รายงานสหกิจศึกษาฉบับสมบูรณ์

การปรับปรุงและพัฒนาการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคาร ภายในสถานีรถไฟฟ้าใต้ดินสายสีส้ม

Enhancing the efficiency and effectiveness of performance by using Building Information Modelling in Mass Rapid Transit Orange line project

นางสาวสุปริญญา จันทรสุรินทร์

นางสาวอาทิมา สองทิศ

สาขาวิชาวิศวกรรมขนส่งทางราง ภาควิชาวิศวกรรมศาสตร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

รายงานสหกิจศึกษาฉบับสมบูรณ์

การปรับปรุงและพัฒนาการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคาร ภายในสถานีรถไฟฟ้าใต้ดินสายสีส้ม

Enhancing the efficiency and effectiveness of performance by using Building Information Modelling in Mass Rapid Transit Orange line project

นางสาวสุปริญญา จันทรสุรินทร์

นางสาวอาทิมา สองทิศ

สาขาวิชาวิศวกรรมขนส่งทางราง ภาควิชาวิศวกรรมศาสตร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงงานสหกิจศึกษา การปรับปรุงและพัฒนาการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคารภายในสถานีรถไฟฟ้าใต้ดินสายสีส้ม
ชื่อ - สกุล นักศึกษา นางสาวสุปริญญา จันทรสุรินทร์
นางสาวอาทิมา สองทิศ
คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมเครื่องกล สาขาวิศวกรรมขนส่งทางราง
ชื่อ – สกุล อาจารย์นิเทศ ดร.จิตตราภรณ์ วงศางาม
ชื่อ – สกุล ผู้นิเทศงาน นายยุทธนา คงคารัตน์
ชื่อสถานประกอบการ บริษัท ทีม คอนซัลติ้ง เอนจิเนียริ่ง แอนด์ แมเนจเม้นท์ จำกัด (มหาชน)

บทคัดย่อ

โครงงานสหกิจศึกษาเรื่อง การปรับปรุงและพัฒนากรปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคารภายในสถานีรถไฟฟ้าสายสีส้ม ถูกจัดทำขึ้นโดยมีวัตถุประสงค์ในการแก้ไขปัญหาที่เกิดขึ้นในระหว่างการเขียนแบบจำลอง 3 มิติ เพื่อทำให้ระยะเวลาในการสร้างแบบจำลอง 3 มิติลดลง หลีกเลี่ยงปัญหาการปฏิบัติงานที่ซ้ำซ้อนซึ่งช่วยให้มีข้อมูลที่ผิดพลาดลดน้อยลง อีกทั้งยังสามารถลดค่าใช้จ่ายในส่วนของแรงงานได้เช่นกัน ซึ่งแบบจำลองสารสนเทศอาคาร (Building Information Modeling ; BIM) เป็นกระบวนการที่ถูกพัฒนาขึ้นสำหรับการก่อสร้างโดยเริ่มตั้งแต่การออกแบบอาคาร การวางโครงสร้างของอาคาร การประเมินราคา รวมไปถึงการวางแผนงานระบบภายในอาคารโดยการสร้างแบบจำลอง 3 มิติ อันประกอบไปด้วยข้อมูลสารสนเทศ ทำให้เกิดการทำงานร่วมกันระหว่างสถาปนิก วิศวกร ผู้รับเหมาและผู้ที่เกี่ยวข้อง ในโครงงานครั้งนี้โปรแกรม Autodesk Revit 2017 และซอฟท์แวร์ Dynamo ได้ถูกใช้ในการสร้างแบบจำลองของอาคารและระบบ ได้แก่ 1. ระบบประปาและสุขาภิบาล 2. ระบบป้องกันอัคคีภัยและ 3. ระบบแจ้งเหตุเพลิงไหม้ภายในสถานีรถไฟฟ้าใต้ดินสายสีส้มในสัญญาที่ 1 และ 2 จึงทำให้พบปัญหาซึ่งก่อให้เกิดความล่าช้าและความซ้ำซ้อนในระหว่างการสร้างแบบจำลอง 3 มิติ ดังนั้นผลการศึกษาพบว่าปัญหาที่เกิดขึ้นถูกแก้ไขได้โดยใช้โปรแกรม Autodesk Revit 2017 ซอฟท์แวร์ Dynamo โปรแกรม Microsoft Excel และภาษา Python ซึ่งการเลือกใช้โปรแกรมเหล่านี้จะขึ้นอยู่กับปัญหาที่แตกต่างกันออกไป ทำให้ช่วยลดการซ้ำซ้อนในการปฏิบัติงานเนื่องจากใช้โปรแกรมการปฏิบัติงานลดน้อยลง ทำให้ระยะเวลาในการสร้างแบบจำลอง 3 มิติโดยรวมลดลงสูงสุดถึง 92.02 % และค่าใช้จ่ายในส่วนของแรงงานสามารถลดลงได้สูงสุดถึง 49.16%

คำสำคัญ : แบบจำลองสารสนเทศอาคาร, Revit, สถานีรถไฟฟ้าใต้ดินสายสีส้ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Cooperative Title: Enhancing the efficiency of the performance by using BIM in MRT orange line project

Student intern name: Suparinya Jandrasurin
Athima Songthit

Faculty: Engineering **Department:** Mechanical Engineering

Advisor name: Dr. Jittraporn Wongsa-ngam

Mentor Name: Yuttana Kongkarat

Company: TEAM Consulting Engineering and Management Public Company Limited

ABSTRACT

The objective of this project is to solve many kinds of problem found during the time when we draw a 3D model. We can reduce the time usage, avoid the duplicate works and reduce personnel costs by solving all these problems. Building Information Modeling or BIM is developed specially for building construction. Its use goes beyond the planning and design phase of the project extending to supporting processes including cost management, facility operation, and construction management. This cause interoperability between architects, engineers, contractors, and any other relevant person. The software used in this project is Autodesk Revit 2017 and Dynamo. This software is used in designing sanitary systems, fire alarm systems, and fire protection systems in Mass Transit' s Orange line project for the first and second contract. The result of this project is that we can use Autodesk Revit 2017, Dynamo, Microsoft Excel and Python to help us solve these problems. We found the original method or the normally used method to be less efficient than using the new method which we have developed for this project. The new methods can reduce the time used in drawing a 3D model for almost 92.02% and reduce the personnel cost by 49.16%

Keyword: Building Information Modeling, Revit, Mass Rapid Transit Orange Line project

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความกรุณาของอาจารย์ที่ปรึกษาที่มีพระคุณยิ่ง กราบขอบพระคุณ ดร.จิตตราภรณ์ วงศางาม ที่กรุณารับเป็นที่ปรึกษาและเสียสละเวลาที่มีค่าในการให้คำปรึกษาเกี่ยวกับแนวคิดและคำเสนอแนะอันเป็นประโยชน์อย่างยิ่งในการทำปริญญานิพนธ์ ตลอดจนตรวจสอบแก้ไขข้อบกพร่องทั้งเนื้อหาและรูปแบบของเล่มปริญญานิพนธ์มาโดยตลอด และขอขอบพระคุณอาจารย์ทุกท่านในสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่ได้ประสิทธิ์ประสาทความรู้และอบรมสั่งสอน อีกทั้งยังให้ความช่วยเหลือผู้ศึกษาในส่วนที่เกี่ยวข้องกับปริญญานิพนธ์จนสำเร็จ

ขอขอบคุณ คุณยุทธนา คงคารัตน์และทีมงานทุกท่านที่ให้คำแนะนำเกี่ยวกับหัวข้อปริญญานิพนธ์ แนวทางการเขียนเล่มปริญญานิพนธ์ รวมไปถึงคำแนะนำและการช่วยเหลือในการใช้โปรแกรมและการปฏิบัติงาน

ขอขอบคุณเจ้าหน้าที่และบุคคลากรของภาควิชาวิศวกรรมเครื่องกล สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังและบริษัททีม คอนซัลติ้ง เอนจิเนียริ่ง แอนด์ แมเนจเม้นท์ จำกัด (มหาชน) ในการช่วยเหลือและติดต่อประสานงานให้ความสะดวกตลอดมา

ขอกราบขอบพระคุณบิดามารดาของครอบครัวจันทรสุรินทร์และครอบครัวสองทิศที่สนับสนุนให้ได้รับการศึกษาเล่าเรียน ให้กำลังใจและกำลังทรัพย์ตลอดจนให้การสนับสนุนทุกสิ่งทุกอย่างเสมอมาจนสำเร็จการศึกษา

ท้ายที่สุดขอขอบคุณเพื่อนๆ ที่ให้ความช่วยเหลือ ความร่วมมือและกำลังใจที่ให้ผู้ศึกษาอย่างสม่ำเสมอ ขอขอบคุณนายณภัทร สงวนดิลกรัตน์ รุ่นพี่ที่คอยให้คำแนะนำเกี่ยวกับข้อมูลในการจัดทำปริญญานิพนธ์ และขอขอบคุณ ดร.สรัสไชย องค์ประเสริฐ เป็นกรณีพิเศษสำหรับการเสียสละเวลาในการให้คำวิพากษ์วิจารณ์ผลงาน ทำให้ผู้ศึกษาสามารถสรุปผลการศึกษาได้อย่างสมบูรณ์ และขอขอบคุณผู้เป็นเจ้าของข้อมูลทุกข้อมูลที่ผู้ศึกษาได้ใช้ในการศึกษา มา ณ โอกาสนี้

ประโยชน์จากปริญญานิพนธ์ฉบับนี้ ผู้ศึกษาขอมอบแก่ทุกท่านที่สนใจในการศึกษาและนำไปประยุกต์ใช้ในการปฏิบัติงาน รวมถึงเสริมสร้างความรู้แก่ผู้ศึกษาท่านอื่นในครั้งต่อไป

สุปริญญา จันทรสุรินทร์
อาทิมา สองทิศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อ I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญรูปภาพ VII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญ 1

1.3 ขอบเขตของการวิจัย 2

1.4 วิธีการดำเนินการวิจัย 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 4

2.1 กระบวนการปฏิบัติงานที่ใช้ 4

2.1.1 กระบวนการแบบจำลองสารสนเทศอาคาร (BIM) 4

2.2 ระบบซอฟท์แวร์ที่เกี่ยวข้องในงานวิจัย 8

2.2.1 โปรแกรม Autodesk Revit 8

2.2.2 ซอฟท์แวร์ Dynamo 17

2.2.3 ภาษา Python 19

2.3 PYCHARM 22

2.3.1 ลักษณะเด่นของ Pycharm 23

2.4 ทฤษฎีที่เกี่ยวข้องงานวิจัย 23

2.4.1 สมการการคำนวณค่า Head loss สำหรับวาล์วและข้อต่อของท่อจากสมการการคำนวณของ Hazen and Williams 24

2.4.2 สมการการคำนวณค่าความดันตก (Pressure drop) ในท่อจากสมการการคำนวณของ Haaland 24

2.4.3 The Moody chart 25

2.4.4 Equivalent Length of Straight for valves and fitting 25

2.4.5 สมการการคำนวณหาขนาดของปั๊มน้ำ 26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

บทที่ 3 วิธีดำเนินการวิจัย 27

3.1 กระบวนการทำงาน 27

3.2 ปัญหาหลักที่ก่อให้เกิดผลกระทบในระหว่างการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคาร 28

3.3 วิธีการแก้ปัญหา 28

บทที่ 4 ผลการวิจัย 80

4.1 ผลการวิจัย 80

4.2 การเปรียบเทียบระยะเวลาและค่าชั่วโมงแรงงานในการปฏิบัติงานด้วยวิธีการเดิมและวิธีการใหม่ 91

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ 94

5.1 สรุปผลการวิจัย 94

5.2 ข้อเสนอแนะ 95

บรรณานุกรม 96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ 2.1 แสดงชื่อและหน้าที่ของคำสั่งในการเขียน LOOKUP TABLE.....17
ตารางที่ 4.1 แสดงระยะเวลาและค่าชั่วโมงแรงงานในการปฏิบัติงานด้วยวิธีการเดิมและวิธีการใหม่ของทุกสถานี
ในสถานีรถไฟฟ้าใต้ดินสายสีส้ม.....91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่ 2.1 แสดงภาพอธิบายแนวคิดของกระบวนการแบบจำลองสารสนเทศอาคาร (BIM) 4
รูปที่ 2.2 แสดงตัวอย่างของเก้าอี้โดยที่ข้อมูลที่ไม่ใช่กราฟิก (NON-GRAPHICS) เพิ่มขึ้นตามระดับชั้นของโครงการ 6
รูปที่ 2.3 แสดงลักษณะของมิติแต่ละมิติ 7
รูปที่ 2.4 แสดงลักษณะและเครื่องมือเบื้องต้นในโปรแกรม AUTODESK REVIT 9
รูปที่ 2.5 แสดงตัวอย่างของ FAMILY 10
รูปที่ 2.6 แสดงตัวอย่างของ SCHEDULE ในโปรแกรม AUTODESK REVIT 11
รูปที่ 2.7 แสดงหน้าต่างคำสั่งเพื่อกำหนด PROPERTIES ของ SCHEDULE 13
รูปที่ 2.8 แสดงภาพตัวอย่างของระบบสุขาภิบาลและการประปา (SANITARY AND PLUMBING SYSTEM) ในโปรแกรม AUTODESK REVIT 14
รูปที่ 2.9 แสดงสัญลักษณ์ของ SECTION ซึ่งสามารถแก้ไขได้โดยการยืดเข้า-ออกของลูกศรสีน้ำเงิน 15
รูปที่ 2.10 แสดงภาพมุมมองที่โปรแกรมแสดงขึ้นหลังจากที่ผู้ใช้ทำการตัด SECTION (ภาพขวา) 16
รูปที่ 2.11 แสดงตัวอย่างของ GRID LINES 16
รูปที่ 2.12 แสดงลักษณะของซอฟท์แวร์ DYNAMO 18
รูปที่ 2.13 แสดงส่วนประกอบของโหนด 19
รูปที่ 2.14 แสดงหน้าต่างการทำงานของ PYTHON ใน PYCHARM 23
รูปที่ 2.15 แสดงตัวอย่างแผนภาพ MOODY CHART 25
รูปที่ 2.16 แสดงภาพตาราง EQUIVALENT LENGTH ของข้อต่อท่อเพื่อใช้ในการคำนวนความดันตก 25
รูปที่ 3.1 แสดงแนวคิดกระบวนการจัดส่งของ BIM 27
รูปที่ 3.2 แสดงภาพแบบจำลอง 3 มิติของอาคาร 5 ชั้น 29
รูปที่ 3.3 แสดงหน้าแปลนของแบบจำลองในระดับชั้นที่ 4 29
รูปที่ 3.4 แสดงหน้า SECTION VIEW ของอาคาร 5 ชั้น 30
รูปที่ 3.5 แสดงการใช้ DETIAL VIEW 30
รูปที่ 3.6 แสดงชั้นระดับที่ได้หลังจากการใช้ DETAIL VIEW 31
รูปที่ 3.7 แสดงเครื่องมือ SECTION/ELEVATION HEIGHT EDITOR 31
รูปที่ 3.8 แสดงการใช้งานเครื่องมือ SECTION/ELEVATION HEIGHT EDITOR 32
รูปที่ 3.9 แสดงการใช้งานเครื่องมือ SECTION/ELEVATION HEIGHT EDITOR 32
รูปที่ 3.10 แสดงแถบเครื่องมือ PYREVIT และเครื่องมือใน RIBBON เมื่อรันโค้ดเสร็จสมบูรณ์ 36
รูปที่ 3.11 แสดงค่า BOP (BOTTOM OF PIPE) ที่เปลี่ยนแปลงไปเมื่อทำการกดใช้เครื่องมือ PIPES 377
รูปที่ 3.12 แสดงวิธีการระบุชั้นระดับที่ถูกต้องของท่อชนิดกลม (PIPES) ในขั้นตอนที่ 9 41
รูปที่ 3.13 แสดงแผนภาพ DYNAMO ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 1 42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

รูปที่ 3.14 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 2.....42
รูปที่ 3.15 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 3.....43
รูปที่ 3.16 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 3.3.....43
รูปที่ 3.17 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 4.....44
รูปที่ 3.18 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 5.....45
รูปที่ 3.19 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 6.3.....46
รูปที่ 3.20 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 7.....47
รูปที่ 3.21 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 8 ถึง 10.....47
รูปที่ 3.22 แสดงการแจ้งเตือนเมื่อโปรแกรมพบว่ามีท่อที่ไม่เชื่อมต่อกันและให้ยุติการเชื่อมต่อนั้น.....48
รูปที่ 3.23 แสดงการแจ้งเตือนเมื่อโปรแกรมพบว่ามีท่อที่ไม่เชื่อมต่อกันและให้ลบวัตถุ (ท่อ) นั้น.....48
รูปที่ 3.24 แสดงแผนภาพ Python Script สำหรับการนำไปใช้ในการการสร้างแบบจำลอง 3 มิติของสถานีรถไฟฟ้าใต้ดินสายสีส้มในขั้นตอนการระบุชื่อของชั้นระดับ.....49
รูปที่ 3.25 แสดงแผนภาพ Python Script สำหรับการนำไปใช้ในการสร้างแบบจำลอง 3 มิติของสถานี.....50
รูปที่ 3.26 แสดงข้อมูลต่างๆ ใน Pipe Schedule.....51
รูปที่ 3.27 แสดงข้อมูลใน Pipe Selecting Data Schedule.....52
รูปที่ 3.28 แสดงข้อมูลใน Pipe Fitting Schedule.....53
รูปที่ 3.29 แสดงข้อมูลใน Pipe Accessory Schedule.....54
รูปที่ 3.30 แสดงตาราง Equivalent Length ของอุปกรณ์ข้อต่อ (Elbow).....55
รูปที่ 3.31 แสดงตาราง Equivalent Length ของ Pipe Accessory.....55
รูปที่ 3.32 แสดงหน้าต่าง Family Type.....56
รูปที่ 3.33 แสดงแผนภาพ Dynamo แสดงการถ่ายโอนข้อมูลของตาราง Equivalent Length จาก Excel ไปยัง Revit ของข้อต่อชนิด Butt-Welded-CS.....57
รูปที่ 3.34 แสดงข้อมูลใน Pipe Fitting Selecting Data Schedule.....57
รูปที่ 3.35 แสดงข้อมูลใน Pipe Accessory Selecting Data Schedule.....58
รูปที่ 3.36 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 1.....59
รูปที่ 3.37 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 2.....59
รูปที่ 3.38 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 3.....60
รูปที่ 3.39 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 4.....60
รูปที่ 3.40 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 5.....60
รูปที่ 3.41 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 6.....61
รูปที่ 3.42 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 7.....61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

