้ระบบทั้งหมด ว่ามีสิทธิ์ในการลบงานที่มีอยู่ในระบบได้หรือไม่

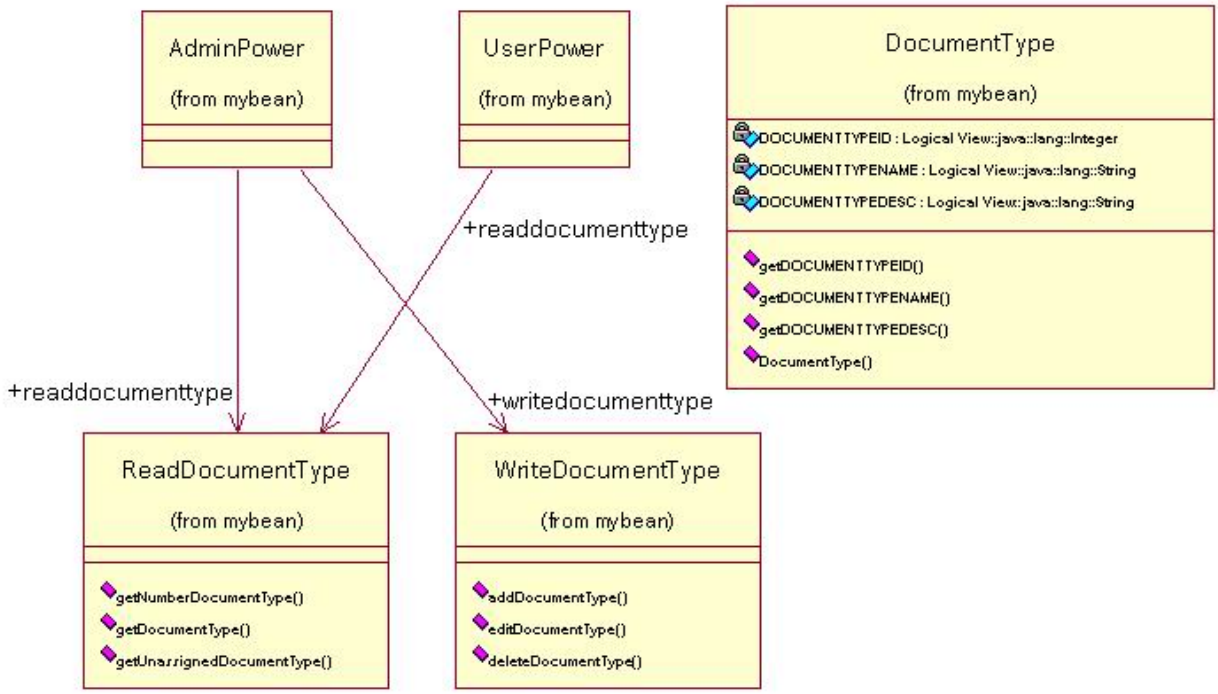
- คลาสที่เกี่ยวกับสถานะบทบาทผู้ใช้ระบบ (RoleState)



รูปที่ 3-14 ใช้ในข้อมูลเกี่ยวกับสถานะบทบาทผู้ใช้ระบบ (RoleState)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านและเขียนได้ ที่ผู้ใช้งานสามารถเปลี่ยนแปลงข้อมูลได้เพราะว่า เมื่อผู้ใช้งานลอคอินเข้ามาระบบแล้ว ถ้าหากว่าผู้ใช้มีสิทธิ์ในการเข้าถึงบทบาทผู้ใช้งานที่ผู้ใช้ต้องการเปลี่ยนแปลงค่าผู้ใช้ก็สามารถเข้าไปทำได้
อ่านข้อมูล (Read RoleState)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลสถานะบทบาทผู้ใช้ระบบ (RoleState class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์สถานะบทบาทผู้ใช้ระบบ (RoleState instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write RoleState)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลสถานะบทบาทผู้ใช้ระบบ(RoleState class)
กลุ่มผู้ใช้ระบบ (RoleState)	อ๊อปเจคต์ข้อมูลความสัมพันธ์ระหว่างกลุ่มผู้ใช้ระบบและบทบาทของผู้ใช้งาน

