

## โครงการพัฒนาระบบทะเบียนซอฟต์แวร์

สมพล ภาวดีรุจิรา\*

ธนาคารกรุงเทพ จำกัด (มหาชน)

\*Correspondence: sompol.par@bbl.co.th

doi: 10.14456/jisb.2017.1

### บทคัดย่อ

บทความนี้นำเสนอการพัฒนาระบบทะเบียนซอฟต์แวร์ในชื่อ “AssetMeter” เพื่อให้องค์กรที่สนใจนำไปใช้จัดการสินทรัพย์ทั้งฮาร์ดแวร์และซอฟต์แวร์ให้มีการเก็บรวบรวมข้อมูลของสินทรัพย์เหล่านั้นไว้อย่างเป็นระบบเพื่อประกอบการตัดสินใจ และควบคุมการใช้งานให้ถูกต้องตามกฎหมายหรือนโยบายขององค์กร โดยระบบ AssetMeter ที่พัฒนามีสถาปัตยกรรมแบบเว็บแอปพลิเคชัน (Web-Based Application) ที่นำมาใช้งานภายในองค์กรในลักษณะอินเทอร์เน็ต โดยระบบนี้จะครอบคลุมการทำงานตั้งแต่ผู้บริหารจัดซื้อฮาร์ดแวร์และลิขสิทธิ์ซอฟต์แวร์ จัดการลิขสิทธิ์และสัญญาต่างๆ ให้เพียงพอต่อการใช้งาน ส่งมอบสินทรัพย์ที่ซื้อให้ผู้ดูแลระบบ ติดตั้งระบบและติดต่อสื่อสารกับผู้ที่เกี่ยวข้องให้เข้ามามีส่วนร่วมในการจัดการดูแลรักษาสินทรัพย์ จนกระทั่งสินทรัพย์เหล่านั้นถูกยกเลิกการใช้งาน

ในการพัฒนาระบบ ผู้พัฒนาใช้แนวคิดการวิเคราะห์และออกแบบระบบเชิงวัตถุ (Object-Oriented analysis and design) โดยใช้ UML 2.3 (Unified Modeling Language) ในการสร้างตัวแบบระบบ พัฒนาโปรแกรมโดยใช้ภาษา C# ภายใต้ ASP.NET 4.5 Framework และใช้โปรแกรม Visual Studio Community 2015 เป็นเครื่องมือในการพัฒนาระบบร่วมกับโปรแกรมจัดการฐานข้อมูล Microsoft SQL Server 2012

**คำสำคัญ:** ทะเบียนสินทรัพย์ ซอฟต์แวร์ สัญญา การวิเคราะห์และออกแบบระบบเชิงวัตถุ

## Developing a Software Asset Management System

**Sompol Paradeerujira\***

Bangkok Bank Public Company Limited

\*Correspondence: sompol.par@bbl.co.th

doi: 10.14456/jisb.2017.1

### Abstract

AssetMeter, a development project, consisted of software asset management (SAM). It aimed to help interested organizations to manage hardware and software assets for data collection, enhancing decision support and controlling software usage according to rights and policies. Its system architecture was a web-based application implemented within the enterprise, similar to intranet software. Development scope covered purchasing hardware and software licenses. Management of copyright and contracts was sufficient to operate. Purchased assets were delivered to the administrator. Installation and open communication with participants in managed assets occurred until assets were disposed of.

Object-oriented analysis and design was applied to create the system model with Unified Modeling Language (UML 2.3). This system was developed by C# programming language. ASP.NET 4.5 Framework and Visual Studio Community 2015 were integrated with Microsoft SQL Server 2012 as development tools.

**Keywords:** Asset Management, Software, License, Object-oriented analysis and design

## บทนำ

ในปัจจุบัน องค์กรธุรกิจต่างๆ มีการนำเทคโนโลยีสารสนเทศทั้งฮาร์ดแวร์และซอฟต์แวร์มาใช้งานเพิ่มขึ้นอย่างต่อเนื่อง จึงต้องมีการควบคุมให้มีการใช้งานอย่างถูกต้องและปลอดภัย เช่น เครื่องแม่ข่ายที่มีประสิทธิภาพสูง มีการใช้ลิขสิทธิ์ของโปรแกรม (License) มากขึ้นตามการใช้งานระบบสารสนเทศที่มากขึ้น ซึ่งมีผลต่อค่าใช้จ่ายขององค์กร เช่น ค่าปรับจากการใช้งานเกินสัญญา เป็นต้น และยังมีผลต่อการตรวจสอบโดยบริษัทผู้ให้บริการโปรแกรม เช่น Oracle, Microsoft เป็นต้น ถ้าการใช้งานไม่เป็นไปตามสัญญา ดังนั้นการรวบรวมข้อมูลฮาร์ดแวร์ ข้อมูลซอฟต์แวร์ และข้อมูลสัญญาการดูแลบำรุงรักษา (Maintenance Agreement) สิทธิประโยชน์เหล่านี้จากฐานข้อมูลที่อยู่กระจายตามเครื่องแม่ข่ายของหน่วยงานต่างๆ มาไว้ที่เดียวกัน เพื่อนำมาใช้ในการวางแผนการลงทุนจึงเป็นเรื่องจำเป็น ระบบ AssetMeter จึงถูกพัฒนาขึ้นเพื่อเป็นระบบงาน (Application) ที่เป็นศูนย์กลางที่รวบรวมข้อมูลเทคโนโลยีสารสนเทศที่จำเป็นต่อการวางแผนการลงทุนและสนับสนุนการตัดสินใจให้กับองค์กรต่างๆ ต่อไป