รูปที่ 3.43 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในท่อ (PIPE) ในขั้นตอนที่ 8.....62
รูปที่ 3.44 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในท่อ (PIPE) ในขั้นตอนที่ 9.....62
รูปที่ 3.45 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในท่อ (PIPE) ในขั้นตอนที่ 10.....63
รูปที่ 3.46 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์ข้อต่อ (PIPE FITTING)ในขั้นตอนที่1..63
รูปที่ 3.47 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์ข้อต่อ (PIPE FITTING)ในขั้นตอนที่2..64
รูปที่ 3.48 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์ข้อต่อ (PIPE FITTING)ในขั้นตอนที่3..64
รูปที่ 3.49 แสดงตัวอย่างภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์ข้อต่อ (PIPE FITTING) ชนิด TEE SAN WITH GROOVE ในขั้นตอนที่ 4.....65
รูปที่ 3.50 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์ข้อต่อ (PIPE FITTING)ในขั้นตอนที่5..65
รูปที่ 3.51 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์ข้อต่อ (PIPE FITTING)ในขั้นตอนที่6..66
รูปที่ 3.52 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์ข้อต่อ (PIPE FITTING)ในขั้นตอนที่7..66
รูปที่ 3.53 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์ข้อต่อ (PIPE FITTING)ในขั้นตอนที่8..67
รูปที่ 3.54 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์จำพวกวาล์ว (PIPE ACCESSORIES) ในขั้นตอนที่ 1.....68
รูปที่ 3.55 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์จำพวกวาล์ว (PIPE ACCESSORIES) ในขั้นตอนที่ 2.....68
รูปที่ 3.56 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์จำพวกวาล์ว (PIPE ACCESSORIES) ในขั้นตอนที่ 3.....69
รูปที่ 3.57 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์จำพวกวาล์ว (PIPE ACCESSORIES) ในขั้นตอนที่ 4.....69
รูปที่ 3.58 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์จำพวกวาล์ว (PIPE ACCESSORIES) ในขั้นตอนที่ 5.....70
รูปที่ 3.59 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์จำพวกวาล์ว (PIPE ACCESSORIES) ในขั้นตอนที่ 6.....70
รูปที่ 3.60 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์จำพวกวาล์ว (PIPE ACCESSORIES) ในขั้นตอนที่ 7.....71
รูปที่ 3.61 แสดงภาพขยายแผนภาพ DYNAMO ของการคำนวณในอุปกรณ์จำพวกวาล์ว (PIPE ACCESSORIES) ในขั้นตอนที่ 8.....71
รูปที่ 3.62 แสดงตารางข้อมูลต่างๆ ของ PIPE SCHEDULE ในโปรแกรม MICROSOFT EXCEL.....72
รูปที่ 3.63 แสดงตารางข้อมูลต่างๆ ของ PIPE FITTING SCHEDULE ในโปรแกรม MICROSOFT EXCEL.....72
รูปที่ 3.64 แสดงตารางข้อมูลต่างๆ ของ PIPE ACCESSORIES SCHEDULE ในโปรแกรม MICROSOFT EXCEL.....72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

รูปที่ 3.65 แสดงภาพขยายแผนภาพ DYNAMO ในการสร้าง GRID LINE ในมุมมองสามมิติในขั้นตอนที่ 1 73
รูปที่ 3.66 แสดงภาพขยายแผนภาพ DYNAMO ในการสร้าง GRID LINE ในมุมมองสามมิติในขั้นตอนที่ 2 73
รูปที่ 3.67 แสดงภาพขยายแผนภาพ DYNAMO ในการสร้าง GRID LINE ในมุมมองสามมิติในขั้นตอนที่ 3 73
รูปที่ 3.68 แสดงภาพขยายแผนภาพ DYNAMO ในการสร้าง GRID LINE ในมุมมองสามมิติในขั้นตอนที่ 5 75
รูปที่ 3.69 แสดงภาพขยายแผนภาพ DYNAMO ในการระบุ GRID LINE ที่ใกล้ที่สุดของวัตถุที่ต้องการในขั้นตอนที่ 2 76
รูปที่ 3.70 แสดงภาพขยายแผนภาพ DYNAMO ในการระบุ GRID LINE ที่ใกล้ที่สุดของวัตถุที่ต้องการในขั้นตอนที่ 3 77
รูปที่ 3.71 แสดงภาพขยายแผนภาพ DYNAMO ในการระบุ GRID LINE ที่ใกล้ที่สุดของวัตถุที่ต้องการในขั้นตอนที่ 4 78
รูปที่ 3.72 แสดงภาพขยายแผนภาพ DYNAMO ในการระบุ GRID LINE ที่ใกล้ที่สุดของวัตถุที่ต้องการในขั้นตอนที่ 5 78
ภาพที่ 3.73 แสดงภาพขยายแผนภาพ DYNAMO ในการระบุ GRID LINE ที่ใกล้ที่สุดของวัตถุที่ต้องการ ในขั้นตอนที่ 6 79
รูปที่ 4.1 แสดงมุมมองของวัตถุในชั้นระดับที่ต้องการ 80
รูปที่ 4.2 แสดงแถบเครื่องมือ PYREVIT (หมายเลข 1) และเครื่องมือ PIPES ใน RIBBON(หมายเลข 2).....81
รูปที่ 4.3 แสดงชั้นระดับ (LEVEL) และค่า BOP ของท่อที่ต้องการ (กรอบสีแดง).....82
รูปที่ 4.4 แสดงการระบุชั้นระดับของท่อชนิดกลม (PIPES) ในกรอบสีแดง.....83
รูปที่ 4.5 แสดงการระบุชั้นระดับของท่อชนิดเหลี่ยม (DUCTS) ในกรอบสีแดง.....83
รูปที่ 4.6 แสดง PIPE SCHEDULE ที่ได้จากการปฏิบัติงานจริงของสถานีรถไฟฟ้าใต้ดินสายสีส้ม 84
รูปที่ 4.7 แสดง PIPE FITTING SCHEDULE ที่ได้จากการปฏิบัติงานจริงของสถานีรถไฟฟ้าใต้ดินสายสีส้ม 85
รูปที่ 4.8 แสดงชั้นระดับ (กรอบสีแดง) และค่า OFFSET (กรอบสีน้ำเงิน) ในหน้าต่าง PROPERTIES ของอุปกรณ์ข้อต่อ 85
รูปที่ 4.9 แสดงตารางคุณสมบัติต่างๆท่อ อุปกรณ์ข้อต่อและ PIPE ACCESSORY ทั้งหมดในระบบประปาและสุขาภิบาล 86
รูปที่ 4.10 แสดง GRID LINES ในรูปแบบ 3 มิติ.....87
รูปที่ 4.11 แสดง GRID LINES ในรูปแบบ 3 มิติเมื่อนำไปใช้ในการปฏิบัติงานจริงในสถานีรถไฟฟ้าใต้ดินสายสีส้ม ในระบบสุขาภิบาลและการประปา.....88
รูปที่ 4.12 แสดงตำแหน่งของ ELEMENT ในมุมมอง 2 มิติ (กรอบสีแดง).....89
รูปที่ 4.13 แสดงตำแหน่งของ ELEMENT ในมุมมอง 3 มิติ (กรอบสีแดง).....89

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

รูปที่ 4.14 แสดงตำแหน่งของวัตถุในมุมมอง 3 มิติเมื่อนำไปใช้ในการปฏิบัติงานจริงในสถานีรถไฟฟ้าใต้ดินสายสีส้มในระบบสุขาภิบาลและการประปา.....90
รูปที่ 4.15 แสดงกราฟเปรียบเทียบชั่วโมงแรงงานที่ใช้ในระหว่างการแก้ไขด้วยวิธีการเดิมและวิธีการใหม่ 92
รูปที่ 4.16 แสดงกราฟเปรียบเทียบจำนวนเงินที่ใช้ในระหว่างการแก้ไขด้วยวิธีการเดิมและวิธีการใหม่.....93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

แบบจำลองสารสนเทศอาคาร หรือ BIM (Building Information Modeling) เป็นกระบวนการซึ่งถูกพัฒนาขึ้นเพื่อใช้ในกระบวนการออกแบบและก่อสร้างอาคาร โดยการใช้โปรแกรม Autodesk Revit และซอฟท์แวร์ Dynamo จำลองอาคาร (Building Model) และข้อมูลสารสนเทศ (Information) ในองค์ประกอบของแบบจำลองอาคาร ซึ่งผู้ใช้สามารถทำงานได้ในทั้งการเขียนแบบในมุมมอง 2 มิติและการสร้างแบบจำลอง 3 มิติขึ้นมาได้ภายในโปรแกรมเดียว อีกทั้งผู้ใช้ยังสามารถเพิ่มข้อมูลวัตถุหรือข้อมูล ลงไปในองค์ประกอบของแบบจำลอง 3 มิติ อาทิเช่น ข้อมูลเชิงกราฟฟิก (ขนาด, ระยะ, สี, วัตถุ) ทั้งนี้กระบวนการแบบจำลองสารสนเทศอาคารจะช่วยให้การปฏิบัติงานเป็นไปอย่างสะดวกและรวดเร็วมากกว่าการปฏิบัติงานด้วยวิธีการเดิม เนื่องจากในการปฏิบัติงานด้วยวิธีการเดิมนั้น ผู้ใช้จะต้องปฏิบัติงานจากหลายโปรแกรม จากนั้นจึงนำงานที่ได้มารวบรวมกันเพื่อให้ได้งานเพียงชิ้นเดียวตามที่ต้องการ

ทั้งนี้การปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศนี้ยังพบปัญหาที่ก่อให้เกิดผลที่ตามมาคือการสร้างแบบจำลองเป็นไปอย่างล่าช้าหรือพบข้อมูลที่มีความผิดพลาด ดังนั้นวิธีการแก้ไขปัญหาที่เกิดขึ้นเหล่านี้ขึ้นอยู่กับรูปแบบของปัญหา ซึ่งโดยส่วนใหญ่จะเป็นการนำซอฟท์แวร์และโปรแกรมอื่นเข้ามาช่วยในการแก้ไขปัญหา เช่น ซอฟท์แวร์ Dynamo โปรแกรม Microsoft Excel และภาษา Python อาทิเช่น การปรับปรุงเครื่องมือในโปรแกรม Autodesk Revit ให้ใช้งานได้อย่างมีประสิทธิภาพมากยิ่งขึ้น การนำซอฟท์แวร์ Dynamo เข้ามาเพื่อใช้ในการเขียนโค้ดเพิ่มเติมเพื่อเพิ่มเครื่องมือที่ไม่พบในโปรแกรม Autodesk Revit การนำโปรแกรม Microsoft Excel มาใช้ในการเรียบเรียงข้อมูลที่ได้จากการคำนวณ หรือการนำภาษา Python เข้ามาเขียนโค้ดเพิ่มเติมเพื่อสร้างเครื่องมือใหม่ในโปรแกรม Autodesk Revit เป็นต้น

ผู้ศึกษาคาดว่าผลลัพธ์ที่ได้จากการดำเนินการวิจัยครั้งนี้สามารถช่วยให้การปฏิบัติงานไปเป็นด้วยความสะดวกและรวดเร็ว อีกทั้งยังช่วยให้ชิ้นงานที่เกิดขึ้นมีความละเอียดและมีข้อมูลที่ครอบคลุมมากยิ่งขึ้น นอกจากนี้ยังสามารถทำให้บุคคลที่เกี่ยวข้องต่อการปฏิบัติงานนี้สามารถเข้าใจในชิ้นงานได้

1.2 วัตถุประสงค์ของการวิจัย

1) ปรับปรุงการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคารโดยใช้โปรแกรม Autodesk Revit ควบคู่กับซอฟท์แวร์ Dynamo โปรแกรม Microsoft Excel และภาษา Python
2) ลดระยะเวลาในการปฏิบัติงานของกระบวนการแบบจำลองสารสนเทศอาคาร
3) หลีกเลี่ยงการซ้ำซ้อนของการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคารซึ่งก่อให้เกิดความผิดพลาดของข้อมูล
4) ลดค่าใช้จ่ายด้านแรงงานของการปฏิบัติงานภายในสถานีรถไฟฟ้าสายสีส้มในสัญญาที่ 1 และ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของการวิจัย

1) ศึกษาและวิเคราะห์ปัญหาที่เกิดขึ้นระหว่างการปฏิบัติงานของกระบวนการแบบจำลองสารสนเทศอาคารภายในสถานีรถไฟฟ้าสายสีส้มในสัญญาที่ 1 และ 2
2) แก้ปัญหาที่เกิดขึ้นโดยใช้โปรแกรม Autodesk Revit ซอฟท์แวร์ Dynamo โปรแกรม Microsoft Excel และภาษา Python

1.4 วิธีการดำเนินการวิจัย

1) กำหนดหัวข้อและขอบเขตของการดำเนินการวิจัย
2) ศึกษาการปฏิบัติงานของกระบวนการแบบจำลองสารสนเทศอาคาร (BIM) ภายในสถานีรถไฟฟ้าใต้ดินสายสีส้มในสัญญาที่ 1 และ 2
3) ศึกษาโปรแกรม Autodesk Revit รวมถึงการทำงานของโปรแกรมและการใช้งานเครื่องมือภายในโปรแกรม
4) ศึกษาซอฟท์แวร์ Dynamo ซึ่งเป็นเครื่องมือในโปรแกรม Autodesk Revit ที่ช่วยในการแก้ไขปัญหาการออกแบบที่ซับซ้อนและปรับปรุงขั้นตอนของกระบวนการแบบจำลองสารสนเทศอาคารในโปรแกรม Autodesk Revit
5) ศึกษาภาษา Python เพื่อใช้เป็นภาษาในการเขียนโปรแกรมการทำงานและศึกษาโค้ดซึ่งใช้ในการเพิ่มเครื่องมือหรือตัวช่วยที่ไม่พบในโปรแกรม Autodesk Revit
6) ศึกษาและวิเคราะห์ปัญหาที่เกิดขึ้นระหว่างปฏิบัติงานกระบวนการแบบจำลองสารสนเทศอาคารภายในสถานีรถไฟฟ้าใต้ดินสายสีส้มในสัญญาที่ 1 และ 2
7) ออกแบบกระบวนการแก้ไขปัญหาที่เกิดขึ้นระหว่างปฏิบัติงานของกระบวนการแบบจำลองสารสนเทศอาคารด้วยโปรแกรม Autodesk Revit ซอฟท์แวร์ Dynamo โปรแกรม Microsoft Excel และภาษา Python
8) ทำการทดลองใช้โปรแกรม Autodesk Revit เชื่อมโยงกับการใช้ซอฟท์แวร์ Dynamo และโปรแกรม Microsoft Excel อีกทั้งร่วมกับภาษา Python เพื่อแก้ไขปัญหาที่พบจากข้อที่ 1.4.5
9) ปรับปรุงและแก้ไขจุดผิดพลาดที่เกิดขึ้นระหว่างการแก้ไขปัญหา
10) เปรียบเทียบระยะเวลาและค่าใช้จ่ายด้านแรงงานที่ใช้ในการปฏิบัติงานเพื่อแก้ไขปัญหาด้วยการใช้วิธีการเดิมและวิธีการใหม่
11) บันทึกและสรุปผลการทดลอง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1) วิธีการปฏิบัติงานของกระบวนการแบบจำลองสารสนเทศอาคาร (BIM) ด้วยโปรแกรม Autodesk Revit ซอฟท์แวร์ Dynamo โปรแกรม Microsoft Excel และภาษา Python มีความสะดวกและรวดเร็วมากยิ่งขึ้น
2) แบบจำลองทั้งในรูปแบบ 2 มิติและ 3 มิติที่เกิดขึ้นมีความละเอียดและมีข้อมูลที่ครอบคลุมมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ทำให้บุคคลที่เกี่ยวข้องต่อการปฏิบัติงานนี้สามารถเข้าใจในแบบจำลองทั้งในรูปแบบ 2 มิติและ 3 มิติที่เกิดขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