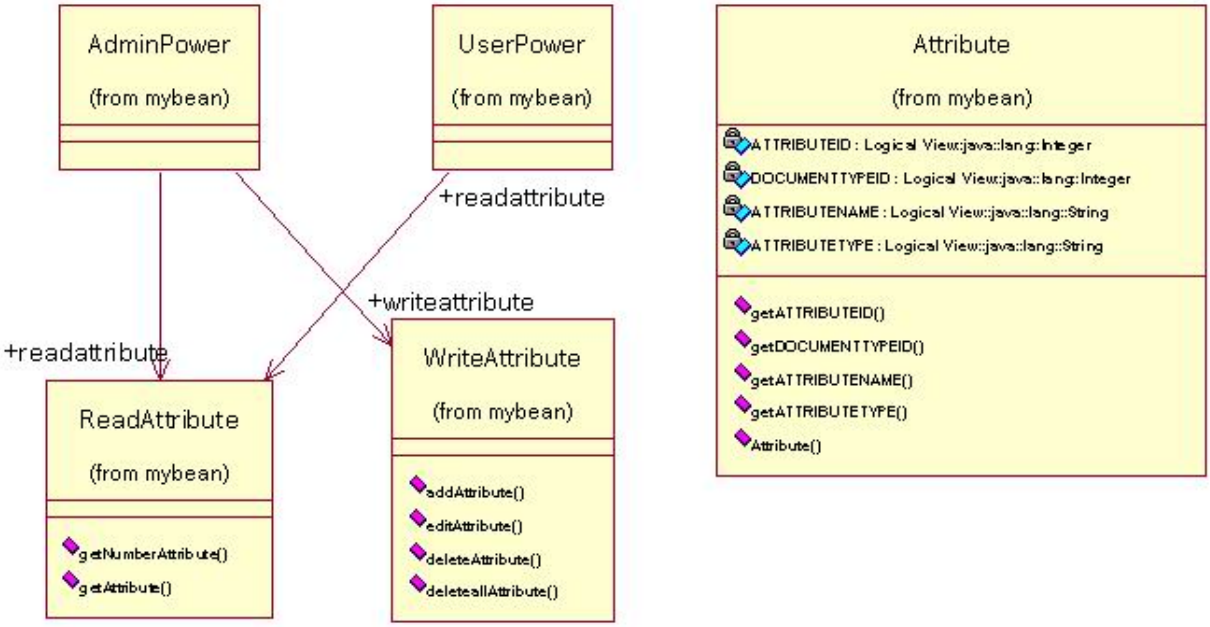
- คลาสที่เกี่ยวกับชนิดของเอกสาร (DocumentType)



รูปที่ 3-15 ใช้ในข้อมูลเกี่ยวกับชนิดของเอกสาร (DocumentType)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านได้
อ่านข้อมูล (Read DocumentType)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลชนิดของเอกสาร (DocumentType class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class) เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์ชนิดของเอกสาร(DocumentType instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (WriteDocumentType)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลชนิดของเอกสาร (DocumentType class)
ชนิดของเอกสาร (DocumentType)	อ๊อปเจคต์ข้อมูลของชนิดของเอกสาร เพื่อใช้ประกอบกับการสร้างอ๊อปเจคต์การไหลของงาน (workflow)

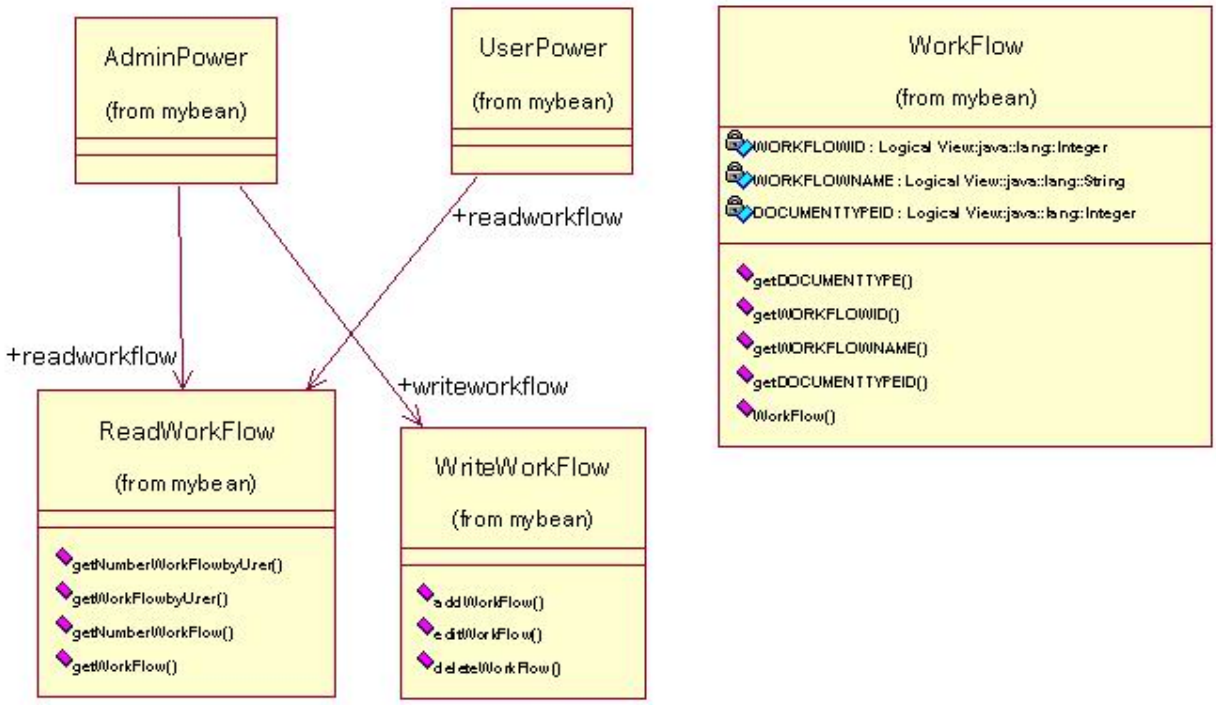
- คลาสที่เกี่ยวกับแอททริบิต (Attribute)



รูปที่ 3-16 ใช้ในข้อมูลเกี่ยวกับแอททริบิต (Attribute)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านได้
อ่านข้อมูล (Read Attribute)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลแอททริบิต (Attribute class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์แอททริบิต (Attribute instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (WriteAttribute)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลแอททริบิต(Attribute class)
แอททริบิต (Attribute)	อ๊อปเจคต์ข้อมูลของแอททริบิตในแต่ละชนิดของเอกสาร (DocumentType) เพื่อใช้ประกอบกับการสร้างอ๊อปเจคต์การไหลของงาน (workflow)