## การวิเคราะห์ความต้องการของระบบ

ระบบ AssetMeter เริ่มต้นจากการเก็บรวบรวมข้อมูลความต้องการโดยใช้วิธีสัมภาษณ์จากกลุ่มตัวอย่างที่เป็นผู้มีส่วนได้ส่วนเสียทั้งหมด ซึ่งประกอบด้วยผู้มีส่วนเกี่ยวข้องต่างๆ ดังนี้

**ผู้บริหารโครงการ (Project Manager)** ทำหน้าที่เรียกดูความต้องการลิขสิทธิ์ของโปรแกรม (License) ของแต่ละแผนก จัดสรรลิขสิทธิ์ของโปรแกรมลงอุปกรณ์ที่ต้องใช้ซอฟต์แวร์ โดยผู้มีส่วนเกี่ยวข้องนี้อาจได้แก่ ผู้บริหารฝ่ายต่างๆ ขององค์กรลูกค้าที่ใช้ AssetMeter เช่น ผู้บริหารฝ่ายขาย เป็นต้น โดยผู้บริหารโครงการต้องการระบบงานเพื่อจัดการลิขสิทธิ์ของโปรแกรม และมีปัญหาเวลางานขยายตัวแล้วต้องใช้ลิขสิทธิ์ของโปรแกรมเพิ่มหรือไม่ยอมหยุดใช้งานเมื่อหมดสัญญาแล้ว ดังนั้นระบบจึงต้องมีฟังก์ชันการร้องขอลิขสิทธิ์ของโปรแกรม จัดสรรลิขสิทธิ์ของโปรแกรม และแจ้งเตือนสัญญา เป็นต้น

**ผู้บริหารระดับสูง (Top Manager)** ทำหน้าที่พิจารณาคำร้องขอลิขสิทธิ์ของโปรแกรม เปรียบเทียบกับผู้ให้บริการภายนอก และวัดผลการใช้งานสิทธิประโยชน์ผ่านการเรียกดูรายงานต่างๆ โดยผู้มีส่วนเกี่ยวข้องนี้อาจได้แก่ ผู้ก่อตั้ง (Founder) หรือผู้บริหารฝ่ายทะเบียนทรัพย์สินขององค์กรลูกค้าที่ใช้ AssetMeter เป็นต้น โดยผู้บริหารระดับสูงต้องการข้อมูลประกอบการวางแผนการลงทุนในอนาคต ระบบจึงต้องมีฟังก์ชันการพิจารณาคำร้องขอและเรียกดูรายงานประกอบได้ เป็นต้น

**ผู้ดูแลสัญญา (Contract Admin)** ทำหน้าที่บริหารจัดการข้อมูลสัญญาและซอฟต์แวร์ ตรวจสอบการใช้งานให้เป็นไปตามนโยบาย กฎหมาย และกฎระเบียบขององค์กร โดยผู้มีส่วนเกี่ยวข้องนี้อาจได้แก่ เจ้าหน้าที่บริหารส่วนงานกลางที่รับผิดชอบเกี่ยวกับการดูแลลิขสิทธิ์ของโปรแกรมหรือซอฟต์แวร์ขององค์กรลูกค้าที่ใช้ AssetMeter เป็นต้น โดยผู้ดูแลสัญญามีปัญหาเรื่องการใช้อิทธิของโปรแกรมเกินจากที่ผู้ผลิตหรือผู้จัดจำหน่าย (vendor) กำหนด ดังนั้นระบบจึงต้องมีฟังก์ชันการบันทึกการรับลิขสิทธิ์ของโปรแกรม การตรวจสอบซอฟต์แวร์ เป็นต้น

**ผู้ดูแลระบบ (System Admin)** ทำหน้าที่ติดตั้งอุปกรณ์ ซอฟต์แวร์ หรือยกเลิกการใช้อุปกรณ์หรือซอฟต์แวร์ตามที่ได้รับมอบหมาย โดยผู้มีส่วนเกี่ยวข้องนี้อาจได้แก่ เจ้าหน้าที่วิศวกรรมระบบขององค์กรลูกค้าที่ใช้ AssetMeter เป็นต้น โดยผู้ดูแลระบบต้องการระบบงานเพื่อจัดการอุปกรณ์การใช้งาน ดังนั้นระบบจึงต้องมีฟังก์ชันการจัดการหรือยกเลิกอุปกรณ์ การบันทึกการติดตั้งหรือถอนซอฟต์แวร์ เป็นต้น