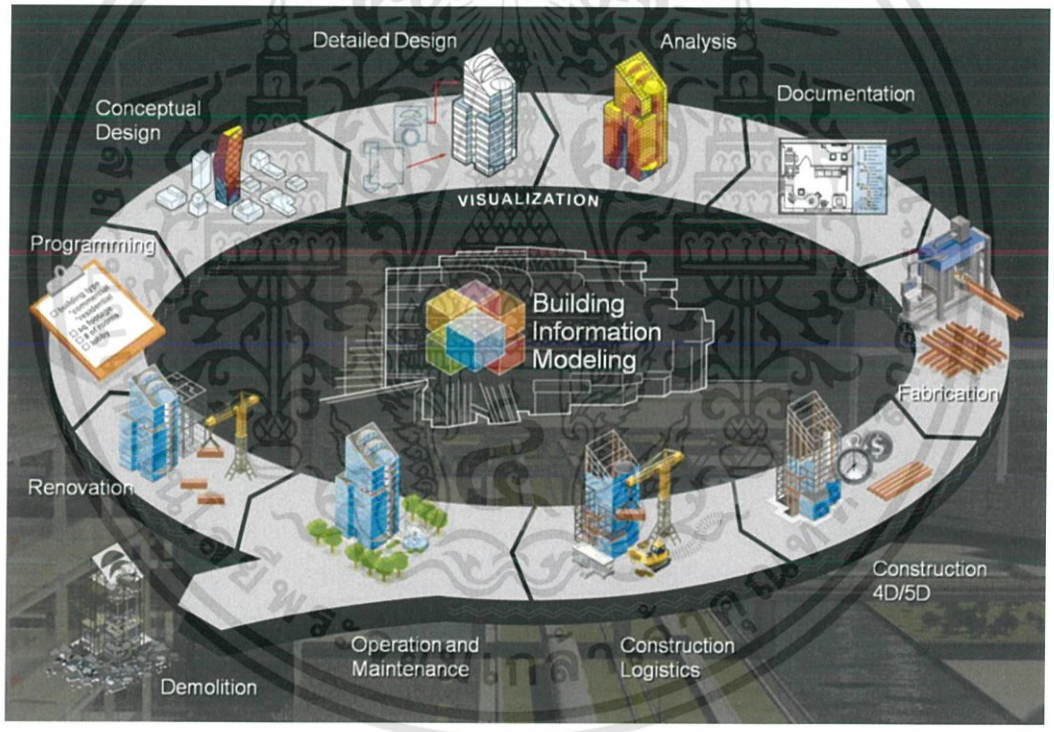
แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 กระบวนการปฏิบัติงานที่ใช้

ในโครงงานนี้ ผู้ศึกษาได้ใช้กระบวนการแบบจำลองสารสนเทศอาคาร (Building Information Modelling) ในการปฏิบัติงานเพื่อสร้างแบบจำลองของรถไฟฟ้าใต้ดินสายสีส้มในสัญญาที่ 1 และ 2

2.1.1 กระบวนการแบบจำลองสารสนเทศอาคาร (BIM)

กระบวนการแบบจำลองสารสนเทศอาคาร[1] หรือ Building Information Modelling (BIM) เป็นกระบวนการ (Process) ไม่ใช่ซอฟท์แวร์ (Software) ซึ่งเป็นแนวคิดที่ถูกพัฒนาขึ้นเพื่อใช้ในกระบวนการออกแบบและก่อสร้างอาคาร โดยการจำลองอาคาร (Building Model) และข้อมูลสารสนเทศ (Information) ในองค์ประกอบของแบบจำลองอาคาร



รูปที่ 2.1 แสดงภาพอธิบายแนวคิดของกระบวนการแบบจำลองสารสนเทศอาคาร (BIM)

1. แนวคิดของกระบวนการแบบจำลองสารสนเทศอาคาร

แนวคิดของ BIM ได้ถูกนำเสนอเป็นครั้งแรกโดย Charles M. Eastman ตีพิมพ์ในวารสารเอไอเอ (AIA Journal) เมื่อปี ค.ศ. 1975 ปัจจุบัน BIM ถูกนำมาใช้กับงานออกแบบทางสถาปัตยกรรมมากขึ้น เนื่องจากมีความสามารถในการผนวกการทำงานออกแบบสถาปัตยกรรมทั้งในรูปแบบ 2 มิติ และ 3 มิติ เข้าด้วยกันอย่างมีประสิทธิภาพ ซึ่งมีเป้าหมายในการบูรณาการการทำงานในขั้นตอนของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบและการก่อสร้างอาคารสถาปัตยกรรม คือลดขั้นตอน ลดความซ้ำซ้อน ลดความขัดแย้ง และลดปัญหาอันเกิดมาจากข้อมูลที่ผิดพลาดและอันเกิดขึ้นมาจากกระบวนการทำงานในลักษณะ จาก ภาพที่ 2.1 แสดงให้เห็นว่า BIM จึงถูกวางกระบวนการเริ่มต้นตั้งแต่การวางโจทย์ของโครงการ การออกแบบแนวคิดของโครงการ ไปจนถึงขั้นตอนการพัฒนาเพื่อนำไปสู่แบบสำหรับการก่อสร้าง งานก่อสร้างและการควบคุมการก่อสร้าง ไปจนถึงการการดูแลและบำรุงรักษาอาคารภายหลังจากที่สร้างอาคารนั้นเสร็จแล้ว เรียกง่ายๆ ว่า BIM นั้นมองกระบวนการตั้งแต่การเริ่มต้นอาคารไปจนครบวงจรชีวิต (Live Cycle) ของมันนั่นเอง

2. หลักการและกระบวนการของ BIM

การทำงานของ BIM เป็นการสร้างแบบจำลองอาคาร (Building Model) ขึ้นด้วยโปรแกรมคอมพิวเตอร์ โดยการจำลองอาคารนี้จะประกอบขึ้นไปด้วยองค์ประกอบของอาคาร (Building Component) เช่น เสา ผนัง พื้น หลังคา ประตู หน้าต่าง ซึ่งองค์ประกอบจะประกอบไปด้วยข้อมูลกราฟิก (Graphics) ทั้งในรูปแบบ 2 มิติและ 3 มิติ เช่น ขนาด ระยะ สี วัสดุ เป็นต้น และข้อมูลที่ไม่ใช่กราฟิก เช่น ข้อมูลผู้ผลิต รุ่น ราคา เป็นต้น ซึ่ง BIM จะทำการเก็บแบบจำลองอาคารพร้อมข้อมูลสารสนเทศทั้งหมดโดยจะรวบรวมไว้ที่ฐานข้อมูลกลางของระบบ

BIM สามารถแสดงผลแบบจำลองอาคารให้อยู่ในรูปของมุมมอง (View) ในลักษณะที่เหมาะสมตามการใช้งานได้ เช่น มุมมองรูป 2 มิติ ได้แก่ ผนังพื้น รูปด้าน รูปตัด เป็นต้น หรือมุมมองรูป 3 มิติ ได้แก่ รูปทัศนียภาพ ภาพ Isometric เป็นต้น นอกจากนี้ยังสามารถแสดงผลในรูปของตารางรายการข้อมูลเช่น ปริมาณวัสดุ หรือพื้นที่ใช้สอยจากแบบจำลองอาคาร และเนื่องจาก BIM จัดเก็บข้อมูลแบบจำลองอาคารไว้ในฐานส่วนกลาง ดังนั้นเมื่อผู้ใช้ทำการแก้ไขเปลี่ยนแปลงส่วนใดส่วนหนึ่งในแบบจำลองอาคาร การแก้ไขก็จะส่งผลไปยังฐานข้อมูลกลาง ทำให้การแสดงผลแบบจำลองอาคารในทุกมุมที่เกี่ยวข้อง มีความเปลี่ยนแปลงตามไปด้วย

นอกจากนี้ BIM ยังมีการสร้างความสัมพันธ์ด้านตัวแปร (Parameter) ระหว่างองค์ประกอบในแบบจำลองอาคาร ทำให้สามารถเปลี่ยนแปลงขนาดและระยะของงานออกแบบได้สะดวกและรวดเร็วมากยิ่งขึ้น

3. BIM กับมาตรฐานข้อมูลกลาง (Industry Foundation Classes; IFC)

ในปัจจุบันมีโปรแกรมสำหรับ BIM อยู่หลายผลิตภัณฑ์ ตัวอย่างเช่น

ก. Autodesk Revit โดยบริษัท Autodesk
ข. ArchiCAD โดยบริษัท Graphicsoft
ค. Vectorworks Allplan และ Architecture โดยบริษัท Nemetschek
ง. AECOsim Builiding Designer โดยบริษัท Nemetschek
จ. Tekla Structure โดยบริษัท Tekla
ฉ. Solibri Model Checker และ Solibri Model Viewer โดยบริษัท Solibri

โดยซอฟท์แวร์แต่ละตัวจะมีลักษณะของการเก็บข้อมูลที่ต่างกันและไม่สามารถแลกเปลี่ยนข้อมูลกันได้อย่างสมบูรณ์แบบ จึงเกิดแนวคิดในการกำหนดมาตรฐานข้อมูลกลางเพื่อใช้ในการแลกเปลี่ยนข้อมูลระหว่างซอฟท์แวร์ BIM โดยหน่วยงานชื่อ buildingSMART (เดิมคือ IAI หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

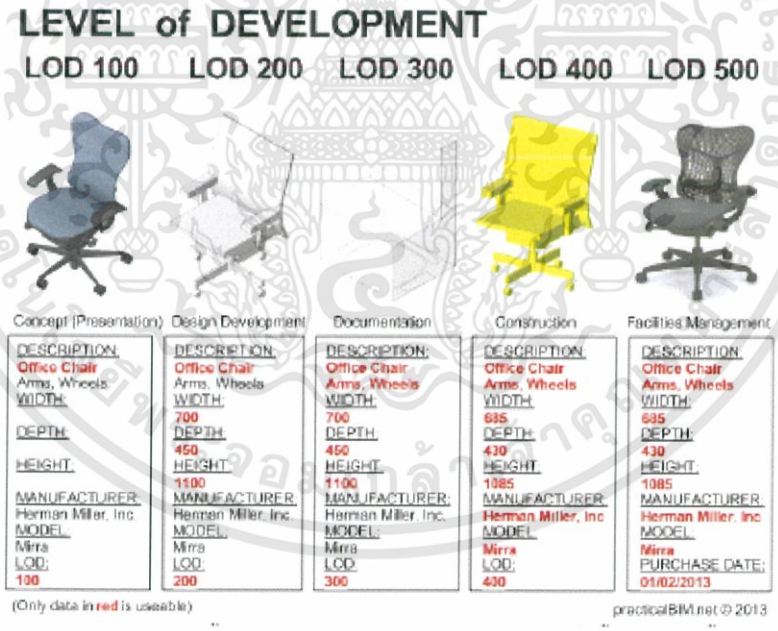
International Alliance for Interoperability) ได้พัฒนาฐานข้อมูลกลางขึ้นที่มีชื่อเรียกว่า IFC หรือ Industry Foundation Classes ขึ้น อย่างไรก็ตามมาตรฐานดังกล่าวยังคงอยู่ในระหว่างการพัฒนา ปัจจุบันจึงมีซอฟท์แวร์เพียงแค่ไม่กี่ตัวเท่านั้นที่สามารถนำเข้าและส่งไฟล์เข้า IFC ได้

4. การกำหนดมาตรฐาน BIM

การทำงานออกแบบสถาปัตยกรรมในแต่ละขั้นตอน ตั้งแต่การทำแบบร่างอาคารไปจนถึงการทำแบบรายละเอียดอาคาร จะมีความต้องการของข้อมูลและรายละเอียดที่แตกต่างกัน ดังนั้นการนำ BIM ไปใช้งานจึงต้องกำหนดมาตรฐานในการสร้างแบบจำลองให้สอดคล้องกับขั้นตอนในการทำงานด้วย ทั้งนี้มาตรฐานดังกล่าวประกอบด้วย

ก. มาตรฐานรายละเอียดขององค์ประกอบอาคารและแบบจำลองอาคาร

มาตรฐานนี้มีชื่อเรียกว่า Level of Development (LOD) หมายถึงการกำหนดระดับรายละเอียดขององค์ประกอบอาคารและแบบจำลองอาคาร รวมทั้งข้อมูลสารสนเทศประกอบให้สอดคล้องกับการทำงานในขั้นตอนในงานออกแบบสถาปัตยกรรมนั่นเอง มาตรฐานนี้อาจจะกำหนดเป็นค่าตัวเลข เช่น LOD100 LOD200 ดังรูปที่ 2.2 หรืออาจจะเป็นขั้นตอนในการทำงาน เช่น ขั้นตอนแนวความคิดในการออกแบบ (Conceptual Design) ขั้นตอนการพัฒนาแบบ (Design Development) เป็นต้น ทั้งนี้ขึ้นอยู่กับวัตถุประสงค์ของการนำ BIM ไปใช้ในการออกแบบสถาปัตยกรรมและเพื่อให้เกิดความเข้าใจกับทุกฝ่ายที่เกี่ยวข้อง



รูปที่ 2.2 แสดงตัวอย่างของเก้าอี้โดยที่ข้อมูลที่ไม่ใช่กราฟิก (Non-graphics) เพิ่มขึ้นตามระดับชั้นของโครงการ

ข. มาตรฐานการทำงานร่วมกันและการแบ่งปันข้อมูล

ในกรณีที่งานมีความซับซ้อนและมีผู้ทำงานอยู่หลายคน จำเป็นที่จะต้องกำหนดมาตรฐานนี้เพื่อให้เกิดประสิทธิภาพในการทำงานโดยมาตรฐานนี้จะเกี่ยวข้องกับการกำหนดผู้ใช้ว่าสามารถทำงานในส่วนใดของแบบจำลองอาคารได้บ้าง

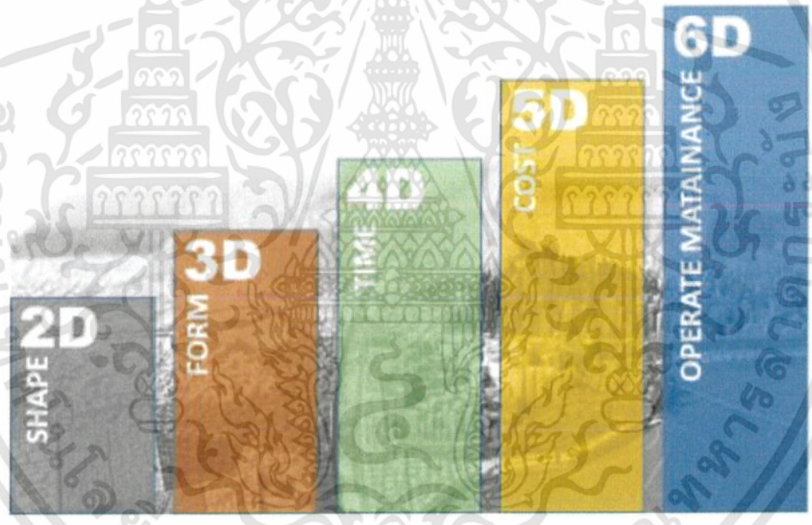
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค. มาตรฐานการกำหนดมหมวดหมู่และประเภทของข้อมูลองค์ประกอบอาคาร

ในกรณีที่มีการสร้างองค์ประกอบอาคารขึ้นมาเอง ประกอบไปด้วยมาตรฐานการตั้งชื่อรายละเอียดข้อมูลสารสนเทศและตัวแปรที่เกี่ยวข้อง

5. ความแตกต่างระหว่าง 2 มิติ 3 มิติ 4 มิติ 5 มิติ และ 6 มิติ

จากการทำงานในลักษณะของ 2 มิติและ 3 มิติ ในลักษณะเดิมนั้น จะมีการเพิ่มเติมมิติของเวลา (มิติที่ 4) ดังรูปที่ 2.3 ซึ่งจะใช้ในการะบวนการก่อสร้างที่จะต้องมีการวางแผนและควบคุมระยะเวลาในการดำเนินการ การเพิ่มในส่วนของราคาค่าใช้จ่าย (มิติที่ 5) เข้ามาเพื่อให้มิติของแบบก่อสร้าง เวลาและค่าใช้จ่ายนั้นมีความสัมพันธ์กัน และเมื่ออาคารได้มีการสร้างขึ้นเสร็จเรียบร้อยแล้ว แบบจำลอง BIM นั้นจะถูกพัฒนาให้เหมือนกันกับอาคารที่ก่อสร้างขึ้น พร้อมกับข้อมูลส่วนประกอบของอาคารที่ได้ถูกใส่ลงในไปในแบบจำลองแล้ว ซึ่งแบบจำลองในลักษณะนี้จะถูกเรียกว่า 'As-Built BIM' โดยแบบจำลองนี้ก็จะถูกนำไปใช้สำหรับการบริหารและบำรุงรักษา (มิติที่ 6) ทำให้ครบวงจรกระบวนการตั้งแต่ขั้นตอนการออกแบบ การก่อสร้าง การดูแลอาคาร ไปจนถึงอาคารนั้นถูกรื้อทำลาย จาก 2 มิติไปจนถึง 6 มิตินั่นเอง



รูปที่ 2.3 แสดงลักษณะของมิติแต่ละมิติ

6. ความแตกต่างระหว่าง BIM และ CAD

Computer Aided Design (CAD) คือเทคโนโลยีทางคอมพิวเตอร์ที่นำมาช่วยในการออกแบบและเขียนแบบ โดยแบ่งออกเป็น CAD ในระบบ 2 มิติและ CAD ในระบบ 3 มิติ