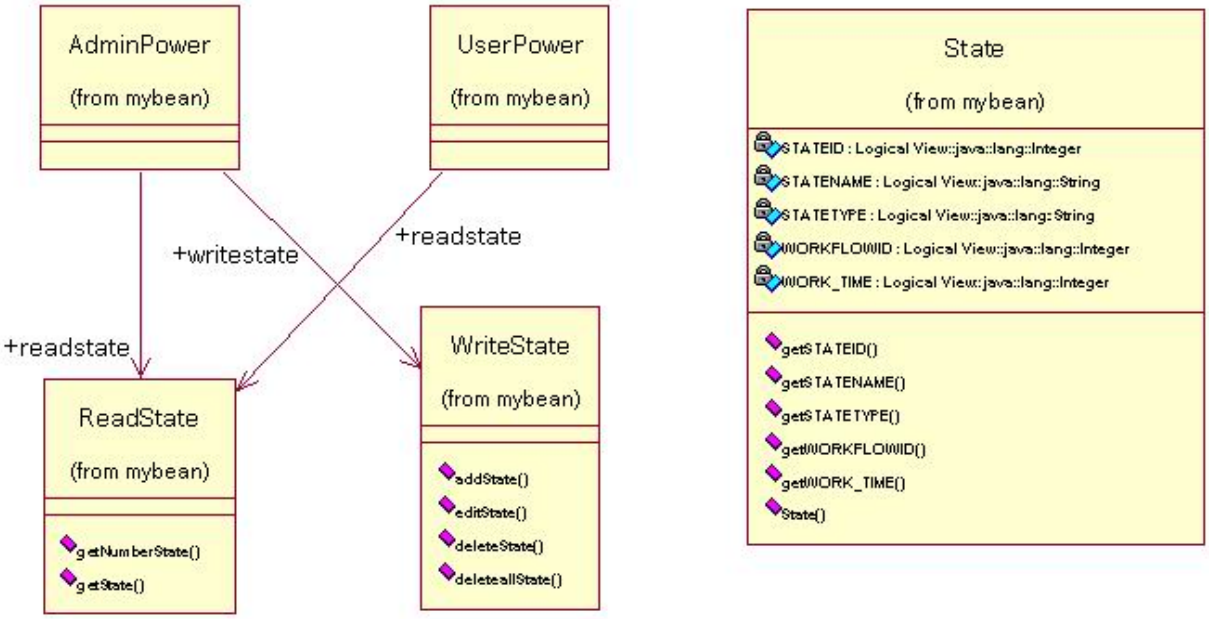
- คลาสที่เกี่ยวกับการไหลของงาน (Workflow)



รูปที่ 3-17 ใช้ในข้อมูลเกี่ยวกับการไหลของงาน (Workflow)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านได้
อ่านข้อมูล (Read Workflow)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลการไหลของงาน (Workflow class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์การไหลของงาน (Workflow instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write Workflow)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลการไหลของงาน (Workflow class)
การไหลของงาน (Workflow)	อ๊อปเจคต์ข้อมูลของการไหลของงาน (Workflow) เพื่อใช้ประกอบกับการสร้างอ๊อปเจคต์งาน (work)

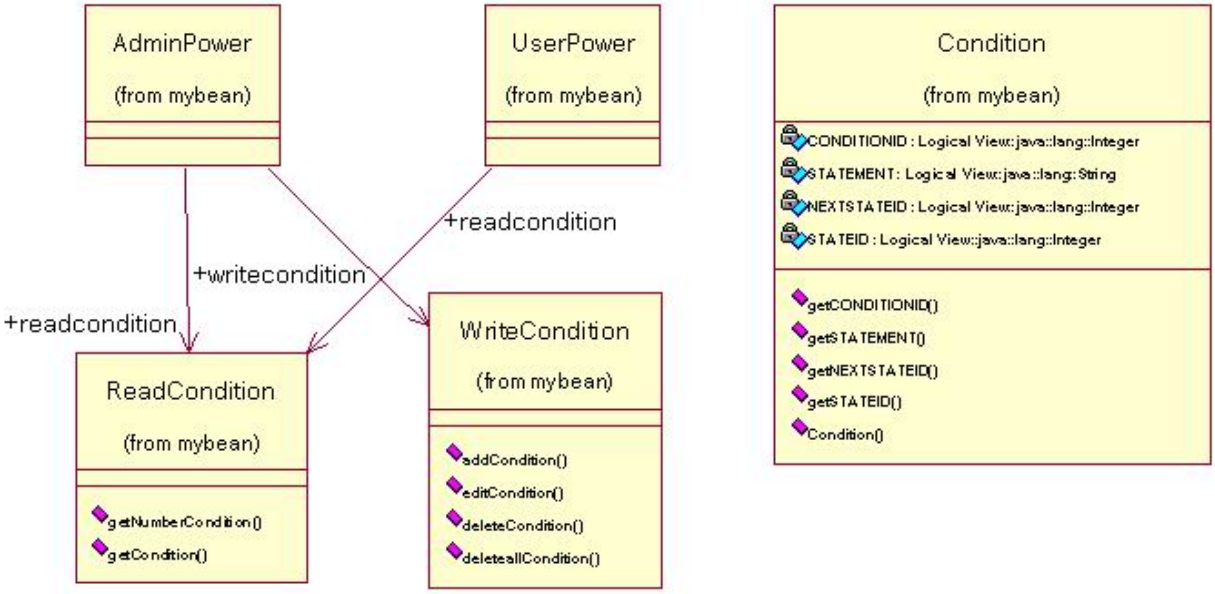
- คลาสที่เกี่ยวกับการสแตท (State)