เมื่อนำข้อมูลที่เก็บรวบรวมได้จากกลุ่มตัวอย่างที่เป็นผู้มีส่วนได้ส่วนเสียเหล่านั้นมาวิเคราะห์โดยทำตัวแบบระบบ (System Models) ด้วย UML 2.3 จึงกำหนดขอบเขตของระบบ AssetMeter ที่พัฒนาให้ครอบคลุมการทำงานตั้งแต่ผู้บริหารจัดซื้อฮาร์ดแวร์และลิขสิทธิ์ซอฟต์แวร์ จัดการลิขสิทธิ์และสัญญาต่างๆ ให้เพียงพอต่อการใช้งาน ส่งมอบสิทธิประโยชน์ที่ซื้อให้ผู้ดูแลระบบติดตั้งระบบและติดต่อสื่อสารกับผู้ที่เกี่ยวข้องให้เข้ามามีส่วนร่วมในการจัดการดูแลรักษา สิทธิประโยชน์ จนกระทั่งถึงสิทธิประโยชน์ถูกยกเลิกการใช้งาน โดยภายในระบบประกอบด้วยฟังก์ชันงานต่างๆ ดังตารางที่ 1

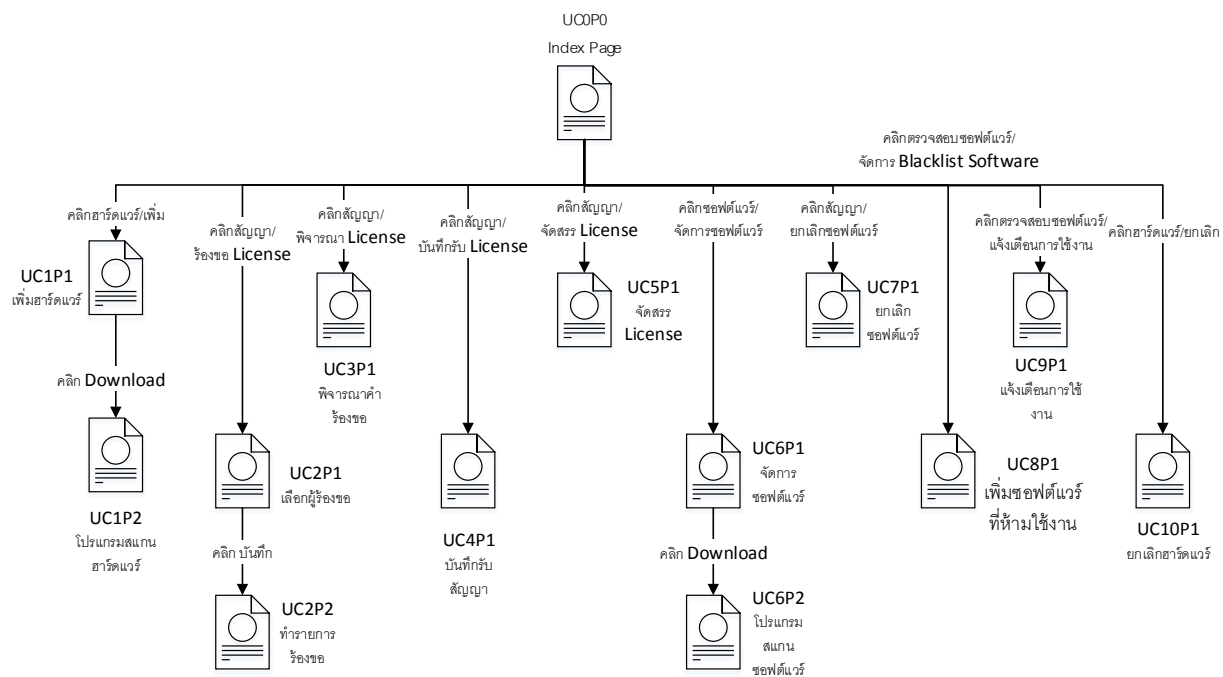
ตารางที่ 1 แสดงฟังก์ชันทั้งหมดของระบบ

ฟังก์ชัน	คำอธิบาย
จัดการอุปกรณ์	ผู้ดูแลระบบที่ได้รับมอบหมายให้ดูแลอุปกรณ์ สามารถเพิ่มหรือแก้ไขข้อมูล รายละเอียดเกี่ยวกับอุปกรณ์ ได้แก่ รหัสอุปกรณ์ รหัสหน่วยงานเจ้าของ ชื่อ อุปกรณ์ จำนวน Core จำนวนหน่วยความจำ(Memory) ชื่อรุ่นหน่วยประมวลผล และสถานะอุปกรณ์ที่กำลังจะติดตั้งระบบรองรับโครงการใหม่ๆ ได้
ยกเลิกอุปกรณ์	ผู้ดูแลระบบสามารถลบข้อมูลอุปกรณ์ที่ถูกยกเลิกการใช้งานออกจากระบบ และย้ายอุปกรณ์จากโครงการหนึ่งไปใช้ในโครงการอื่นแทนได้
ร้องขอลิขสิทธิ์ของโปรแกรม ( License )	ผู้บริหารโครงการของแต่ละแผนกสามารถแสดงความจำนงขอเพิ่มจำนวนการใช้งานซอฟต์แวร์ เพื่อให้ผู้ดูแลสัญญาเก็บรวบรวมข้อมูลไว้เพื่อเป็นข้อมูลในการจัดซื้อได้
พิจารณาคำร้องขอลิขสิทธิ์ของโปรแกรม	เมื่อมีคำร้องขอใช้งานซอฟต์แวร์ ผู้บริหารระดับสูงสามารถพิจารณาอนุมัติ/ปฏิเสธแต่ละคำร้องขอได้ โดยผลการอนุมัติ/ปฏิเสธจะถูกบันทึกในระบบ เพื่อสรุปเป็นลิขสิทธิ์ของโปรแกรม (License) ที่องค์กรต้องการสั่งซื้อ