CAD ในระบบ 2 มิติเป็นการใช้ซอฟท์แวร์คอมพิวเตอร์มาแทนที่วิธีการเขียนแบบดั้งเดิมที่ใช้กระดาษและอุปกรณ์การเขียนแบบ การใช้ซอฟท์แวร์ CAD ในระบบ 2 มิติ ยังคงใช้วิธีทำงานเลียนแบบการเขียนแบบเดิม เช่น ผังพื้น รูปด้าน รูปตัด และแบบขยาย เป็นต้น โดยใช้เครื่องมือกราฟิกแบบเรขาคณิตสำหรับการสร้างรูปอันประกอบไปด้วย เส้นตรง เส้นโค้ง วงกลม ตัวหนังสือ เป็นต้น ดังนั้นผลลัพธ์ที่ได้จากการใช้ CAD ในระบบ 2 มิติ จะเหมือนกับการเขียนแบบด้วยมือ เพียงแต่เปลี่ยนเครื่องมือในการเขียนเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CAD ในระบบ 3 มิติเป็นการสร้างรูปทรงที่มีลักษณะเป็น 3 มิติ โดยมีองค์ประกอบทางกราฟิกที่ซับซ้อนมากกว่ากราฟิก 2 มิติ ได้แก่ การประกอบกันของระนาบผิว (Surface) เป็นปริมาตร และวัตถุแบบทึบตัน (Solid) โดยการแสดงผลในมุมมองที่แตกต่างกัน รวมทั้งการใส่วัสดุและสร้างภาพที่มีความเสมือนจริง (Render) เพื่อใช้ในการนำเสนอที่ชัดเจนยิ่งขึ้น

อย่างไรก็ตาม CAD จะเป็นการสร้างรูปกราฟิกแบบเรขาคณิต ประกอบขึ้นเป็นตัวงานออกแบบสถาปัตยกรรม คำสั่งที่ใช้งานใน CAD จะเป็นคำสั่งในการสร้างรูปกราฟิกเพียงเท่านั้น เช่น การเขียนรูปเสาในผังพื้น เกิดจากการเขียนรูปสี่เหลี่ยมผืนผ้า หรือเสาในรูป 3 มิติ เกิดจากการสร้างจากกล่องสี่เหลี่ยม เท่านั้น

ในขณะที่ BIM จะใช้วิธีการสร้างวัตถุองค์ประกอบของอาคารเลียนแบบจริง เช่น มีคำสั่งพื้น ผนัง หลังคา เสา คาน ทั้งในรูป 2 มิติและ 3 มิติ มาประกอบกันเป็นตัวอาคารจำลอง โดยสามารถใส่ข้อมูลสารสนเทศของวัตถุองค์ประกอบเพิ่มเติมเข้าไปได้ ดังนั้นอาจจะกล่าวได้ว่า CAD เน้นไปที่การสร้างรูปและการเขียนรูปโดยใช้กราฟิก ส่วน BIM เน้นไปที่การสร้างแบบจำลองอาคาร

2.2 ระบบซอฟท์แวร์ที่เกี่ยวข้องในงานวิจัย

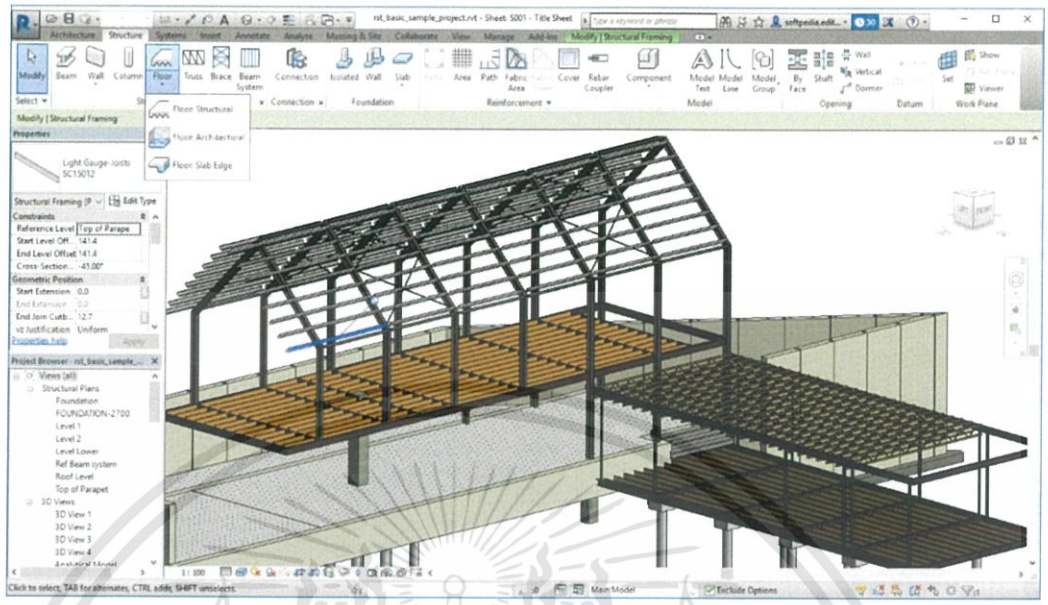
ในปัจจุบันซอฟท์แวร์ที่ใช้ในกระบวนการแบบจำลองสารสนเทศอาคารมีหลายซอฟท์แวร์ด้วยกัน แต่ผู้ศึกษาได้เลือกใช้ซอฟท์แวร์ดังต่อไปนี้เพื่อใช้ในการศึกษา

2.2.1 โปรแกรม Autodesk Revit

Autodesk Revit[2] คือโปรแกรมซอฟท์แวร์ BIM ของบริษัท Autodesk สำหรับสถาปนิค ภูมิสถาปัตย์ วิศวกรโครงสร้าง วิศวกรระบบอาคาร นักออกแบบและบุคคลที่มีส่วนร่วมในสายงานนี้ โปรแกรมนี้จะช่วยทำให้ผู้ใช้ออกแบบอาคารจำลองขึ้นมาได้ ไม่ว่าจะเป็นทั้งโครงสร้างอาคารและรายละเอียดของส่วนประกอบที่อยู่ข้างใน โดยผู้ใช้สามารถทำงานในทั้งมุมมอง 2 มิติและ 3 มิติ อีกทั้งยังสามารถเข้าถึงข้อมูลได้ตามที่ต้องการ เนื่องจากข้อมูลพวกนี้จะถูกเก็บไว้ในฐานรวมข้อมูลกลางทั้งหมด ทั้งนี้ Autodesk Revit เป็นซอฟท์แวร์ BIM ในรูปแบบ 3 มิติที่ประกอบไปด้วยเครื่องมือที่สามารถวางแผนและติดตามตรวจสอบขั้นตอนในวงจรชีวิตของการก่อสร้างได้ โดยเริ่มต้นจากการเขียนแบบของอาคารจนถึงการบำรุงรักษาอาคารภายหลังจากที่ก่อสร้างเสร็จและการรื้อถอนอาคาร

1. ตัวอย่างหลักการทำงานของโปรแกรม Autodesk Revit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



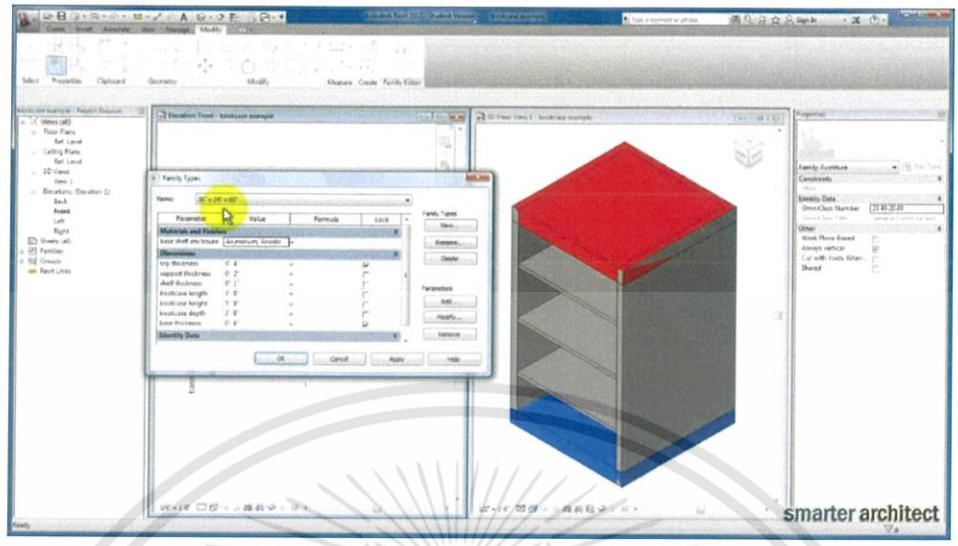
รูปที่ 2.4 แสดงลักษณะและเครื่องมือเบื้องต้นในโปรแกรม Autodesk Revit

จากรูปที่ 2.4 สามารถอธิบายลักษณะของการทำงานและเครื่องมือเบื้องต้นของโปรแกรม Autodesk Revit ได้ดังนี้

ก. การออกแบบ

ผู้ใช้สามารถปรับเปลี่ยน แก้ไขและประกอบแบบจำลอง 3 มิติในพื้นที่ทำงานของโปรแกรมได้ โดยเครื่องมือที่ใช้ในการออกแบบนั้นสามารถใช้ร่วมกับวัตถุกราฟิกหรือแบบจำลอง 3 มิติที่ไม่ได้สร้างขึ้นในโปรแกรม Autodesk Revit ที่โหลดนำเข้าไปในโปรแกรมได้ แต่โปรแกรม Autodesk Revit นั้นไม่ใช่โปรแกรมแบบ NURMS (Nonuniform Rational B-Spline หรือโมเดลที่มาจากการคำนวณทางคณิตศาสตร์ที่มักจะนิยมใช้เพื่อสร้างและปรับเปลี่ยนจุดและพื้นผิวของแบบจำลองได้โดยตรง) ทำให้โปรแกรม Autodesk Revit มีข้อจำกัดในเรื่องการควบคุมรูปหลายเหลี่ยม (Polygon) ของวัตถุ ยกเว้นในบางกรณี เช่น หลังคา คาน เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงตัวอย่างของ family

ในโปรแกรม Autodesk Revit สามารถแบ่งประเภทของวัตถุได้เป็น 3 ประเภทใหญ่ๆ ซึ่งวัตถุเหล่านี้ใน Autodesk Revit จะเรียกว่า family ดังแสดงในรูปที่ 2.5

1) System Families เป็นองค์ประกอบพื้นฐานที่ถูกรวบรวมมาจากไซต์ก่อสร้าง เช่น ผนัง กำแพง พื้น ท่อ เป็นต้น
2) Loadable Families/Components วัตถุที่อยู่ใน Family นี้จะถูกสร้างขึ้น นำเข้าหรือดาวน์โหลดจากไฟล์ RFA จากภายนอก สำหรับ Loadable Families จะประกอบไปด้วยหมวดหมู่แค็ตตาล็อคของวัตถุที่ผู้ใช้สามารถสร้างและใช้ได้ โดยทำการดาวน์โหลดเพียงแค่ 1 ชนิดที่ต้องการสำหรับใช้ในโครงการ
3) In-Place Families วัตถุที่อยู่ใน Family นี้เป็นวัตถุที่มีองค์ประกอบเฉพาะ ผู้ใช้สร้างรูปทรงเรขาคณิตชนิดนี้เพื่ออ้างอิง ปรับขนาดหรือปรับสิ่งอื่นที่อยู่ในโครงการเมื่อมีการเปลี่ยนแปลงรูปทรงเรขาคณิตอ้างอิง องค์ประกอบประเภทนี้จะมีเครื่องมือในการสร้างเช่นเดียวกับการสร้าง Loadable Families/Components

ผู้ใช้สามารถสร้าง Families ที่มีความสมจริงได้ ไม่ว่าจะเป็นเฟอร์นิเจอร์ไปจนถึงเครื่องใช้ไฟฟ้าจำพวกหลอดไฟ หรือนำเข้ามาจากโปรแกรมอื่น ทั้งนี้ Families สามารถสร้างเป็น Parametic Model ที่มีทั้งมิติและคุณสมบัติ สองสิ่งนี้จะทำให้ผู้ใช้สามารถปรับเปลี่ยนพารามิเตอร์บางตัวเพื่อปรับให้ได้ Families ตามต้องการได้ อีกทั้ง Families ที่มีพารามิเตอร์ที่เฉพาะตัวนั้นสามารถบันทึกเป็น Type หรือหมวดหมู่ได้ เช่น หน้าต่างสองบานที่มีขนาดแตกต่างกันหรือมีสีที่แตกต่างกัน เป็นต้น

ข. Interoperability and IFC

หลังจากที่ผู้ใช้สร้างโมเดลอาคารหรือ Families ในโปรแกรม Autodesk Revit แล้ว ผู้ใช้สามารถสร้างภาพเสมือนจริงโดยการใช้เครื่องมือ Rendering ใน Autodesk Revit ได้โดยการที่ผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อาจจะต้องกำหนดวัตถุของส่วนประกอบนั้นขึ้นมาเองให้มีสีและขนาดตามที่ต้องการหรืออาจจะเป็นวัตถุส่วนประกอบที่มีอยู่ในโปรแกรมอยู่แล้ว

ค. การทำงานแบบ Collaborate

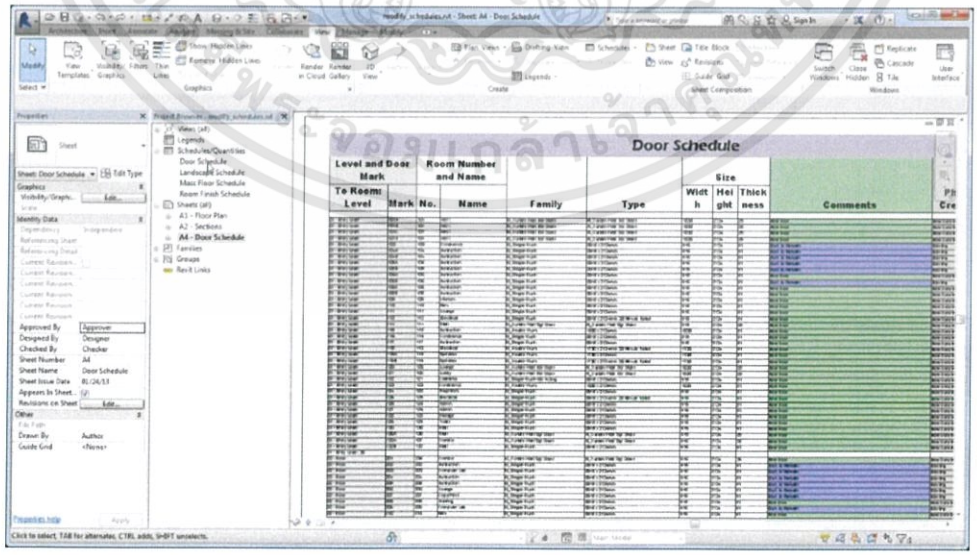
การที่ผู้ร่วมงานในโครงการหนึ่งนั้นสามารถเข้าถึงข้อมูลของโมเดลได้จากฐานข้อมูลกลางของโปรแกรม Autodesk Revit เป็นผลทำให้ผู้ใช้แต่ละคนสามารถลดอัตราการชนกันของงานของแต่ละบุคคลและไม่จำเป็นที่จะต้องเสียเวลาในการทำงานใหม่อีกครั้ง อีกทั้งผู้ใช้แต่ละบุคคลสามารถบันทึกและอัพเดทชิ้นงานได้ตลอดเวลา

2. ตัวอย่างเครื่องมือ

เครื่องมือพื้นฐานในโปรแกรม Autodesk Revit นั้นค่อนข้างจะมีความคล้ายคลึงกับเครื่องมือในโปรแกรม AutoCAD ไม่ว่าจะเป็นเครื่องมือ Move Copy Rotate Mirror Annotate เป็นต้น แต่ส่วนที่เพิ่มเติมเข้ามาคือเครื่องมือที่อำนวยความสะดวกในการสร้างและออกแบบอาคารให้มีความสมจริงมากยิ่งขึ้น เครื่องมือพวกนี้จะใช้ในการสร้าง Architecture และ Structure ของอาคาร เช่น Wall Roof Slab Floor เป็นต้น ต่อไปนักศึกษาจะขอทำอธิบายเครื่องมือที่ปรากฏอยู่เนื้อหาภายในรายงานเล่มนี้

ก. Schedule[3]

Schedule เป็นบัญชีแสดงองค์ประกอบและคุณลักษณะขององค์ประกอบที่อยู่ภายในแบบจำลอง เครื่องมือนี้สามารถลงรายการขององค์ประกอบได้ เช่น ผนัง ประตู หน้าต่าง หรือการคำนวนปริมาณพื้นที่และปริมาตรในพื้นที่หนึ่งๆ ดังรูปที่ 2.6 และสามารถลงรายการเรียงลำดับหน้าของ Sheet ได้เช่นกัน ซึ่งการใช้ Schedule เป็นอีกหนึ่งหนทางที่เราสามารถติดตามข้อมูลภายในแบบจำลองของเราได้ เนื่องจากเมื่อสร้าง Schedule ขึ้นมา ข้อมูลที่ปรากฏคือข้อมูลขององค์ประกอบ เมื่อเราทำการปรับเปลี่ยนคุณสมบัติของตัวองค์ประกอบ ค่าที่อยู่ใน Schedule ย่อมเปลี่ยนไปด้วยเช่นกัน



รูปที่ 2.6 แสดงตัวอย่างของ Schedule ในโปรแกรม Autodesk Revit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถแบ่งประเภทของ Schedule ได้เป็น 6 ประเภทหลัก ได้แก่

1) Schedule/Quantities

เป็นประเภทของ Schedule ที่นิยมใช้กันโดยทั่วไป Schedule ประเภทนี้สามารถลงรายการและแสดงปริมาณองค์ประกอบแต่ละหมวดหมู่ได้ ตัวอย่างเช่น Schedule ของผนัง Schedule ของพื้น เป็นต้น แต่จะมีความสามารถที่จำกัด ทำให้ลงรายการได้เพียงแค่หมวดหมู่เดียวเท่านั้น อย่างไรก็ตามผู้ใช้สามารถสร้าง Multicategory Schedule ขึ้นมาได้ หรือใช้ Field (หมวดหมู่ข้อมูลที่ผู้ใช้สามารถเลือกได้ว่าต้องการให้ข้อมูลชนิดไหน ซึ่งปรากฏอยู่บนหน้าต่าง Schedule Properties เช่น ประเภท หมายเลข พื้นที่ เป็นต้น