รูปที่ 3-18 ใช้ในข้อมูลเกี่ยวกับสแตท (State)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านได้
อ่านข้อมูล (Read State)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลสแตท (State class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์สแตท(State instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write State)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลสแตท (State class)
การสแตท (State)	อ๊อปเจคต์ข้อมูลของสแตทในแต่ละการไหลของงาน (Workflow) เพื่อใช้ประกอบกับการสร้างอ๊อปเจคต์งาน (Work)

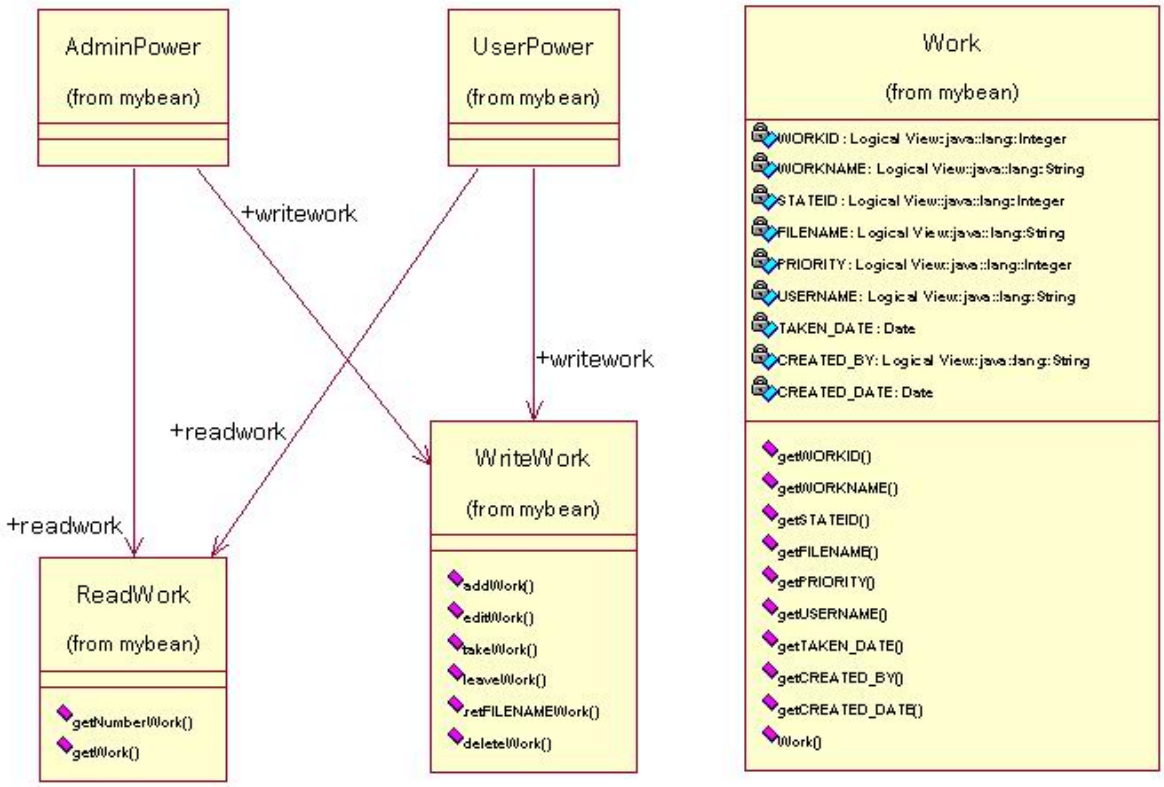
- คลาสที่เกี่ยวกับการเงื่อนไข (Condition)



รูปที่ 3-19 ใช้ในข้อมูลเกี่ยวกับเงื่อนไข (Condition)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่านได้
อ่านข้อมูล (Read Condition)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลเงื่อนไข (Condition class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์เงื่อนไข (Condition instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write Condition)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลเงื่อนไข (Condition class)
การเงื่อนไข (Condition)	อ๊อปเจคต์ข้อมูลของเงื่อนไขในแต่ละสแตท(State) เพื่อใช้ประกอบกับการสร้างอ๊อปเจคต์งาน (Work)