บันทึกลิขสิทธิ์ของโปรแกรม	เมื่อผู้บริหารและเจ้าของซอฟต์แวร์ทำการลงนามสัญญาการใช้งานซอฟต์แวร์เรียบร้อยแล้ว ผู้ดูแลสัญญามีหน้าที่บันทึกรายละเอียดสัญญาเข้าระบบเพื่อให้ซอฟต์แวร์พร้อมใช้งานในองค์กรอย่างถูกต้อง
จัดสรรลิขสิทธิ์ของโปรแกรม	ผู้บริหารโครงการของแต่ละแผนกสามารถจัดสรรจำนวนสิทธิ์การใช้งานซอฟต์แวร์ให้กับอุปกรณ์ต่างๆ ในแผนกตามที่ระบุในคำร้องขอลิขสิทธิ์ของโปรแกรมของแผนกที่ได้รับอนุมัติได้
ติดตั้งซอฟต์แวร์	เมื่อดำเนินการติดตั้งซอฟต์แวร์ตามรุ่น หรือความสามารถหลักที่โครงการกำหนดเรียบร้อยแล้ว ผู้ดูแลระบบสามารถตรวจสอบข้อมูลการติดตั้งซอฟต์แวร์ได้ว่าตรงกับที่จัดสรรลิขสิทธิ์ของโปรแกรมไว้หรือไม่โดยใช้โปรแกรมสแกนจากเครื่องคอมพิวเตอร์ ถ้าผลการสแกนพบว่าตรงกับที่จัดสรร จะเปลี่ยนสถานะเป็นอนุมัติ (Approved) และจากนั้นระบบจะเริ่มนับการใช้เวลาซอฟต์แวร์ แต่สำหรับซอฟต์แวร์อื่นที่ไม่ตรงกับการจัดสรร จะเปลี่ยนสถานะเป็นติดตั้ง (Installed)
ยกเลิกซอฟต์แวร์	เมื่อต้องการยกเลิกการใช้งานซอฟต์แวร์หรือย้ายซอฟต์แวร์ให้ไปทำงานบนอุปกรณ์อื่น ผู้บริหารโครงการของแต่ละแผนกสามารถยกเลิกการจัดสรรลิขสิทธิ์ของโปรแกรมได้ ซึ่งจะส่งผลให้ผู้ดูแลระบบทำการถอนการติดตั้งซอฟต์แวร์พร้อมคืนจำนวนใบอนุญาตกลับสู่ระบบเพื่อให้สามารถนำไปจัดสรรให้กับอุปกรณ์อื่นๆ ได้
ตรวจสอบซอฟต์แวร์	ผู้ดูแลสัญญาสามารถตรวจสอบรายการซอฟต์แวร์ที่มีการใช้งานไม่เป็นไปตามนโยบายขององค์กรที่ถูกติดตั้งในอุปกรณ์ทั้งหมดได้โดยการกำหนดซอฟต์แวร์ที่ห้ามใช้งานตามนโยบายขององค์กร ระบบจะค้นหาชื่อซอฟต์แวร์นั้นในทุกๆ อุปกรณ์ขององค์กรให้โดยอัตโนมัติ
แจ้งเตือนสัญญา	ระบบแจ้งเตือนให้ผู้ดูแลสัญญาทราบว่า มีซอฟต์แวร์ที่ใกล้หมดอายุ (3 เดือน) แล้วหรือหมดอายุ ผู้ดูแลสัญญาสามารถส่งค่าแจ้งเตือนให้กับผู้ดูแลโครงการที่เป็นผู้ใช้ซอฟต์แวร์นั้นทราบได้ โดยระบบจะบันทึกข้อมูลการแจ้งเตือน ได้แก่ วันที่แจ้งเตือนล่าสุด และจำนวนการแจ้งเตือนไว้ในระบบ