2) Graphical Column Schedule

Schedule ประเภทนี้ค่อนข้างแตกต่างออกไปจาก Schedule ประเภทอื่น นิยมใช้ทั่วไปในวิศวกรโครงสร้าง ซึ่งโครงสร้างจำพวกเสาจะแสดงออกมาใน Schedule ตามตำแหน่งที่ Grid Line (เส้นบอกตำแหน่งขององค์ประกอบ) ขององค์ประกอบนั้นตัดกัน และบอกถึงส่วนที่จำกัดทั้งด้านบนและด้านล่าง รวมไปถึงค่า offset (ระยะฐีมีการเผื่อไว้จากส่วนที่จำกัด) ด้วยเช่นกัน

3) Material Takeoff Schedule

Schedule ประเภทนี้จะลงรายการของวัสดุและองค์ประกอบย่อยในหมวดหมู่ของ Family หนึ่งที่ต้องการ ผู้ใช้สามารถใช้ Schedule ประเภทนี้ในการระบุวัสดุที่ถูกใช้ในองค์ประกอบหรือเป็นส่วนประกอบของวัตถุใดวัตถุหนึ่ง ตัวอย่างเช่น การที่ผู้ใช้ต้องการทราบถึงปริมาตรของคอนกรีตที่ถูกใช้ทั้งหมดในโมเดลชิ้นงาน ไม่ว่าคอนกรีตนี้จะอยู่ในรูปแบบของเสา คาน หรือ พื้น ผู้ใช้สามารถใช้ Schedule ประเภทนี้ในการแสดงถึงปริมาณคอนกรีตทั้งหมดได้

4) Sheet list

ผู้ใช้สามารถลงรายการ Sheet (แผ่นงานที่ผู้ใช้สร้างขึ้น) ทั้งหมดในโปรเจคได้

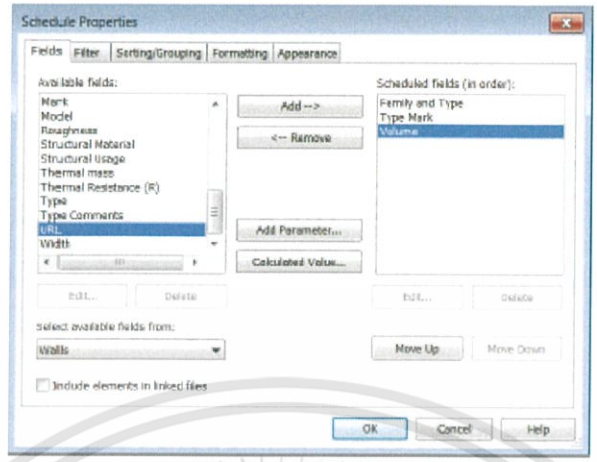
5) Note Block

เครื่องมือนี้จะทำให้ผู้ใช้สามารถสร้าง Schedule ที่สามารถลงรายการถึงสัญลักษณ์ Annotation (เครื่องมือที่ใช้ในการวัดระยะ) ที่ใช้ในโปรเจคได้

6) View list Schedule

Schedule ประเภทนี้สามารถแสดงรายการของ View และคุณสมบัติของมันโดยการสร้าง Schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



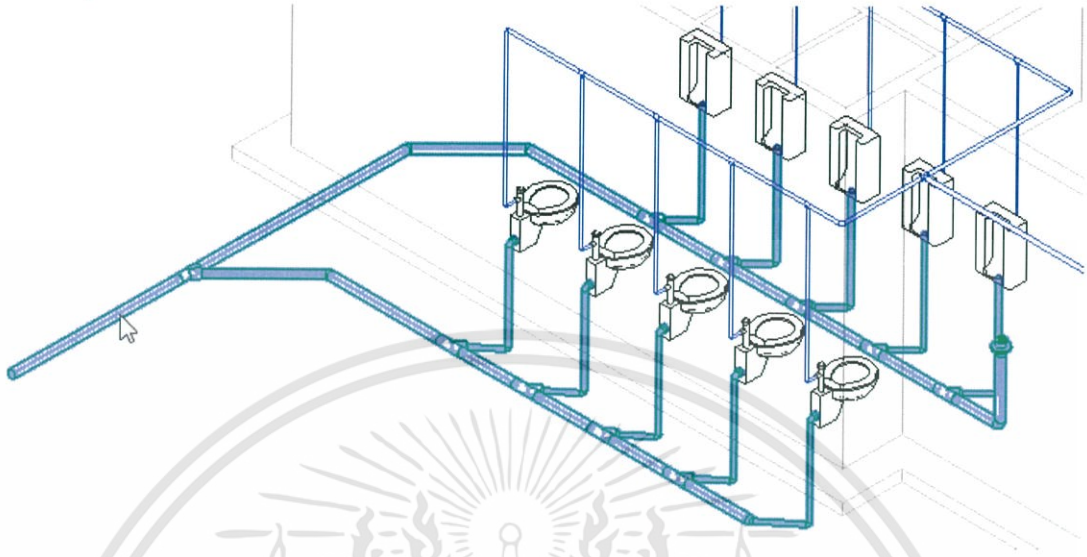
รูปที่ 2.7 แสดงหน้าต่างคำสั่งเพื่อกำหนด Properties ของ Schedule

ในหน้าต่าง Schedule Properties ดังแสดงในรูปที่ 2.7 มีแถบเครื่องมือปรากฏขึ้นมา ดังต่อไปนี้

1) Fields เป็นแถบเครื่องมือเพื่อให้ผู้ใช้เลือกหมวดหมู่ข้อมูลที่ต้องการให้ปรากฏอยู่ใน Schedule Properties คือคุณสมบัติที่เกี่ยวข้องกับผนัง (Field ในแต่ละหมวดหมู่จะไม่เหมือนกัน ทั้งนี้ขึ้นอยู่กับหมวดหมู่ที่ผู้ใช้เลือก) หากผู้ใช้ทำการเพิ่ม Parameter อื่นเข้าไปใน Parameter นั้นจะปรากฏอยู่ในแถบเครื่องมือนี้ด้วยเช่นกัน
2) Filter ในแถบเครื่องมือนี้ ผู้ใช้สามารถเลือกได้ว่าต้องการให้ข้อมูลใดแสดงบนหน้า Schedule
3) Sorting/Grouping แถบเครื่องมือนี้จะช่วยให้ผู้ใช้สามารถควบคุมลำดับของข้อมูลที่แสดงบน Schedule และ Parameter เป็นตัวควบคุมลำดับเหล่านั้น เช่น หากผู้ใช้สร้าง Sheet Index ขึ้นมา ผู้ใช้สามารถเลือก Sort โดยการใช้เลขลำดับของ Sheet หรือชื่อของ Sheet ตามที่ต้องการให้ข้อมูลนั้นแสดงบน Schedule
4) Formatting ผู้ใช้สามารถควบคุมลักษณะของ Field ที่แสดงบน Schedule ได้โดยการใช้แถบเครื่องมือนี้ ไม่ว่า Fields นั้นจะแสดงอยู่บน Schedule ในขณะนั้นด้วยหรือไม่ ผู้ใช้สามารถเปลี่ยนคุณลักษณะของแต่ละ Fields ได้ ไม่ว่าจะเป็น Alignment, heading, orientation ในส่วนหัวเรื่องได้ อีกทั้งยังสามารถเปลี่ยนหน่วยของการวัดระยะ เช่น เมื่อเริ่มโปรแกรม โปรแกรมจะถูกตั้งค่าโดยอัตโนมัติเพื่อแสดง Schedule ของผู้ใช้เป็นหน่วยฟุต ในที่นี้ผู้ใช้สามารถเปลี่ยนเป็นหน่วยเมตรหรือหน่วยเซนติเมตรได้ตามที่ต้องการ
5) Apperance แถบเครื่องมือนี้เป็นตัวควบคุมลักษณะทางกราฟิกของ Schedule เช่น การเปลี่ยนขนาดและฟอนต์ในแต่ละแถวและส่วนหัวเรื่องของ Schedule ได้ อีกทั้งยังสามารถปรับเส้น grid line ได้ (ในที่นี้คือเส้นตารางของ Schedule หรือการเลือกให้แสดงเส้น grid line นี้หรือไม่)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข. Pipe



รูปที่ 2.8 แสดงภาพตัวอย่างของระบบสุขาภิบาลและการประปา (Sanitary and Plumbing System) ในโปรแกรม Autodesk Revit

จากรูปที่ 2.8 Piping Systems[4] คือท่อในระบบที่วางเอาไว้เพื่อแสดงเส้นทางและส่วนที่เชื่อมต่อกันระหว่างสุขภัณฑ์ในระบบสุขาภิบาล เมื่อผู้ใช้ได้วางสุขภัณฑ์เหล่านี้ลงในงาน ผู้ใช้สามารถสร้างระบบน้ำประปา (Domestic Cold System) ระบบน้ำร้อน (Domestic Hot System) และระบบสุขาภิบาล (Sanitary System) เพื่อเชื่อมต่อสุขภัณฑ์เหล่านี้ได้

ระบบท่อสามารถสร้างขึ้นโดยใช้วิธีดังต่อไปนี้[5]

1) เมื่อผู้ใช้วางองค์ประกอบสุขภัณฑ์ลงในโครงการ องค์ประกอบเหล่านี้จะยังไม่ได้ถูกกำหนดระบบ แต่ในทางกลับกัน ถ้าเชื่อมต่อท่อหรือองค์ประกอบที่มีการตั้งค่าระบบไว้แล้ว องค์ประกอบสุขภัณฑ์เหล่านี้จะถูกตั้งค่าระบบให้เหมือนกับท่อหรือองค์ประกอบอื่นที่มาเชื่อมต่อกับมัน
2) ผู้ใช้สามารถเลือกองค์ประกอบนั้นโดยตรงและทำการใส่ระบบลงไปในหน้าต่าง Properties หลังจากที่องค์ประกอบนี้ถูกกำหนดระบบแล้ว ผู้ใช้สามารถใช้เครื่องมือ Generate Layout Workflow เพื่อสร้างเส้นทาง (Routing) ของท่อได้ และทำการสร้างท่อขึ้นมาได้อย่างอัตโนมัติโดยไม่จำเป็นต้องใช้คำสั่ง Pipe เพื่อทำการวาดท่อ

ตัวอย่างคำสั่ง[6] ที่สามารถใช้ร่วมกับเครื่องมือ Pipe

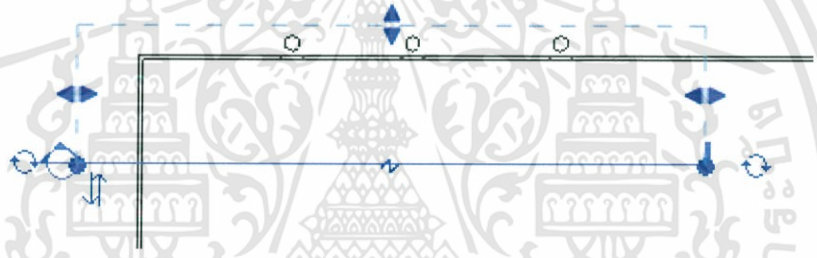
1) Slope off, Slope up และ Slope down เป็นการวาดท่อให้มีความชันที่ต้องการ
2) Automatically Connect เป็นคำสั่งที่ช่วยให้ท่อสองท่อเชื่อมต่อโดยอัตโนมัติ ซึ่งคำสั่งนี้มีประโยชน์เป็นอย่างมากเมื่อทำการเชื่อมต่อท่อในระดับที่ต่างกัน แต่มีข้อควรระวังในการเชื่อมต่อของท่อที่ไม่ได้เกี่ยวข้องกับระบบที่ผู้ใช้กำลังออกแบบ

คุณสมบัติของท่อใน Properties browser มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) Constraints เช่น Horizontal Justification, Vertical Justification, Reference Level, Offset และ Slope
2) Dimensions เช่น Outside diameter, Inside diameter, Size และ Length
3) Mechanical เช่น System Classification, System Type, System Name, Pipe Segment, Diameter, Roughness, Material, Invert Elevation และ Area
4) Mechanical– Flow เช่น Flow, Reynold Number, Flow state, Friction Factor, Velocity, Friction และ Pressure Drop เป็นต้น
5) Identity Data เช่น Image, Comment และ Mark
6) Phasing เช่น Phasing Create และ Phase Demolished
7) Insulation เช่น Overall Size, Insulation Thickness และ Insulation Type

ค. Section views



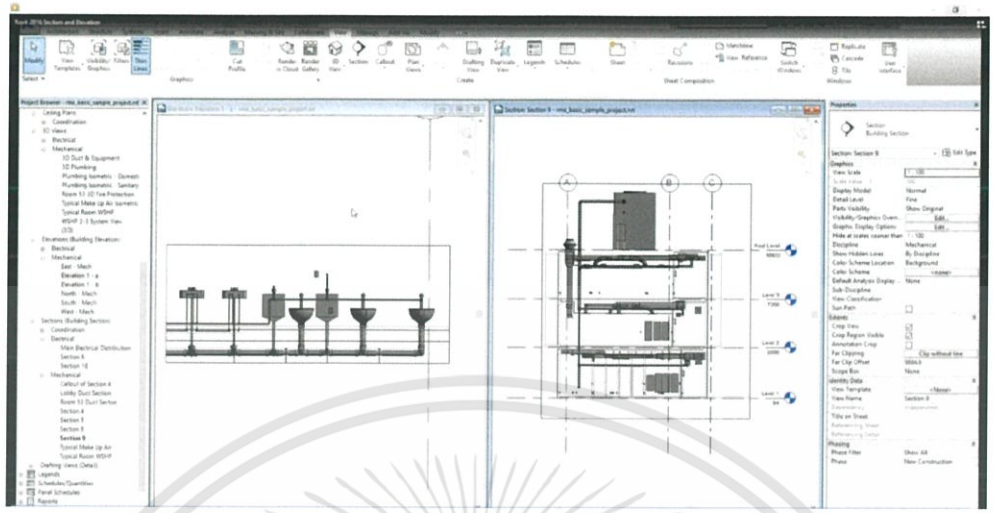
รูปที่ 2.9 แสดงสัญลักษณ์ของ Section ซึ่งสามารถแก้ไขได้โดยการยืดเข้า-ออกของลูกศรสีน้ำเงิน

โปรแกรม Autodesk Revit สามารถให้ผู้ใช้จัดการกับแบบจำลองทั้งรูปแบบ 2 มิติและ 3 มิติ นั่นหมายความว่าโปรแกรม Autodesk Revit ได้จัดเตรียมเครื่องมือที่แสดงมุมมอง (Views) เหล่านี้ไว้แล้ว ไม่ว่าจะเป็นมุมมองจาก Floor plan Elevations และ Sections นอกจากนี้ผู้ใช้ยังสามารถคัดลอกมุมมองและปรับเปลี่ยนได้ตามที่ต้องการอีกด้วย ดังแสดงในรูปที่ 2.9

สามารถแบ่งประเภทของมุมมอง[7] (View types) ได้ดังนี้

1) Plan View: Floor Views, Reflected Ceiling Plans และ Structural Plans
2) Elevation Views
3) Section Views
4) 3D Views
5) Schedules
6) Legends
7) Drafting Views

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 แสดงภาพมุมมองที่โปรแกรมแสดงขึ้นหลังจากที่ผู้ใช้ทำการตัด Section (ภาพขวา)

จากรูปที่ 2.10 เป็นภาพของมุมมองของชั้นระดับในแนวขวางที่แสดงขึ้นหลังจากใช้เครื่องมือ Section

ง. Grid Lines

รูปที่ 2.11 แสดงตัวอย่างของ Grid Lines

คำสั่ง Grid เป็นคำสั่งเครื่องมือที่ใช้วัดระยะ (Annotation) ที่วางคอลัมน์ Grid Lines ในชิ้นงานที่ผู้ใช้ปฏิบัติงานอยู่ ผู้ใช้สามารถวางอุปกรณ์ที่ต้องการลงไประหว่าง Grid Lines ได้โดยที่ Grid Lines ไม่จำกัดว่าผู้ใช้จะใช้งานแค่ในรูปแบบของเส้นตรงเพียงเท่านั้น เนื่องจากผู้ใช้สามารถปรับให้ Grid Lines เป็นเส้นโค้งได้เช่นเดียวกัน ทั้งนี้จะขึ้นอยู่กับชิ้นงาน ดังแสดงในรูปที่ 2.11 นอกจากนี้ผู้ใช้สามารถลบหรือซ่อน Grid Lines ได้หลังจากที่ใส่ Grid Lines ลงไปในชิ้นงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Autodesk Revit จะทำการใส่ตัวเลขหรือตัวอักษรโดยอัตโนมัติเมื่อผู้ใช้ทำการวาง Grid lines ลงไปในชิ้นงาน[8] ซึ่งผู้ใช้สามารถปรับเปลี่ยนเลขหรือตัวอักษรเหล่านี้ได้โดยการเลือกที่ตัวเลขเดิม จากนั้นจึงทำการกรอกตัวเลขหรือตัวอักษรใหม่ลงไปตามที่ต้องการ

แต่ในที่นี้ Grid lines ยังคงมีข้อจำกัดในการใช้งานคือ Grid lines ไม่สามารถแสดงบนภาพ 3 มิติได้ เว้นแต่ได้มีการเขียนโค้ดจากโปรแกรมอื่นเพิ่มเข้ามาเพื่อแก้ปัญหานี้[9]

จ. Lookup Table [10]