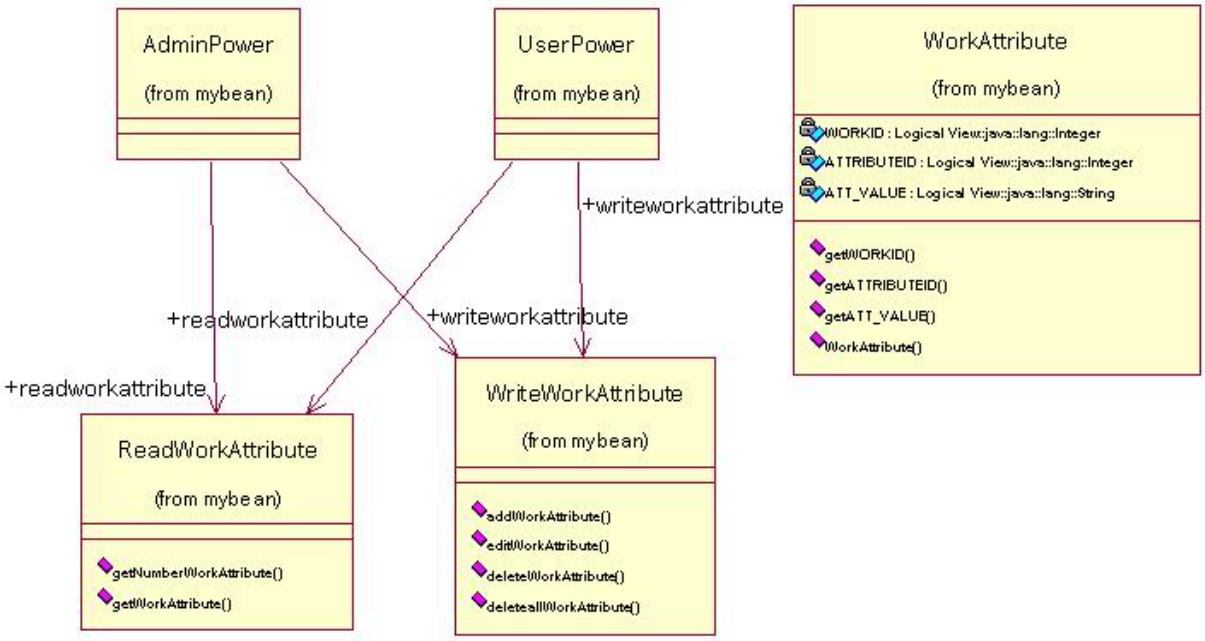
- คลาสที่เกี่ยวกับงาน (Work)



รูปที่ 3-20 ใช้ในข้อมูลเกี่ยวกับงาน (Work)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่าน สร้าง แก้ไข ลบ(ในบางกรณีที่ผู้ใช้นั้นมีสิทธิ์) ข้อมูลงาน
อ่านข้อมูล (Read Work)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลงาน (Work class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์งาน (Work instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write Work)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลงาน (Work class)
งาน (Work)	อ๊อปเจคต์ข้อมูลของงานซึ่งจะมีอ๊อปเจคต์การไหลของงาน (Workflow) อยู่ มีเพื่อเก็บข้อมูลของงานเอาไว้

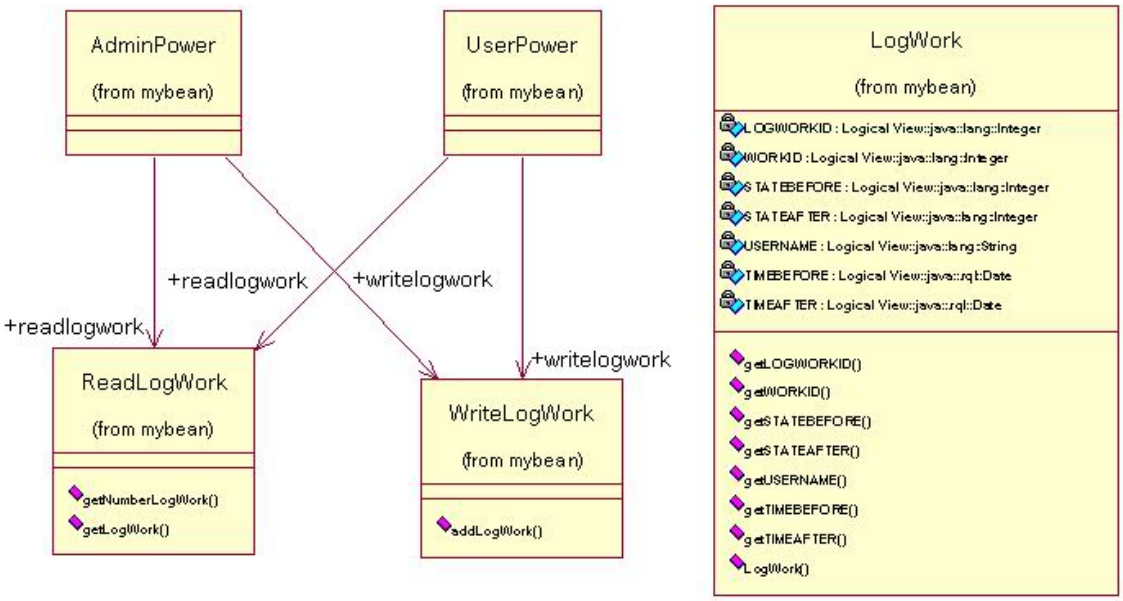
- คลาสที่เกี่ยวกับแอททริบิตงาน (WorkAttribute)