## 1. การออกแบบระบบ

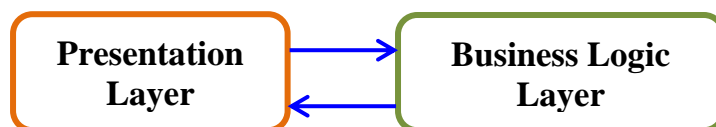
ระบบ AssetMeter ถูกออกแบบโดยใช้สถาปัตยกรรมระบบรับ/ให้บริการ (Client-server) ที่ทำงานแบบเว็บแอปพลิเคชัน (Web-Based Application) โดยการติดตั้งอาจอยู่ในลักษณะ 3 ชั้น (tiers) กล่าวคือ มีการแบ่งโปรแกรมของระบบออกเป็น 3 ส่วน ได้แก่ ส่วนการแสดงผล ส่วนการประมวลผล และส่วนฐานข้อมูล และแต่ละส่วนถูกติดตั้งในอุปกรณ์ที่แยกออกจากกัน โดยในส่วนของไคลเอ็นต์ (Client) จะทำหน้าที่ติดต่อกับผู้ใช้โดยรับข้อมูลจากผู้ใช้ส่งไปที่แอปพลิเคชันเซิร์ฟเวอร์ (Application Server) และรับผลลัพธ์จากการประมวลผลของแอปพลิเคชันเซิร์ฟเวอร์มาแสดงผลในรูปแบบที่ผู้ใช้ต้องการ โดยในระหว่างการประมวลผลของ แอปพลิเคชันเซิร์ฟเวอร์จะติดต่อกับดาต้าเบสเซิร์ฟเวอร์ (Database Server) เพื่อเรียกข้อมูลจากฐานข้อมูลมาใช้ในการประมวลผลและนำข้อมูลที่ประมวลผลได้ส่งให้ดาต้าเบสเซิร์ฟเวอร์นำไปจัดเก็บลงในฐานข้อมูล ซึ่งในการทำงานของระบบ AssetMeter ที่ติดตั้งบนแอปพลิเคชันเซิร์ฟเวอร์อยู่ภายใต้ .Net Framework version 4.5

ในส่วนแสดงผลของระบบ AssetMeter ถูกออกแบบให้ประกอบด้วยหน้าจอต่างๆ ที่มีความสัมพันธ์กัน ดังแสดงในแผนภาพ Web Flow (ภาพที่ 1)



ภาพที่ 1 แผนภาพ Web flow ของระบบ

ในการพัฒนาโปรแกรมของระบบ AssetMeter ได้ประยุกต์ใช้แนวทางการวิเคราะห์และออกแบบระบบเชิงอ็อบเจกต์ (Object-oriented analysis and design) โดยแบ่งการทำงานของโปรแกรมแต่ละโปรแกรมออกเป็น 2 ส่วน (ดังแสดงในภาพที่ 2) ได้แก่

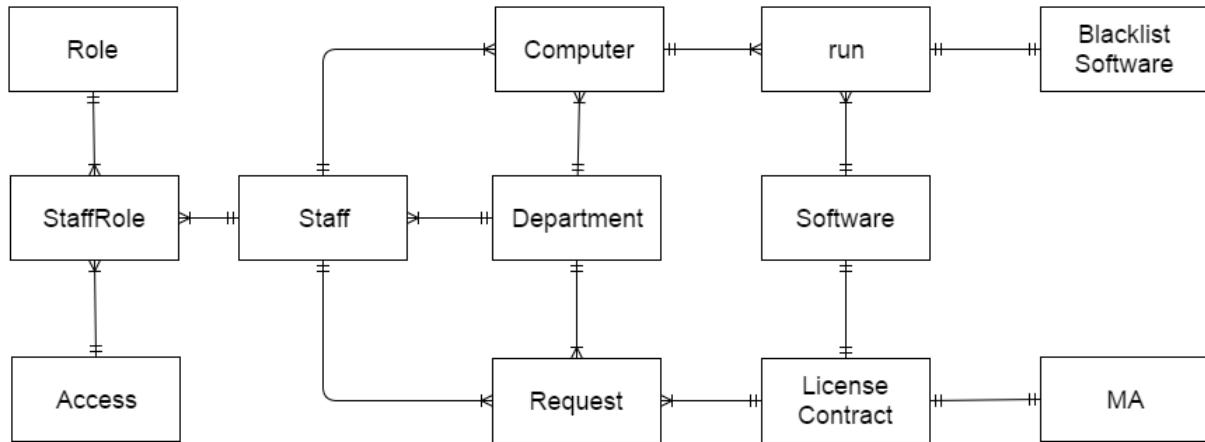


ภาพที่ 2 แนวคิดการออกแบบโปรแกรม

1. ส่วนแสดงผล (Presentation Layer) อยู่ในไฟล์นามสกุล .aspx ที่ทำหน้าที่จัดการหน้าจอ และไฟล์นามสกุล .aspx.cs ที่ทำหน้าที่รองรับการประมวลผลก่อนแสดงผลขึ้นหน้าจอ

2. ส่วนประมวลผล (Business Logic Layer) อยู่ในไฟล์นามสกุล .cs ที่มีการสร้างเป็น class และ method ต่างๆ ตามแนวทางของการวิเคราะห์และออกแบบระบบเชิงอ็อบเจ็ค โดยภายในแต่ละ class จะมีการติดต่อฐานข้อมูล

สำหรับส่วนฐานข้อมูลของระบบ AssetMeter มีโครงสร้าง ดังแสดงใน Entity Diagram ภาพที่ 2 โดยแต่ละรายการ Entity ใช้สำหรับเก็บข้อมูลดังตารางที่ 2