Lookup Table เป็นโครงสร้างข้อมูลประเภทหนึ่ง โดยปกติแล้วใช้เป็น Array เพื่อลดระยะเวลาในการทำงานหรือลดระยะเวลาในการคำนวณค่าต่าง ๆ ซึ่งระยะเวลาการทำงานที่ลดลงมาจากการที่โปรแกรมอ่านค่าได้จากตารางที่ถูกคำนวณหรือกรอกค่าไว้ในโปรแกรม Microsoft Excel เบื้องต้นแล้วโดยจะต้องบันทึกไฟล์ตารางจาก Microsoft Excel เป็นแบบ External Comma-Separated Values หรือไฟล์สกุล .csv

วิธีนี้ทำให้ผู้ศึกษาสามารถระบุขนาดของ Family ตัวใดตัวหนึ่งได้โดยที่ไม่จำเป็นต้องสร้าง Family ของแต่ละขนาดขึ้นมาอีกครั้งหนึ่งซึ่ง Family ที่กล่าวถึงนี้คือ Family ของ Pipe Fitting และ Pipe Accessory โดยโปรแกรม Autodesk Revit ได้เตรียมฟังก์ชั่น size_lookup ไว้สำหรับอ่านค่าจากไฟล์สกุล .csv นี้แล้ว

รูปแบบของของไวยากรณ์ที่ใช้ในการเขียน Lookup_table[11] เป็นดังนี้

result=size_lookup(LookupTableName, LookupColumn, DefaultIfNotFound, LookupValue1, LookupValue2, ..., LookupValueN) Table

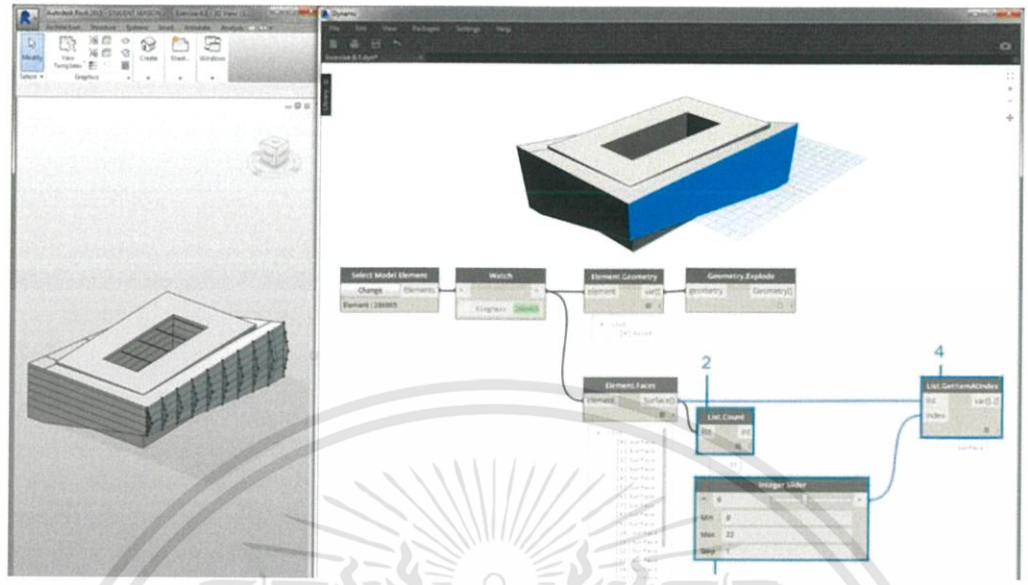
ตารางที่ 1 แสดงชื่อและหน้าที่ของคำสั่งในการเขียน Lookup Table

ชื่อ	หน้าที่
Result	ชื่อพารามิเตอร์ที่ต้องการส่งค่าคืน
LookupTableName	ชื่อของไฟล์สกุล .csv ที่ผู้ศึกษาต้องการให้โปรแกรมอ่านค่า
LookupColume	ชื่อของ colume ที่ผู้ศึกษาต้องการให้โปรแกรมอ่านค่า
DefaultNotFound	ค่าที่ถูกส่งกลับมาเมื่อโปรแกรมไม่สามารถอ่านค่า LookupValue ได้
LookupValue	ค่าที่ผู้ศึกษาต้องการให้โปรแกรมอ่านและส่งค่ากลับมาใน Revit

หมายเหตุ ฟังก์ชั่น size_lookup สามารถใช้ได้กับพารามิเตอร์แบบ instance และอ่านค่าที่เป็นตัวเลขเท่านั้น

2.2.2 ซอฟท์แวร์ Dynamo

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 แสดงลักษณะของซอฟท์แวร์ Dynamo

Dynamo เป็นซอฟท์แวร์ที่มีการเขียนโปรแกรมในรูปแบบ Vitual หรือแบบจินตภาพที่สามารถใช้คู่กับโปรแกรม Autodesk Revit และ Maya ได้ อีกทั้งยังสามารถใช้ระบบปฏิบัติการแบบเดี่ยวได้เช่นกัน โดยผู้สร้างมีจุดมุ่งหมายให้ทั้งผู้ที่เป็นนักเขียนโปรแกรมและไม่ใช่นักเขียนโปรแกรมสามารถใช้ซอฟท์แวร์ Dynamo นี้ได้ ซึ่ง Dynamo จะทำให้ผู้ใช้วางคำสั่งต่าง ๆ ลงไปในพื้นที่ทำงานและมีการเชื่อมต่อระหว่างคำสั่งกันได้เลย โดยไม่จำเป็นต้องไปเขียนโค้ดเป็นแบบตัวอักษร และผู้ใช้สามารถเห็นผลงานของตัวเองได้ในขณะที่กำลังสร้าง[12] ดังแสดงในรูปที่ 2.12

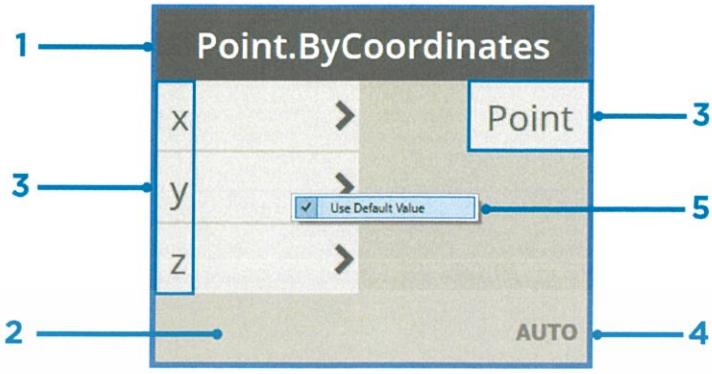
1. หลักการใช้ Dynamo

การเขียนโค้ดใน Dynamo สามารถทำได้โดยการเชื่อมคำสั่งแต่ละคำสั่ง ซึ่งในที่นี้จะเรียกว่าโหนด (Nodes) สิ่งที่ใช้ในการเชื่อมคำสั่งจะเรียกว่า Wires โดยที่ผู้ใช้จำเป็นต้องทราบถึงความสัมพันธ์ของโหนดแต่ละโหนด และทำการรัน (Run) โค้ดที่เขียนไว้ ซึ่งมี 2 แบบ นั่นก็คือการรันแบบอัตโนมัติและการรันแบบ Manual ซึ่งผู้ใช้จำเป็นจะต้องกดปุ่มนี้ทุกครั้งที่ต้องการรันโค้ด

ก. ส่วนประกอบของโหนด[13]

จากรูปที่ 2.13 แสดงให้เห็นถึงส่วนประกอบของโหนด ซึ่งประกอบไปด้วยทั้งหมด 5 ส่วน อาจจะมีข้อยกเว้นสำหรับ Input Node ซึ่งแต่ละส่วนสามารถอธิบายได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 แสดงส่วนประกอบของโหนด

1) Name แสดงชื่อและ Category ของโหนดนั้นๆ
2) Main แสดงส่วนประกอบหลักของโหนด
3) Port (In and Out) ตัวรับสายที่ส่งข้อมูลเข้าไปยังโหนดให้กลายเป็นผลลัพธ์ของการดำเนินการของโหนด
4) Lacing Icon ใช้สำหรับระบุตัวเลือกที่สอดคล้องกันกับรายการ input
5) Default Value เมื่อคลิกขวาที่ Input Port บางโหนดจะมีทั้งค่าเริ่มต้นที่สามารถใช้ได้หรือใช้ไม่ได้

ข. การเชื่อมซอฟท์แวร์ Dynamo เข้ากับโปรแกรม Revit

ผู้ใช้สามารถเขียนโค้ดสั่งให้โปรแกรม Revit ทำงานตามที่ต้องการได้ เนื่องจากในตอนแรกนั้นซอฟท์แวร์ Dynamo ถูกออกแบบให้แสดงรายการคำสั่งที่ทำในโปรแกรม Revit ทั้งหมดที่ผู้ใช้ที่ไม่ใช่คนพัฒนาเข้าถึงได้ยากมาก ฉะนั้นจึงเกิด Comprehensive API (Application Program Interface) เพื่อให้บุคคลภายนอกสามารถสร้างเครื่องมือมาใช้ร่วมด้วย ซึ่งอาจจะบอกได้ว่าซอฟท์แวร์ Dynamo เข้าไปปรับเปลี่ยนข้อมูลของโปรแกรม Revit ผ่านทาง Graphical Algorithm Editor.

2.2.3 ภาษา Python[14]

Python เป็นภาษาเขียนโปรแกรมที่สามารถใช้ได้อย่างมีประสิทธิภาพ ซึ่งถูกพัฒนาขึ้นโดยไม่ยึดติดกับแพล็ตฟอร์ม กล่าวคือสามารถรันภาษา Python ได้ทั้งบน Unix, Linux และ Windows อีกทั้งยังเป็นภาษาที่เป็น Open Source เหมือนกับ PHP ทำให้ผู้ใช้สามารถนำ Python มาพัฒนาโปรแกรมที่ต้องการได้โดยไม่เสียค่าใช้จ่ายใดๆ แต่ในขณะเดียวกันภาษา Python ไม่เหมาะสำหรับโปรแกรมที่มีข้อจำกัดความน่าเชื่อถือสูงหรือโปรแกรมที่มีการสร้างและบำรุงรักษาจากคนหลายคน เนื่องจากความอ่อนแอทางการตรวจสอบเชิงอรรถศาสตร์ อีกทั้งยังไม่เหมาะกับโปรแกรมที่ต้องมีการทำงานโดยใช้หน่วยความจำอย่างหนัก ซึ่งเป็นเหตุมาจากรูปแบบของโค้ดและภาษา Python ค่อนข้างใช้พื้นที่หน่วยความจำสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตามภาษา Python มีข้อได้เปรียบมากกว่าภาษาอื่นๆ เนื่องจากภาษา Python เป็นภาษาที่ง่ายต่อการเรียนรู้ไวยากรณ์ทางภาษา ดังนั้นจึงช่วยลดต้นทุนในการบำรุงรักษาโปรแกรม นอกจากนี้ภาษา Python ช่วยสนับสนุนโมดูลและแพ็คเกจที่เอื้อให้เกิดต้นแบบของโปรแกรมและนำโค้ดกลับมาใช้ใหม่ อีกทั้งยังเป็นโปรแกรมแปลคำสั่งที่มีอยู่ในแหล่งหรือรูปแบบไบนารีสำหรับแพล็ตฟอร์มหลักทั้งหมด

1. Python สำหรับ pyRevit

PyRevit[15] เป็นเครื่องมือที่เขียนมาจากภาษา Python API หลายๆ Scripts มารวมกันให้อยู่ในรูปแบบที่ใช้งานง่ายขึ้น อีกทั้ง pyRevit ยังมี IronPython script library สำหรับ Revit หรือเทียบได้ว่าเป็นคลังข้อมูลที่รวบรวม Script ที่ถูกสร้างขึ้นมาและพัฒนาให้เป็นเครื่องมือกับ Revit

เมื่อทำการดาวน์โหลดและติดตั้ง pyRevit แล้ว จากนั้นทำการเปิดโปรแกรม Autodesk Revit จะสังเกตเห็นแถบเครื่องมือใหม่ปรากฏขึ้นมาชื่อ pyRevit เพื่อเปิด Scripts ทั้งหมดจากแพ็คเกจทำให้รันโค้ดได้ง่ายโดยไม่ต้องดาวน์โหลดโค้ดผ่านRevitPythonShell หรือ IronPython Console อีกทั้งผู้ใช้สามารถเขียน Scripts เพิ่มเติมใน pyRevit ได้ด้วยการกดปุ่ม Reload โดยที่ไม่จำเป็นต้องปิดและผู้ใช้งานสามารถใช้งาน Add-ins และยังสามารถเขียน Script เพิ่มเติมใน pyRevit ได้โดยการกดที่ปุ่ม Reload เท่านั้น ซึ่งไม่จำเป็นต้องปิดและเปิดโปรแกรม Autodesk Revit ใหม่

Scripts ทั้งหมดใน pyRevit สามารถดูได้จากโฟล์เดอร์ชื่อ Extensions ซึ่งเป็น folder ที่สร้างขึ้นในขั้นตอนติดตั้ง Add-Ins อีกทั้งยังสามารถเปิดและแก้ไขได้ แต่ทางผู้ศึกษาจะไม่แก้ไข script ที่มีอยู่เดิม แต่จะสร้างเครื่องมือใหม่ขึ้นมา

ก. ตัวอย่างชุดคำสั่ง

1) .NET Framework class library[16]
เป็น library สำหรับเก็บรวบรวม Classes (ประเภทของข้อมูลที่สร้างโดยผู้ใช้งาน) Interface[17] (ข้อมูลทางทฤษฎีที่ถูกจัดเตรียมไว้โดย Classes) และข้อมูลประเภทอื่นที่สามารถทำให้ผู้ใช้มีสิทธิในการเข้าถึงฟังก์ชั่นของระบบได้

2) Clr.AddReference[18]
ในกรณีที่ผู้ใช้งานต้องการจะทำงานประสานกับ .NET libraries ผู้ชะจต้องดาวน์โหลด Assemblies[19] (หน่วยที่เล็กที่สุดของฟังก์ชั่นใน .NET libraries และมีนามสกุล .dll เสมอ) และใช้ Method AddReference (ฟังก์ชั่นของการทำงานที่กำหนดไว้ในคลาส) ในโมดูล clr (โมดูลคือไฟล์หรือส่วนของโปรแกรมที่ใช้สำหรับกำหนดตัวแปรฟังก์ชันหรือคลาสโดยแบ่งย่อยอีกหน่วยหนึ่งนโปรแกรมหลัก) เพื่อทำหน้าที่เชื่อมระหว่างทั้งสองระบบเข้าไว้

3) RevitAPI.dll (Revit Application Program Interface Assemblies) และ RevitAPIUI.dll (Revit Application Program Interface User Interface) [20]
Assemblies ทั้งสองชนิดนี้จำเป็นต้องระบุในการเขียนโค้ด เนื่องจากทั้งสอง Assemblies นี้มีหน้าที่ในการเชื่อมต่อระหว่าง Macro manager [21] ในทีนี้ Macro manager เป็นส่วนที่ประสานงานกับผู้ใช้เพื่อสั่งการ Revit Macro IDE ให้ทำการแก้ไขหรือสร้าง Macro[22] ซึ่งเป็นคำสั่งหรือฟังก์ชันที่ผู้ใช้เขียนขึ้นมาโดยการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Revit API[23] (ใช้ในการเชื่อมโยงกับโปรแกรมอื่นๆ เช่น Rise Etabs เป็นต้น เข้ากับโปรแกรม Autodesk Revit)

4) Autodesk.Revit.DB (Autodesk.Revit. Database) [24]
เป็นลิงก์ที่นำเข้า Classes ทั้งหมดใน Autodesk.Revit.DB Namespace ซึ่งเป็นที่อยู่ของ Class และ Method ส่วนใหญ่ที่ถูกใช้งานในโปรแกรม Autodesk Revit

5) Autodesk.Revit.UI (Autodesk.Revit. User Interface) [25]
เป็นลิงก์ที่นำเข้า Class ทั้งหมดใน Autodesk.Revit.UI Namespace ซึ่งเป็นที่อยู่ของ Class และ Method ที่เกี่ยวข้องกับการปรับเปลี่ยน Ribbon หรือ Workspace

6) ActiveUI Document [26]
เป็น Class UI Application ซึ่งเป็น Class ซึ่งถือสิทธิในการเข้าถึงการปรับเปลี่ยน User Interface หรือหมายถึงส่วนที่ต่อประสานงานใน Class, Method และไฟล์งานที่ถูกเปิดใช้งานอยู่

7) FilteredElementCollector [27]
เป็น Class ชนิดหนึ่งที่ทำหน้าที่ในการกรองและค้นหาข้อมูลใน element

8) BuildIncategory[28]
เป็น Class ชนิดหนึ่งซึ่งรวบรวม Built-In ทั้งหมดในโปรแกรม Autodesk Revit เช่น OST_CableTray,OST_PipeCurve เป็นต้น ซึ่งแบบจำลอง 3 มิติที่ถูกปรับเปลี่ยนแก้ไขได้เป็นเพราะเกิดการ Transaction ที่สมบูรณ์ หากเกิดการ Transaction ที่ไม่สมบูรณ์ จะทำให้ระบบไม่บันทึกการเปลี่ยนแปลงที่เกิดขึ้นกับแบบจำลอง 3 มิตินั้น