รูปที่ 3-21 ใช้ในข้อมูลเกี่ยวกับการแอททริบิตงาน (WorkAttribute)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่าน สร้าง แก้ไข ลบ(ในบางกรณีที่ผู้ใช้นั้นมีสิทธิ์) ข้อมูลแอททริบิตงาน (WorkAttribute)
อ่านข้อมูล (Read WorkAttribute)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลแอททริบิตงาน (WorkAttribute class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์แอททริบิตงาน (WorkAttribute instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (WriteWorkAttribute)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูล แอททริบิตงาน (WorkAttribute class)
แอททริบิตงาน (WorkAttribute)	อ๊อปเจคต์ข้อมูลของแอททริบิตงานซึ่งจะมีอ๊อปเจคต์การไหลของงาน (Workflow) อยู่ มีเพื่อเก็บข้อมูลของแอททริบิตงานเอาไว้

- คลาสที่เกี่ยวกับเก็บค่าที่เปลี่ยนแปลงของงาน (LogWork)



รูปที่ 3-22 ใช้ในข้อมูลเกี่ยวกับการเปลี่ยนแปลงของงาน (LogWork)

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
ผู้ดูแลระบบ (Admin Power)	อ๊อปเจคต์ชนิดผู้แลระบบสามารถ อ่าน, เพิ่ม, แก้ไข, ลบ ข้อมูล
ผู้ใช้ระบบ (User Power)	อ๊อปเจคต์ชนิดผู้ใช้ สามารถอ่าน สร้าง แก้ไข ลบ(ในบางกรณีที่ผู้ใช้นั้นมีสิทธิ์) ข้อมูลการเปลี่ยนแปลงของงาน (LogWork)
อ่านข้อมูล (Read LogWork)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่ออ่านข้อมูลในฐานข้อมูลการเปลี่ยนแปลงของงาน (LogWork class) และเพื่อรับค่า(Result set) จากดาต้าเบส (Database class)เพื่อแปลงข้อมูลให้อยู่ในรูปของอาเรย์ของอินสเตนส์การเปลี่ยนแปลงของงาน (LogWork instance) เพื่อนำค่านี้ไปแสดงผลที่หน้าเว็บแอปพลิเคชั่น
เขียนข้อมูล (Write LogWork)	ใช้เพื่อเป็นตัวกลางติดกับคลาสดาต้าเบส (Database class)เพื่อเปลี่ยนแปลง ลบ เพิ่มข้อมูลในฐานข้อมูลการเปลี่ยนแปลงของงาน (LogWork)
การเปลี่ยนแปลงของงาน (LogWork)	อ๊อปเจคต์ข้อมูลของการเปลี่ยนแปลงของงาน เก็บค่าความเปลี่ยนแปลงต่างๆในงาน โดยการเปลี่ยนแปลงของงานๆหนึ่งเกิดขึ้นได้หลายครั้งอ๊อปเจคนี้จึงเสมือนการเก็บลอคไฟล์ (log file)

3. Database

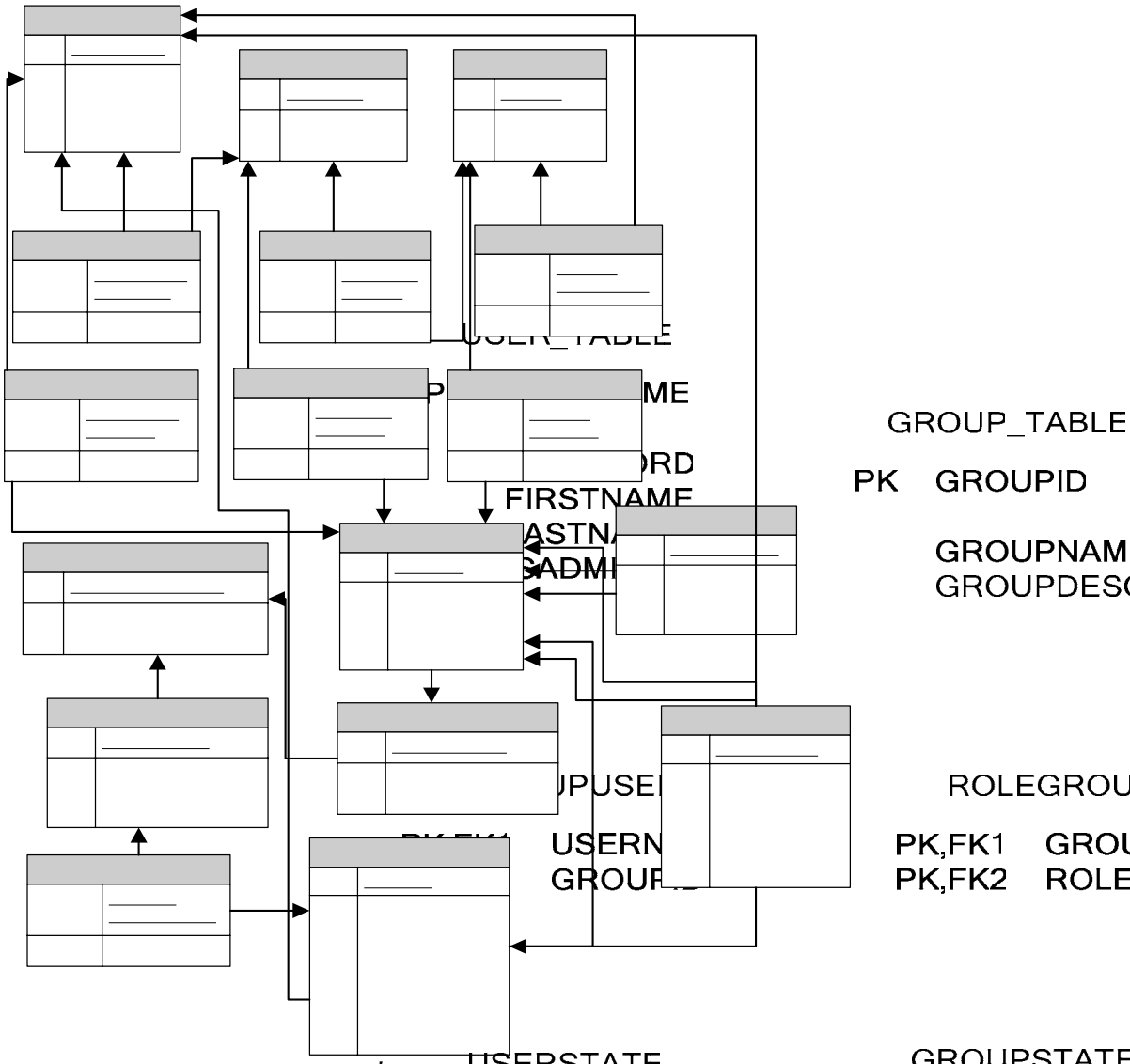
- คลาสดาต้าเบส (Database class)