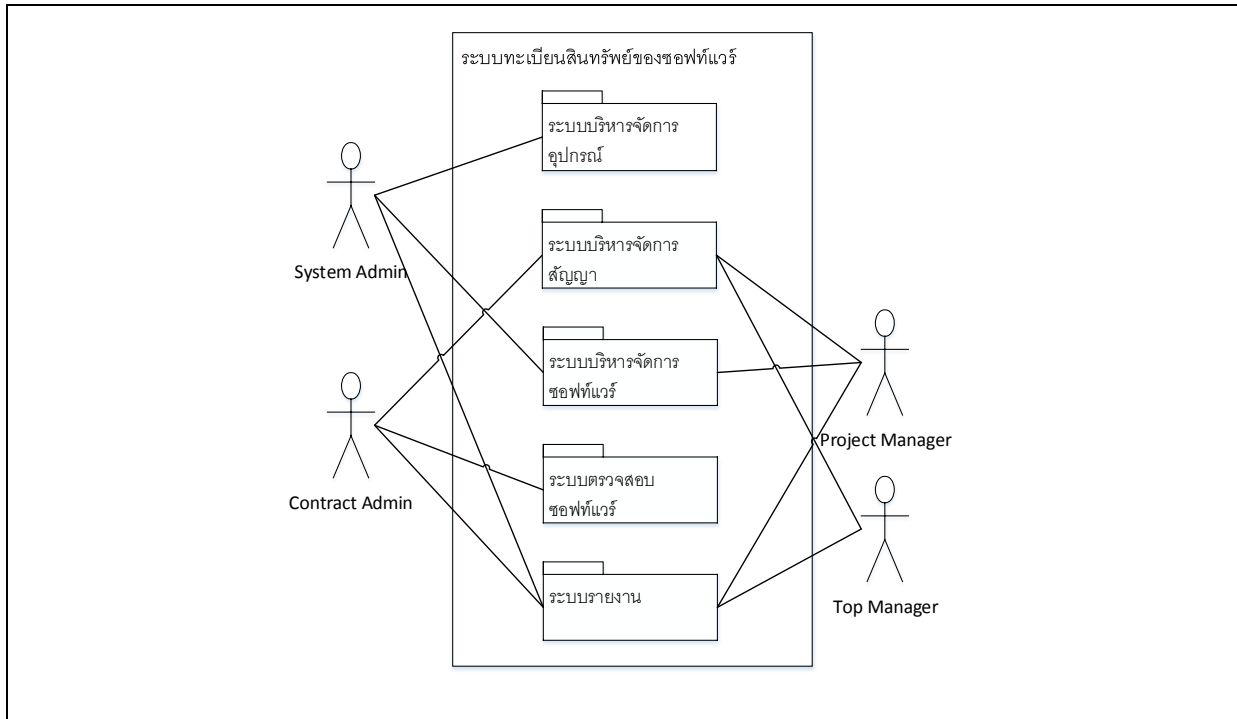
ภาพที่ 2 Entity Diagram แสดงความสัมพันธ์ของข้อมูลในระบบ AssetMeter

ตารางที่ 2 คำอธิบายการเก็บข้อมูลใน Entity

ลำดับ	ชื่อตาราง	ประเภทของข้อมูลที่เก็บในตาราง	คำอธิบาย
1	Access	Master	เก็บข้อมูลการเข้าถึงหน้าจอ
2	Blacklist Software	Master	เก็บข้อมูลซอฟต์แวร์ที่ไม่ได้รับอนุญาต
3	Computer	Master	เก็บข้อมูลอุปกรณ์
4	Department	Master	เก็บข้อมูลแผนก
5	License Contract	Transaction	เก็บข้อมูลสัญญาลิขสิทธิ์ของโปรแกรม (License)
6	MA	Transaction	เก็บข้อมูลการบำรุงรักษา ( Maintenance หรือ MA )
7	Request	Transaction	เก็บข้อมูลคำร้องขอลิขสิทธิ์ของโปรแกรม
8	Role	Master	เก็บข้อมูลบทบาทผู้ใช้งาน
9	Running Software	Transaction	เก็บข้อมูลซอฟต์แวร์ที่ติดตั้งบนอุปกรณ์
10	Software	Transaction	เก็บข้อมูลซอฟต์แวร์
11	Staff	Master	เก็บข้อมูลพนักงาน
12	StaffRole	Master	เก็บข้อมูลบทบาทของพนักงาน