9) Transaction [29]
เป็น Class ที่ทำหน้าที่ในการควบคุมการเปลี่ยนแปลงทั้งหมดที่เกิดขึ้นกับวัตถุในชิ้นงาน ซึ่งหากเกิดการ Transaction ที่สมบูรณ์ วัตถุในชิ้นงานจะถูกปรับเปลี่ยนแก้ไขได้ แต่ถ้าเกิดการ Transaction ที่ไม่สมบูรณ์ วัตถุในชิ้นงานจะไม่ถูกบันทึกการเปลี่ยนแปลงที่เกิดขึ้น ในที่นี้คลาสนี้จะประกอบไปด้วย Method 4 ชนิด นั่นคือ
    1) Start ทำหน้าที่เริ่มการ Transaction
    2) Commit ทำหน้าที่ยุติการ Transaction และบันทึกการเปลี่ยนแปลงทั้งหมด
    3) Rollback ทำหน้าที่ยุติการ Transaction และละทิ้งการเปลี่ยนแปลงทั้งหมด
    4) GetStatus ทำหน้าที่คืนสถานะเดิมของวัตถุในชิ้นงาน

10) WhereElementIsNotElementType[30]
Method ที่ใช้เพื่อทำหน้าที่ในการกลับค่าของ Filter แบบนำ ElementsElement TypeFilter และส่งค่ากลับคืนสู่ Instance Collector (ตัวเก็บข้อมูล)

11) ToElements [31]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Method ชนิดหนึ่งที่ทำหน้าที่ในการส่งข้อมูลผ่าน Filter กลับไปยังพารามิเตอร์ที่ผู้ใช้กำหนดไว้

12) LookupParameter [32]
Method ชนิดหนึ่งที่ทำหน้าที่ในการเรียกพารามิเตอร์ที่มีชื่อตรงตามที่ระบุไว้

13) AsDouble [33]
เมื่อเขียนโค้ดนี้ลงในภาษา Python แล้วค่าที่ส่งกลับมาจะเป็นข้อมูลประเภทตัวเลขที่มีทศนิยม

14) As.ElementId [34]
เมื่อเขียนโค้ดนี้ลงในภาษา Python แล้วค่าที่ส่งกลับมาจะเป็น ID ของ Element ตามที่ต้องการ

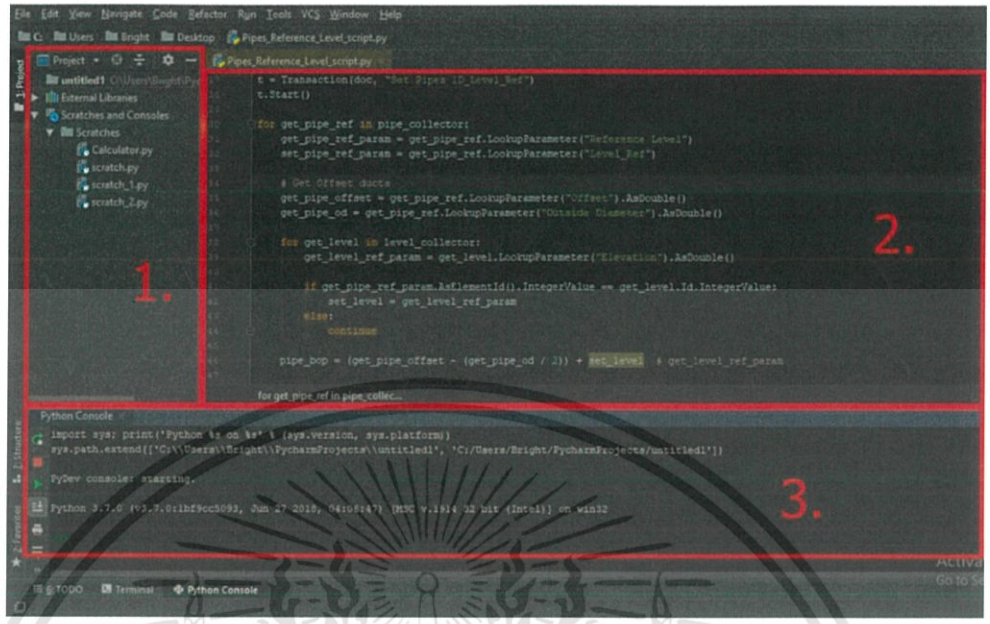
15) IntergerValue [35]
เมื่อเขียนโค้ดนี้ลงในภาษา Python แล้วค่าที่ส่งกลับมาจะเป็นข้อมูลประเภทตัวเลขจำนวนเต็ม

16) AsValueString [36]
เป็น Method ในคลาสพารามิเตอร์ใช้กับ Integer และ Double เพื่อส่งและเปลี่ยนแปลงค่าที่ได้จากพารามิเตอร์กลับมาจากตัวเลขให้กลายเป็นตัวอีกษร

2.3 Pycharm [37]

Pycharm เป็นซอฟท์แวร์ที่ถูกพัฒนาขึ้นเพื่อการฝึกหัดเขียนภาษา Python อีกทั้งยังเป็นโปรแกรมที่ใช้งานได้ง่ายและสามารถใช้งานได้โดยไม่เสียค่าใช้จ่าย นอกจากนี้ยังรองรับระบบปฏิบัติการ Windows, Linux และ Mac OS โดยสามารถดาวน์โหลดได้จาก https://www.jetbrains.com/pycharm/download และเลือกหมวดในการดาวน์โหลดเป็นหมวด Community ทั้งนี้ Pycharm ถือว่าเป็น IDE (Integrated Development Environment) สำหรับภาษา Python ตัวหนึ่ง ซึ่ง IDE[38] เป็นโปรแกรมที่รวบรวมคำสั่ง เมนู และ GUI (Graphical User Interface) มาสร้างเป็นโปรแกรมที่เหมาะแก่การเขียนโปรแกรม หรืออาจจะกล่าวได้ว่าเป็นโปรแกรมที่สร้างสถาพแวดล้อม (Environment) ให้เหมาะแก่การเขียนโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 แสดงหน้าต่างการทำงานของ Python ใน Pycharm

จากรูปที่ 2.14 ส่วนประกอบของหน้าต่างการทำงานของ Python ใน Pycharm มีดังนี้

หมายเลข 1 แสดงส่วนของบทเรียน โดยปกติจะเป็น Project Structure ของชิ้นงานนั้นๆ
หมายเลข 2 แสดง Workspace ซึ่งเป็นส่วนของโปรแกรมที่เราต้องเขียน (ไฟล์นามสกุล .py)
หมายเลข 3 แสดง Python Console เป็นส่วนของการรันคำสั่งหรือแสดงผลลัพธ์ต่างๆของโปรแกรม ซึ่งสามารถพิมพ์คำสั่งต่างๆบน Python Console ได้เลย

2.3.1 ลักษณะเด่นของ Pycharm[39]

1. มีระบบที่ช่วยในการเขียนโค้ด กล่าวคือสามารถตรวจสอบและแก้ไขจุดที่ผิดพลาดระหว่างการปฏิบัติงาน
2. มีเครื่องมือที่หลากหลาย เช่น Integrated Debugger, Test Runner, Python Profiler เป็นต้น ทำให้การทำงานเป็นไปได้อย่างรวดเร็วและสะดวกสบายมากยิ่งขึ้น
3. สามารถใช้ได้กับทุกแพล็ตฟอร์ม ไม่ว่าจะเป็น Windows, Mac OS และ Linux อีกทั้งยังมี Customizable UI สามารถปรับ Workspace ได้ตามที่ต้องการ

2.4 ทฤษฎีที่เกี่ยวข้องงานวิจัย

ในโครงงานเล่มนี้ สมการและทฤษฎีดังต่อไปนี้จะถูกใช้ในการคำนวณหาค่าต่างๆ เพื่อใช้แก้ปัญหาที่พบในระหว่างการปฏิบัติด้วยกระบวนการแบบจำลองสารสนเทศอาคาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 สมการการคำนวณค่า Head loss สำหรับวาล์วและข้อต่อของท่อจากสมการการคำนวณของ Hazen and Williams

เนื่องจากสมการของ Darcy-Weisbach ยากต่อการใช้งาน เนื่องจากค่าสัมประสิทธิ์แรงเสียดทาน (Friction) เป็นเรื่องยากในการคำนวณออกมาได้อย่างถูกต้อง Hazen and Williams จึงกำหนดสูตรเชิงประจักษ์ (Empirical Formula)[40] ที่ใช้ง่ายต่อการใช้งาน โดยมีการเชื่อมความสัมพันธ์ระหว่างอัตราการไหลของน้ำในท่อกับคุณสมบัติของท่อและความดันตกที่เกิดขึ้นจากแรงเสียดทาน สมการนี้จะนิยมใช้ในการออกแบบระบบท่อน้ำต่างๆ เช่น ระบบหัวกระจายน้ำดับเพลิงอัตโนมัติ ระบบน้ำประปา ระบบการชลประทาน เป็นต้น

ข้อดีของสมการ Hazen–Williams คือการที่ค่าสัมประสิทธิ์ C ไม่ได้เป็นฟังก์ชั่นเรย์โนล์ดนัมเบอร์ แต่ในทางกลับกันสมการนี้ใช้ได้เฉพาะของไหลที่เป็นน้ำเท่านั้น อีกทั้งยังไม่คำนึงถึงอุณหภูมิและความหนืดของน้ำ

$$h_f = k_1 \frac{LQ^{1.852}}{C^{1.852} D^{4.871}} \qquad (2.1)$$

h_f คือการสูญเสียความดันเนื่องมาจากความฝืดในท่อ หน่วยเมตร (m) หรือฟุต (ft)
K_1 คือค่าคงที่ของการแปลงหน่วย
L คือความยาวของท่อ หน่วยเมตร (m) หรือฟุต (ft)
Q คืออัตราการไหลของน้ำ หน่วยลูกบาศก์เมตรต่อวินาที (m^3/s) หรือแกลลอนต่อชั่วโมง (gpm)
D คือขนาดเส้นผ่านศูนย์กลางของท่อไฮโดรลิค หน่วยเมตร (m) หรือนิ้ว (in.)
C คือค่าสัมประสิทธิ์ความขรุขระของผิวภายในท่อ

2.4.2 สมการการคำนวณค่าความดันตก (Pressure drop) ในท่อจากสมการการคำนวณของ Haaland

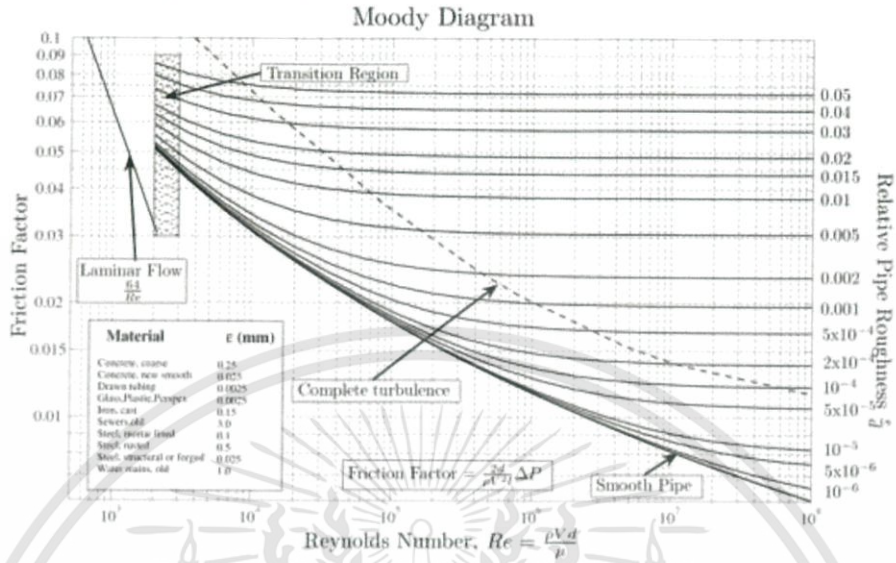
หลากหลายสมการถูกสร้างขึ้นมาเพื่อทำให้การหาค่าสัมประสิทธิ์ของแรงเสียดทานนั้นสามารถคำนวณได้โดยตรงในช่วงของค่าความขรุขระสัมพัทธ์ (Relative Roughness) และค่าตัวเลขเรย์โนลด์ (Reynold Number) ซึ่งสมการที่นิยมใช้ในปัจจุบันคือ Haaland's Equation คิดค้นโดย S.E. Haaland[41]

$$\frac{1}{\sqrt{f}} = -1.8 \log_{10}\left[\left(\frac{\varepsilon/D}{3.7}\right)^{1.11} + \frac{6.9}{Re}\right] \qquad (2.2)$$

f คือสัมประสิทธิ์แรงเสียดทาน (Darcy friction factor)
ε/D คือค่าความขรุขระสัมพัทธ์ (Relative Roughness)
Re คือค่าตัวเลขเรย์โนลด์ (Reynold Number)
D คือเส้นผ่านศูนย์กลางรอบนอกของท่อ (Outside Diameter)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 The Moody chart [42]



รูปที่ 2.15 แสดงตัวอย่างแผนภาพ Moody chart

Moody Chart หรือ Moody Diagram เป็นเส้นกราฟที่แสดงถึงความสัมพันธ์ระหว่างกลุ่มตัวแปรไร้มิติได้แก่ ค่าสัมประสิทธิ์แรงเสียดทาน (Darcy-Weisbach friction factor), ค่าตัวเลขเรย์โนลด์ (Reynold Number) และความขรุขระของพื้นผิว (Surface roughness) สำหรับของไหลที่มีการพัฒนาเต็มที่ในท่อทรงกลม แผนภาพนี้สามารถคาดการณ์ถึงความดันตกและอัตราการไหลรวมถึงสามารถหาตัวแปรที่ไม่ทราบค่าที่เหลือได้เช่นกัน ถ้าทราบถึงตัวแปรอีกสองค่าในเส้นกราฟ

2.4.4 Equivalent Length of Straight for valves and fitting

Equivalent Length of Straight Pipe for Valves and Fittings (m)												
Screwed Fittings		Pipe Size										
		1/4	3/8	1/2	3/4	1	1 1/4	1 1/2	2	2 1/2	3	4
Elbows	Regular 90 deg	0.7	0.9	1.1	1.3	1.6	2.0	2.3	2.6	2.8	3.4	4.0
	Long radius 90 deg	0.5	0.6	0.7	0.7	0.8	1.0	1.0	1.1	1.1	1.2	1.4
	Regular 45 deg	0.1	0.2	0.2	0.3	0.4	0.5	0.6	0.8	1.0	1.2	1.7
Tees	Line flow	0.2	0.4	0.5	0.7	1.0	1.4	1.7	2.3	2.8	3.7	5.2
	Branch flow	0.7	1.1	1.3	1.6	2.0	2.7	3.0	3.7	4.0	5.2	6.4
Return Bends	Regular 180 deg	0.7	0.9	1.1	1.3	1.6	2.0	2.3	2.6	2.8	3.4	4.0
Valves	Globe	6.4	6.7	6.7	7.3	8.8	11.3	12.8	16.5	18.9	24.1	33.6
	Gate	0.1	0.1	0.2	0.2	0.3	0.3	0.4	0.5	0.5	0.6	0.8
	Angle	3.9	4.6	4.6	4.6	5.2	5.5	5.5	5.5	5.5	5.5	5.5
	Swing Check	2.2	2.2	2.4	2.7	3.4	4.0	4.6	5.8	6.7	8.2	11.6
Strainer			1.4	1.5	2.0	2.3	5.5	6.1	8.2	8.8	10.4	12.8

engineeringtoolbox.com

รูปที่ 2.16 แสดงภาพตาราง Equivalent Length ของข้อต่อท่อเพื่อใช้ในการคำนวนความดันตก

ค่า Equivalent Length ของข้อต่อท่อนั้นคือความยาวของท่อตรงที่มีขนาดเท่ากันกับขนาดของข้อต่อท่อที่จะทำให้เกิดความดันตกเช่นเดียวกันกับที่เกิดขึ้นในข้อต่อท่อ[43] ซึ่งความดันตกในข้อต่อท่อนั้นจะขึ้นอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับการเปลี่ยนแปลงของความเร็วและทิศทางที่เปลี่ยนแปลงไปของของไหลที่เกิดขึ้นมาจากลักษณะทางกายภาพของข้อต่อท่อ ในทางกลับกันแรงเสียดทานระหว่างของไหลและผิวของข้อต่อท่อนั้นถือว่ามีผลกระทบ

2.4.5 สมการการคำนวณหาขนาดของปั๊มน้ำ [44]