รูปที่ 3-23 คลาสดาต้าเบส (Database class))

Class name	รายละเอียด หน้าที่และความสัมพันธ์การคลาสอื่นๆ
คลาสดาต้าเบส (Database class)	ติดต่อฐานข้อมูล

3.4 การออกแบบฐานข้อมูลของระบบการไหลของงานเพื่อการจัดการเอกสารในองค์กร

3.4.1 Relational Schema



รูปที่ 3-24 ฐานข้อมูลระบบการไหลของงานเพื่อการจัดการเอกสารในองค์กร (Relational Schema)

ฐานข้อมูลจะประกอบไปด้วยตารางทั้งหมด 17 ตาราง ดังนี้

1. ตาราง USER_TABLE Primary key – [USERNAME]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
USERNAME	สายอักขระ (varchar 20)	ชื่อผู้ใช้งาน
PASSWORD	สายอักขระ (varchar 20)	รหัสผ่านผู้ใช้งาน
FIRSTNAME	สายอักขระ (varchar 50)	ชื่อจริงผู้ใช้งาน
LASTNAME	สายอักขระ(varchar 50)	นามสกุลผู้ใช้งาน
ISADMIN	บิต	แฟลกตรวจว่าเป็นผู้ดูแลระบบหรือไม่

ตารางที่ 3-1 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง USER_TABLE

2. ตารางGROUP_TABLE Primary key – [GROUPID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
GROUPID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำกลุ่ม
GROUPNAME	สายอักขระ (varchar 100)	ชื่อกลุ่ม
GROUPDESC	สายอักขระ (varchar 100)	รายละเอียดกลุ่ม

ตารางที่ 3-2 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง GROUP_TABLE

3. ตารางROLE_TABLE Primary key – [ROLEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
ROLEID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำบทบาท
ROLENAME	สายอักขระ (varchar 100)	ชื่อบทบาท
ROLEDESC	สายอักขระ (varchar 100)	รายละเอียดบทบาท

ตารางที่ 3-3 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง ROLE_TABLE

4. ตารางGROUPUSER Primary key – [USERNAME,GROUPID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
USERNAME	สายอักขระ (varchar 20)	ชื่อผู้ใช้งาน
GROUPID	ตัวเลข (int)	รหัสประจำกลุ่ม

ตารางที่ 3-4 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง GROUPUSER

5. ตาราง ROLEUSER Primary key – [USERNAME,ROLEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
USERNAME	สายอักขระ (varchar 20)	ชื่อผู้ใช้งาน
ROLEID	ตัวเลข (int)	รหัสประจำบทบาท

ตารางที่ 3-5 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง ROLEUSER

6. ตาราง ROLEGROUP Primary key – [GROUPID,ROLEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
GROUPID	ตัวเลข (int)	รหัสประจำกลุ่ม
ROLEID	ตัวเลข (int)	รหัสประจำบทบาท

ตารางที่ 3-6 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง ROLEGROUP

7. ตาราง DOCUMENTTYPE Primary key – [DOCUMENTTYPEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
DOCUMENTTYPEID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำตัวชนิดของเอกสาร
DOCUMENTTYPENAME	สายอักขระ (varchar 100)	ชื่อชนิดของเอกสาร
DOCUMENTTYPEDESC	สายอักขระ (varchar 1000)	รายละเอียดชนิดของเอกสาร

ตารางที่ 3-7 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง DOCUMENTTYPE

8. ตาราง ATTRIBUTE Primary key – [ATTRIBUTEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
ATTRIBUTEID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำตัวแอททริบิวท์
DOCUMENTTYPEID	ตัวเลข (int)	รหัสประจำตัวชนิดของเอกสาร
ATTRIBUTENAME	สายอักขระ (varchar 100)	ชื่อแอททริบิวท์
ATTRIBUTETYPE	สายอักขระ (varchar 10)	ชนิดแอททริบิวท์

ตารางที่ 3-8 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง ATTRIBUTE

9. ตาราง WORKFLOW Primary key – [WORKFLOWID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
WORKFLOWID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำตัว Workflow
DOCUMENTTYPEID	ตัวเลข (int)	รหัสประจำตัวชนิดของเอกสาร
WORKFLOWID NAME	สายอักขระ (varchar 100)	ชื่อ Workflow