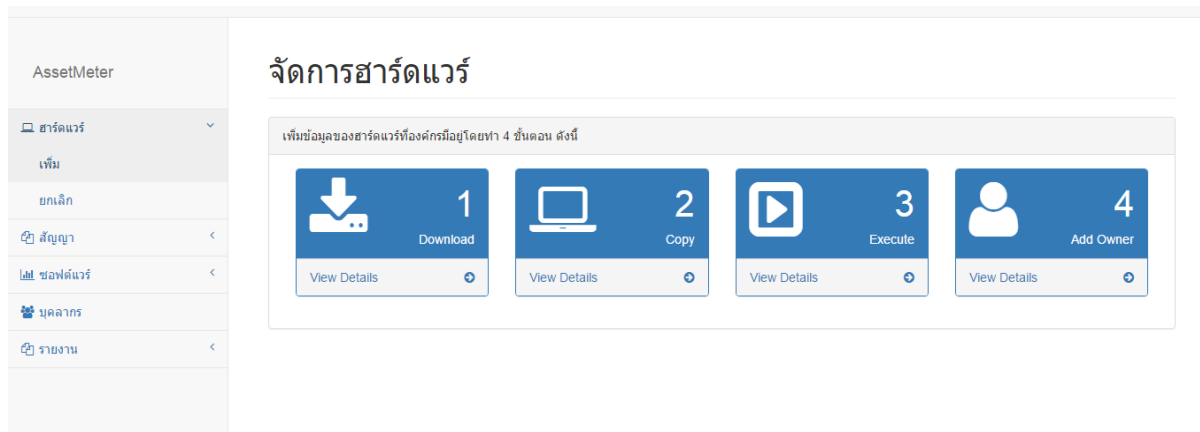
## 2. ระบบ AssetMeter และการใช้งาน

ภายในระบบ AssetMeter ประกอบด้วย 5 ระบบย่อยๆ โดยแต่ละระบบย่อยมีผู้ที่เกี่ยวข้อง ดังแสดงในภาพที่ 3 และมีขอบเขตการทำงานดังนี้



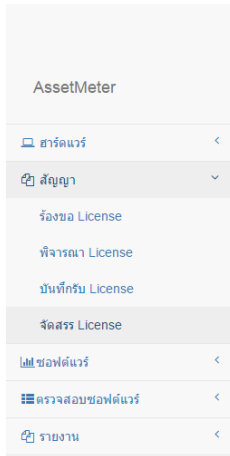
ภาพที่ 3 ระบบย่อยของ AssetMeter

ระบบบริหารจัดการอุปกรณ์ เป็นระบบที่เก็บรวบรวมข้อมูลอุปกรณ์ บันทึกเจ้าของอุปกรณ์ และยกเลิกการใช้งานอุปกรณ์ ดังตัวอย่างหน้าจอในภาพที่ 4



ภาพที่ 4 แสดงตัวอย่างหน้าจอระบบบริหารจัดการอุปกรณ์

ระบบบริหารจัดการสัญญา เป็นระบบที่รองรับการทำงานตั้งแต่การร้องขอลิขสิทธิ์ของโปรแกรม (License) ใหม่ พิจารณาคำร้องขอบันทึกลิขสิทธิ์ของโปรแกรมตามคำร้องที่ได้รับการอนุมัติ และจัดสรรลิขสิทธิ์ของโปรแกรมที่ได้รับไปยังอุปกรณ์ของหน่วยงานต่างๆ ที่ร้องขอ ดังตัวอย่างหน้าจอในภาพที่ 5



## จัดสรร License

จัดสรร License ให้แต่ละอุปกรณ์

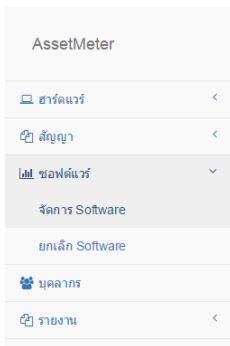
-เลือกแผนก-      -เลือกซอฟต์แวร์-

License Usage	Computer Name	Core	Memory	Processor
<input type="checkbox"/>	WINSERVER	1	2047	Intel(R) Core(TM) i7-3612QM CPU @ 2.10GHz
<input type="checkbox"/>	HONG	4	8053	Intel(R) Core(TM) i7-3612QM CPU @ 2.10GHz
<input type="checkbox"/>	WINCLIENT	4	2047	Intel(R) Core(TM) i7-3612QM CPU @ 2.10GHz

[บันทึก](#)

ภาพที่ 5 แสดงตัวอย่างหน้าจอระบบบริหารจัดการสัญญา

ระบบบริหารจัดการซอฟต์แวร์ เป็นระบบที่เก็บรวบรวมข้อมูลซอฟต์แวร์ที่ติดตั้งอยู่ในแต่ละอุปกรณ์ หรือที่ถูกถอนการติดตั้งแล้ว ทั้งในกรณีลิขสิทธิ์ของโปรแกรมหมดอายุ หรือถูกยกเลิก หรือยกเลิกการจัดสรรตั้งตัวอย่างหน้าจอในภาพที่ 6



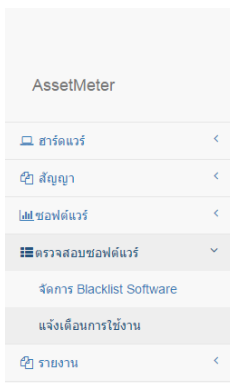
## จัดการซอฟต์แวร์

ตรวจสอบการติดตั้งซอฟต์แวร์ใหม่โดยทำ 4 ขั้นตอน ดังนี้

- Download** (View Details)
- Copy** (View Details)
- Execute** (View Details)
- Verify Software** (View Details)

ภาพที่ 6 แสดงตัวอย่างหน้าจอระบบบริหารจัดการซอฟต์แวร์

ระบบตรวจสอบซอฟต์แวร์ เป็นระบบแจ้งเตือนการหมดอายุทั้งฮาร์ดแวร์และซอฟต์แวร์ รวมถึงการตรวจจับซอฟต์แวร์ที่ไม่ได้รับอนุญาตให้ใช้งาน แต่ถูกติดตั้งในอุปกรณ์ของหน่วยงานดังกล่าวในภาพที่ 7