ตารางที่ 3-9 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง WORKFLOW

10. ตาราง STATE_TABLE Primary key – [STATEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
STATEID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำตัวสถานะ
STATENAME	สายอักขระ (varchar 100)	ชื่อสถานะ
STATETYPE	สายอักขระ (varchar 10)	ชนิดสถานะมีค่าเป็น(1=START,2=NORMAL, 3=FINAL)
WORKFLOWID	ตัวเลข (int)	รหัสประจำตัว Workflow
WORK_TIME	ตัวเลข (int)	เวลากำหนดในการทำงาน (หน่วยเป็นวัน)

ตารางที่ 3-10 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง STATE_TABLE

11. ตาราง CONDITION Primary key – [CONDITIONID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
CONDITIONID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำตัวเงื่อนไข
STATEMENT	สายอักขระ (varchar 10)	เงื่อนไขของการเปลี่ยนสถานะ
NEXTSTATEID	ตัวเลข (int)	รหัสประจำตัวสถานะต่อไป ใช้เพื่อระบุว่าเมื่อผู้ใช้งานใส่ค่าผ่านเงื่อนไขที่กำหนดไว้แล้วมันจะวิ่งไปที่สถานะใดต่อ
STATEID	ตัวเลข (int)	รหัสประจำตัวสถานะ

ตารางที่ 3-11 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง CONDITION

12. ตาราง USERSTATE Primary key – [USERNAME,STATEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
USERNAME	สายอักขระ (varchar 20)	ชื่อผู้ใช้งาน
STATEID	ตัวเลข (int)	รหัสประจำตัวสถานะ

ตารางที่ 3-12 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง USERSTATE

13. ตาราง GROUPSTATE Primary key – [GROUPID,STATEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
GROUPID	ตัวเลข (int)	รหัสประจำกลุ่ม
STATEID	ตัวเลข (int)	รหัสประจำตัวสถานะ

ตารางที่ 3-13 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง GROUPSTATE

14. ตาราง ROLESTATE Primary key – [ROLEID,STATEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
ROLEID	ตัวเลข (int)	รหัสประจำบทบาท
STATEID	ตัวเลข (int)	รหัสประจำตัวสถานะ

ตารางที่ 3-14 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง ROLESTATE

15. ตาราง WORK_TABLE Primary key – [WORKID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
WORKID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำตัวงาน
WORKNAME	สายอักขระ (varchar 200)	ชื่องาน
STATEID	ตัวเลข (int)	รหัสประจำตัวสถานะ

FILENAME	สายอักขระ (varchar 255)	ชื่อเอกสารที่แนบมา (เมื่อมีการส่งไฟล์เข้ามามันจะถูกเก็บไว้ใน ไดเรกทอรี่ชื่อว่า upload)
PRIORITY	ตัวเลข (int)	ความสำคัญของงานว่าจำเป็นมากขนาดไหน (1-99) โดยที่ 1 มีความจำเป็นมากที่สุด
USERNAME	สายอักขระ (varchar 20)	ชื่อผู้รับงานออกไปทำ
TAKEN_DATE	วันเวลา (datetime)	วันเวลาที่ผู้รับงานออกไปทำ
CREATED_BY	สายอักขระ (varchar 20)	ชื่อผู้ใช้สร้างงาน
CREATED_DATE	วันเวลา (datetime)	วันเวลาที่ผู้ใช้สร้างงาน

ตารางที่ 3-15 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง WORK_TABLE

16. ตาราง WORKATTRIBUTE Primary key – [WORKID,ATTRIBUTEID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
WORKID	ตัวเลข (int)	รหัสประจำตัวงาน
ATTRIBUTEID	ตัวเลข (int)	รหัสประจำตัวแอททริบิวท์
ATT_VALUE	ตัวเลข (int)	เก็บค่าของแอททริบิวท์ในแต่งาน เพื่อนำมาประกอปการตัดสินใจในการเปลี่ยนเงื่อนไข (Condition)

ตารางที่ 3-16 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง WORKATTRIBUTE

17. ตาราง LOGWORK Primary key – [LOGWORKID]

ชื่อแอททริบิวท์	ชนิด	คำอธิบาย
LOGWORKID	ตัวเลขวิ่งอัตโนมัติ (int identity)	รหัสประจำตัวการเก็บข้อมูลในการทำงาน
WORKID	ตัวเลข (int)	รหัสประจำตัวงาน
STATEBEFORE	ตัวเลข (int)	รหัสประจำตัวสถานะก่อนเปลี่ยน
STATEBEFORE	ตัวเลข (int)	รหัสประจำตัวสถานะหลังเปลี่ยน
USERNAME	สายอักขระ (varchar 20)	ชื่อผู้รับงานออกไปทำ
TIME_BEFORE	วันเวลา (datetime)	วันเวลาที่ผู้รับงานออกไปทำ
TIME_AFTER	วันเวลา (datetime)	วันเวลาที่ผู้รับงานออกไปทำเสร็จ แล้วส่งไปยัง สถานะถัดไป

ตารางที่ 3-17 ตารางแสดงชื่อแอททริบิวท์ ชนิด และคำอธิบายของตาราง LOGWORK