## จัดการ Blacklist Software

เพิ่มซอฟต์แวร์ที่ห้ามใช้งาน

ชื่อ:       หมายเหตุ:       [บันทึก](#)

Delete	Scan	Software Name	Description
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Line	เผยแพร่ข่าวสารอันเป็นเท็จ
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Steam	ห้ามเล่นเกม

[ออกรายงาน](#)

ภาพที่ 7 แสดงตัวอย่างหน้าจอระบบตรวจสอบซอฟต์แวร์



ระบบรายงาน เป็นระบบแสดงข้อมูลพร้อมจัดทำเป็นรายงานให้กับผู้ใช้งาน โดยผู้ใช้งานสามารถกำหนดเงื่อนไขเพื่อกรองข้อมูลที่ต้องการดูในรายงานได้ สั่งพิมพ์รายงานได้ และบันทึกรายงานในรูปแบบไฟล์ได้ดังตัวอย่างหน้าจอในภาพที่ 8

รายชื่อซอฟต์แวร์ที่ต้องถอนการติดตั้ง

แผนก	ชื่อคอมพิวเตอร์	ชื่อซอฟต์แวร์	รายละเอียด
เทคโนโลยี	HONG	Line	เผยแพร่ข่าวสารอันเป็นเท็จ
เทคโนโลยี	HONG	Steam	ห้ามเล่นเกมส์

ภาพที่ 8 แสดงตัวอย่างหน้าจอระบบรายงาน

### 3. บทสรุป

บทความนี้นำเสนอการพัฒนาระบบ AssetMeter ทั้งการวิเคราะห์หาข้อกำหนดความต้องการของระบบ การออกแบบสถาปัตยกรรมและองค์ประกอบต่างๆ ของระบบ ทั้งหน้าจอ ฐานข้อมูล และโปรแกรมที่เป็นส่วนสำคัญของการประมวลผล รวมทั้งนำเสนอตัวอย่างการใช้ระบบเพื่อจัดการกับฮาร์ดแวร์ ซอฟต์แวร์ จัดสรรลิขสิทธิ์ของโปรแกรม ตรวจสอบซอฟต์แวร์ว่ามีลิขสิทธิ์การใช้งานอย่างถูกต้องตามระเบียบหรือไม่ และจัดทำรายงาน โดยองค์กรที่ซื้อระบบ AssetMeter ไปใช้งานจะได้รับประโยชน์ดังนี้

1. มีข้อมูลฮาร์ดแวร์และซอฟต์แวร์เพื่อนำไปใช้เป็นส่วนหนึ่งของการวางแผนการจัดซื้ออุปกรณ์และซอฟต์แวร์ให้กับองค์กรได้อย่างเหมาะสม
2. รวบรวมข้อมูลด้านลิขสิทธิ์ของซอฟต์แวร์และสัญญาด้านการดูแลรักษาได้อย่างรวดเร็ว ทำให้องค์กรสามารถตรวจสอบการใช้ซอฟต์แวร์ให้เป็นไปอย่างถูกต้องตามกฎหมาย นโยบายและกฎระเบียบขององค์กรได้อย่างมีประสิทธิภาพ

#### บรรณานุกรม

- ปัญญาชาติ ปุณณชัยยะ. (2557). เอกสารประกอบการบรรยายวิชา IS623: Information System Analysis and Design ปีการศึกษา 2557 ภาคการศึกษาที่ 2.
- มูลนิธิสถาบันวิจัยเพื่อการพัฒนาประเทศไทย. (2557). *โครงการสำรวจข้อมูลตลาดซอฟต์แวร์และบริการซอฟต์แวร์ ประจำปี 2556 และคาดการณ์ปี 2557*. สืบค้นจาก [http://www.sipa.or.th/sites/default/files/final\\_sipa\\_2556\\_publish.pdf](http://www.sipa.or.th/sites/default/files/final_sipa_2556_publish.pdf).
- Miller, D. (2557). Startbootstrap-sb-admin-2. Retrieved September 10, 2015, from <https://github.com/BlackrockDigital/startbootstrap-sb-admin-2>.
- Pragim Technologies. JQuery datatables get data from database table. Retrieved September 8, 2015, from <http://csharp-video-tutorials.blogspot.com/2015/08/Jquery-datatables-get-data-from.html>.
- Capterra. Top IT Asset Management Software Products. Retrieved September 10, 2015, from <http://www.capterra.com/it-asset-management-software/>.
- Khan, M. (2009). Export DataSet or DataTable to Word Excel PDF and CSV Formats. Retrieved January 10,

2016, from <http://www.aspsnippets.com/Articles/Export-DataSet-or-DataTable-to-Word-Excel-PDF-and-CSV-Formats.aspx>.

Hruby, P. (2013, December 27). Visio Stencil and Template for UML 2.5. Retrieved September 8, 2015, from <http://www.softwarestencils.com/uml/>.

Brind, M. (2013). Windows Authentication with ASP.NET Web Pages. Retrieved February 20, 2016, from <http://www.mikesdotnetting.com/article/216/Windows-authentication-with-asp-net-web-pages>.