หลังจากที่คำนวณค่า Head Loss ของระบบทั้งหมดแล้ว สามารถใช้สมการนี้ในการคำนวณพลังงานของปั๊มหอยโข่ง หลังจากนั้นจึงนำค่าที่ได้ประมาณค่าหาขนาดของปั๊มน้ำที่เหมาะสม

$$P = \frac{Q\rho gH}{3.6 \times 10^6 \eta} \qquad \text{(2.3)}$$

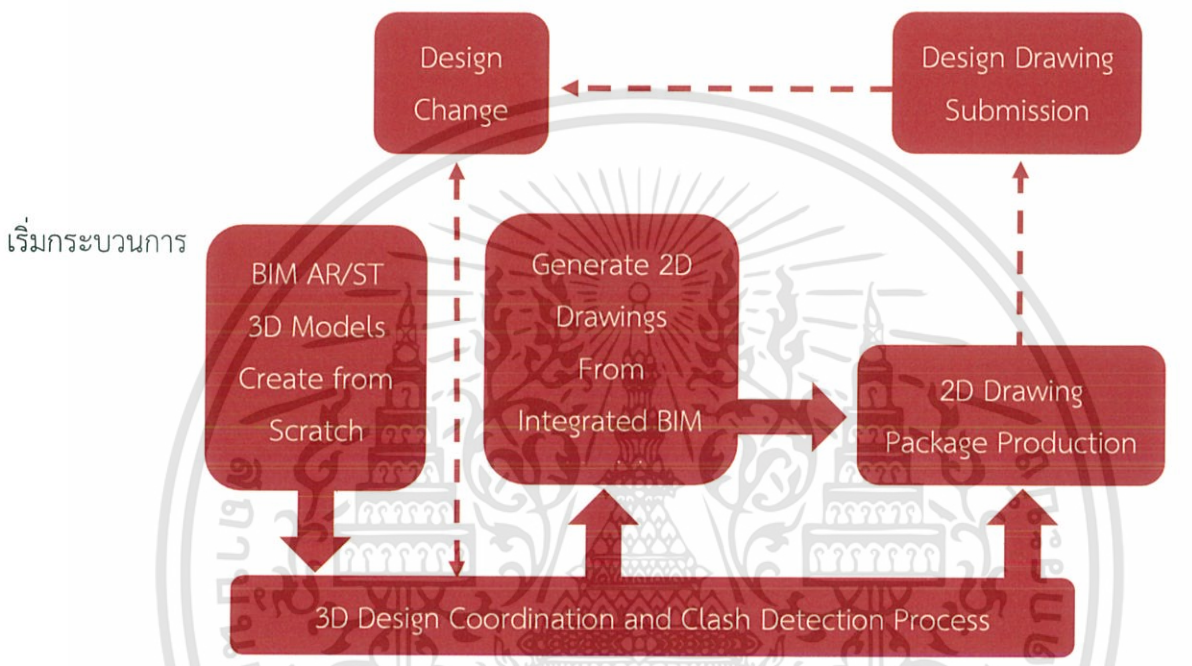
P คือพลังงานของปั๊ม หน่วยกิโลวัตต์ (kW)
Q คืออัตราการไหลของของไหล หน่วยลูกบาศก์เมตรต่อชั่วโมง (m^3/hr)
ρ คือค่าความหนาแน่นของน้ำ หน่วยกิโลกรัมของลูกบาศก์เมตร (kg/m^3)
g คืออัตราเร่งเนื่องจากแรงโน้มถ่วงของโลก หน่วยเป็นเมตรต่อวินาทีกำลังสอง (m/s^2)
H คือการสูญเสียความดันทั้งหมด หน่วยเมตร (m)
η คือประสิทธิภาพของปั๊ม (โดยส่วนใหญ่จะกำหนดให้มีค่าเท่ากับ 0.7)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีดำเนินการวิจัย

3.1 กระบวนการทำงาน



รูปที่ 3.1 แสดงแนวคิดกระบวนการจัดส่งของ BIM

จากรูปที่ 3.1 สามารถอธิบายขั้นตอนกระบวนการปฏิบัติงานด้วย BIM ดังนี้

1. กระบวนการทำงานนี้ เริ่มต้นจากการที่ผู้ใช้เริ่มสร้างแบบจำลอง 3 มิติของระบบ ไม่ว่าจะเป็นระบบสถาปัตยกรรม ระบบโครงสร้าง และระบบอื่นที่ใช้ภายในอาคาร จากแบบที่วาดภาพร่างด้วยมือหรือเขียนขึ้นมาจากโปรแกรมเขียนแบบทั่วไป เช่นโปรแกรม AutoCAD หรือ SketchUp เป็นต้น โดยที่ผู้ใช้จำเป็นที่จะต้องกำหนดจุดตั้งต้นหรือจุดอ้างอิง และต้องกำหนดให้มีค่าตรงกันทุกครั้งก่อนการสร้างแบบจำลอง
2. เมื่อทำการสร้างแบบจำลองของทุกระบบแล้ว นำแบบจำลอง 3 มิติของทุกระบบมารวมกันในโปรแกรม Autodesk Revit ซึ่งในขั้นตอนนี้ ถ้าจุดตั้งต้นหรือจุดอ้างอิงมีค่าไม่ตรงกันในแต่ละระบบ จะทำให้แบบจำลอง 3 มิติที่อยู่ในตำแหน่งที่ไม่ถูกต้อง จึงเป็นสาเหตุทำให้เกิดการชนกันขึ้น ถึงแม้ว่าจุดตั้งต้นหรือจุดอ้างอิงมีค่าที่ตรงกัน แต่ยังคงมีความเป็นไปได้ว่าแบบจำลอง 3 มิติของระบบต่างๆ ยังอาจจะเกิดการชนกันอยู่ เช่น ท่อในระบบสุขาภิบาล (Sanitary) อาจจะวิ่งตัดผ่านกับท่อในระบบปรับความดันบรรยากาศ (Environment Control Systems) ได้เช่นกัน ซึ่งในขั้นตอนนี้ผู้ใช้จำเป็นต้องทำการตรวจสอบแบบจำลอง 3 มิติเพื่อหาจุดที่ระบบชนกัน โดยใช้คำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Interference Check ในโปรแกรม Autodesk Revit ได้ แต่ทั้งนี้เครื่องมือนี้มีประสิทธิภาพไม่มากพอในการใช้งานสำหรับตรวจสอบทุกระบบ (เหมาะกับการตรวจสอบระบบโครงสร้าง)

3. สร้างแบบแปลน 2 มิติขึ้นมาจากโปรแกรม Revit โดยโปรแกรมจะสร้างมุมมองเหล่านี้ขึ้นโดยอัตโนมัติผู้ใช้สามารถปรับเปลี่ยนแบบแปลนนี้ได้เช่นเดียวกับการปรับเปลี่ยนใน 3 มิติ ขั้นตอนนี้มีจุดประสงค์ในการส่งงานให้กับผู้รับเหมาหรือผู้ที่ปฏิบัติงานในพื้นที่ เพื่อความละเอียดของข้อมูลและทำให้บุคคลที่ไม่มีส่วนร่วมในการสร้างแบบจำลอง 3 มิติเข้าใจถึงส่วนประกอบของแบบจำลอง 3 มิติมากขึ้น
4. หลังจากที่ทำการสร้างแบบแปลน 2 มิติทั้งหมดแล้ว ขั้นตอนต่อไปเป็นการส่งแบบเหล่านี้ไปให้ผู้ตรวจสอบ ซึ่งผู้ตรวจสอบมีหน้าที่ที่จะต้องส่งงานเหล่านี้กลับมาเพื่อทำการแก้แบบจำลอง 3 มิติที่มีข้อผิดพลาดหรืออนุมัติให้งานเหล่านี้ส่งไปยังผู้รับเหมาได้
5. ในกรณีที่ผู้ตรวจสอบไม่อนุมัติงาน ผู้ใช้จะต้องนำแบบกลับมาแก้ไขใหม่และทำการตรวจสอบจุดที่มีความผิดพลาด ในส่วนนี้จะเป็นการดำเนินงานวนซ้ำใหม่อีกครั้งหนึ่ง

จากกระบวนการการทำงานที่ได้กล่าวมานี้ ทำให้ผู้ศึกษาค้นพบถึงปัญหาที่ทำให้เกิดระหว่างกระบวนการการทำงานในแต่ละขั้นตอน อันก่อให้เกิดความล่าช้าในการทำงาน ซึ่งจะกล่าวในหัวข้อถัดไป

3.2 ปัญหาหลักที่ก่อให้เกิดผลกระทบในระหว่างการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคาร

จากการปฏิบัติงานตามกระบวนการข้างต้น ได้พบปัญหาที่ก่อให้เกิดผลกระทบที่ตามมา เช่น การทำงานเป็นไปอย่างล่าช้า เกิดการทำงานที่ซ้ำซ้อนและอาจจะทำให้เกิดข้อมูลผิดพลาด เป็นต้น ผู้ศึกษาจึงขอกล่าวถึงปัญหาหลักที่ต้องได้รับการแก้ไข ดังนี้

1. การแสดงมุมมองของส่วนที่เฉพาะเจาะจงของแบบจำลอง 3 มิติโดยใช้เครื่องมือ Section Views
2. การเพิ่มพารามิเตอร์ Reference Level ของท่อใน Schedule
3. การคำนวณหาขนาดของมอเตอร์ปั๊มน้ำในระบบประปาและสุขาภิบาลที่ซ้ำซ้อน
4. การสร้าง Grid Line ในรูปแบบ 3 มิติ
5. การระบุตำแหน่งของวัตถุที่ต้องการในแบบจำลอง 3 มิติด้วย Grid Line

3.3 วิธีการแก้ปัญหา

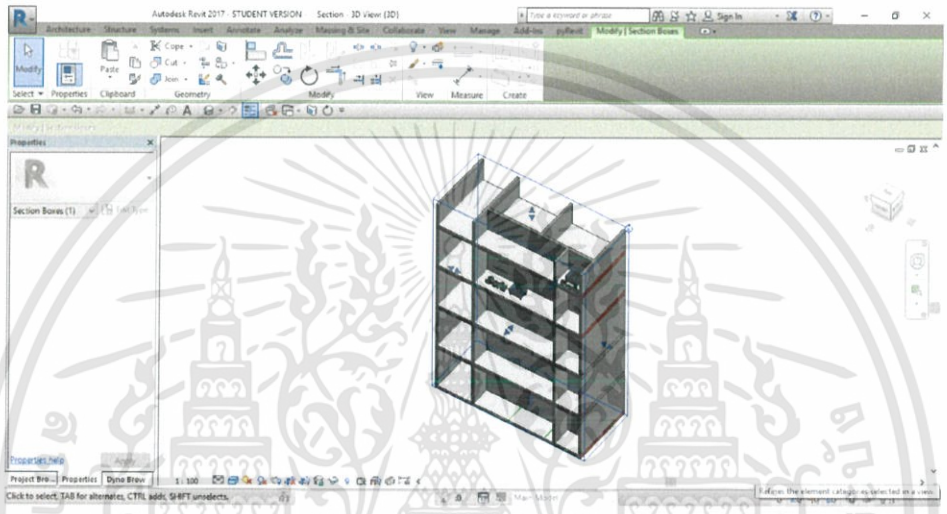
หลังจากที่ผู้ศึกษาระบุปัญหาที่เกิดระหว่างกระบวนการการทำงานออกมาได้แล้วนั้น ขั้นตอนต่อไปเป็นการแก้ไขปัญหาที่เกิดขึ้นเหล่านี้ ซึ่งในแต่ละปัญหาจะมีวิธีการแก้ไขที่แตกต่างกัน อีกทั้งซอฟท์แวร์ที่เข้ามาช่วยในการแก้ปัญหาจะแตกต่างกันด้วย เช่น การนำซอฟท์แวร์ Dynamo เข้ามาใช้ เพื่อการรันโค้ดเพิ่มเติมในสิ่งที่โปรแกรม Autodesk Revit ยังไม่สามารถทำได้ หรือการที่จะต้องใช้ภาษาคอมพิวเตอร์อื่นเข้ามาเพื่อเพิ่มเครื่องมือที่ช่วยในการปฏิบัติงาน เช่น การเขียนโค้ดในภาษา Python ทำการรันผ่านโปรแกรม Autodesk Revit เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. **การแสดงมุมมองของส่วนที่เฉพาะเจาะจงของแบบจำลอง 3 มิติโดยใช้เครื่องมือ Section Views**

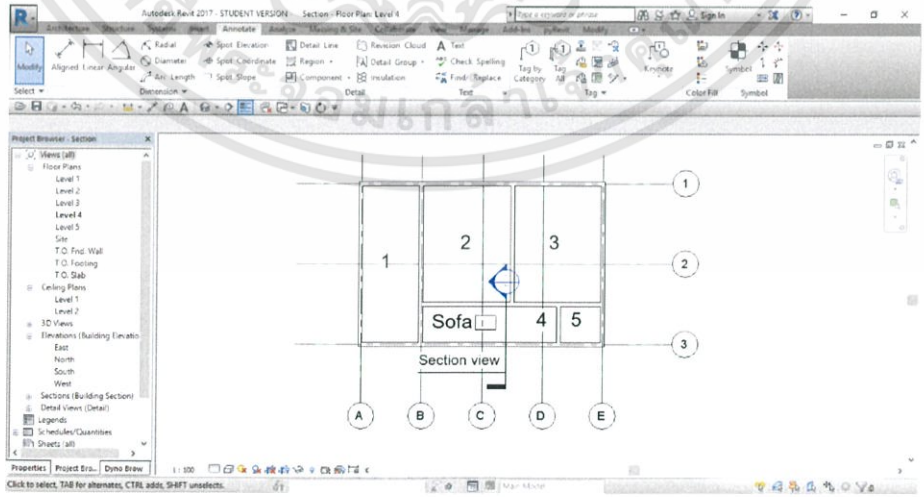
ในการสร้างแบบจำลอง 3 มิติ พบว่าชิ้นงานของผู้ศึกษาได้ถูกแบ่งเป็นหลายชั้นระดับ (Level) เมื่อใช้เครื่องมือ Section แล้วมุมมองที่ได้จะแสดงให้เห็นถึงทุกชั้นระดับที่มีในชิ้นงานดังรูปที่ 3.2 จึงมีวิธีการแก้ไขปัญหาดังนี้ เพื่อให้การใช้เครื่องมือ Section นี้แสดงผลลัพธ์เป็นมุมมองเฉพาะของระดับชั้นที่ต้องการ

1) สร้างแบบจำลอง 3 มิติของอาคาร ซึ่งในที่นี้ผู้ศึกษาได้ทำการสร้างแบบจำลอง 3 มิติของอาคาร 5 ชั้นและวางเฟอร์นิเจอร์ในแต่ละชั้นโดยในกรณีนี้ผู้ศึกษาเฉพาะเฟอร์นิเจอร์ที่วางอยู่ในชั้นที่ 4



รูปที่ 3.2 แสดงภาพแบบจำลอง 3 มิติของอาคาร 5 ชั้น

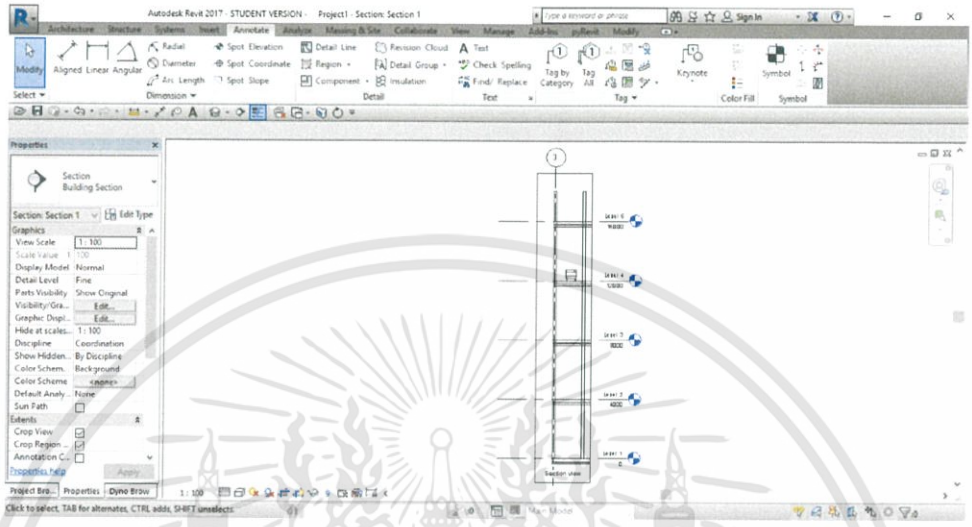
2) เปิดหน้าแปลน (Floor Plans) ของแบบจำลองในระดับชั้นที่ 4 (สังเกตได้จากหน้าต่าง Properties ทางด้านซ้ายมือจากภาพที่ 2 โดยตัวอักษรตัวหนาแสดงถึงหน้าแปลนของระดับชั้นนั้นกำลังถูกใช้งานอยู่) จากนั้นจึงวางเครื่อง Section ลงไปดังรูปที่ 3.3



รูปที่ 3.3 แสดงหน้าแปลนของแบบจำลองในระดับชั้นที่ 4

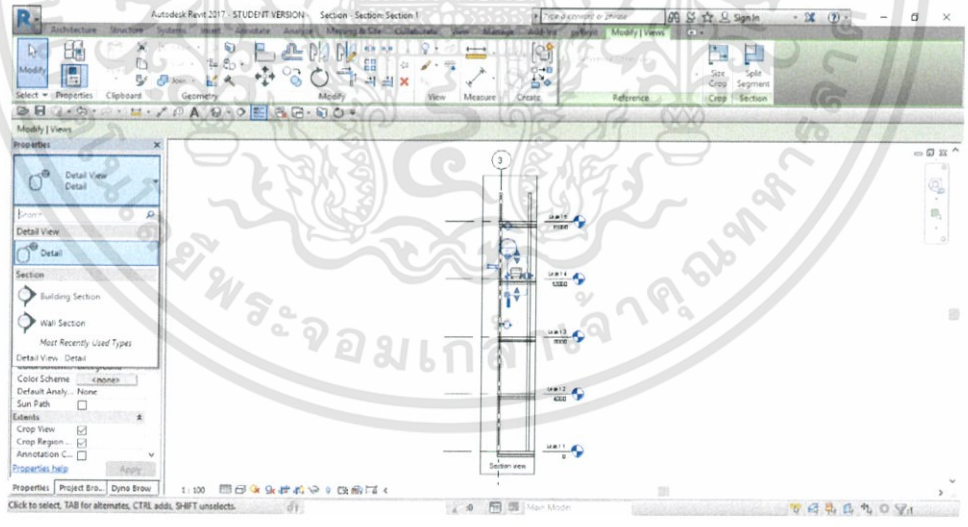
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) เมื่อเปิดหน้า Section View ดังแสงในรูปที่ 3.4 จะเห็นได้ว่าเครื่องมือ Section ได้ทำการตัดแบบจำลอง 3 มิติของอาคารในแนวขวางครบทุกระดับชั้น



รูปที่ 3.4 แสดงหน้า Section View ของอาคาร 5 ชั้น

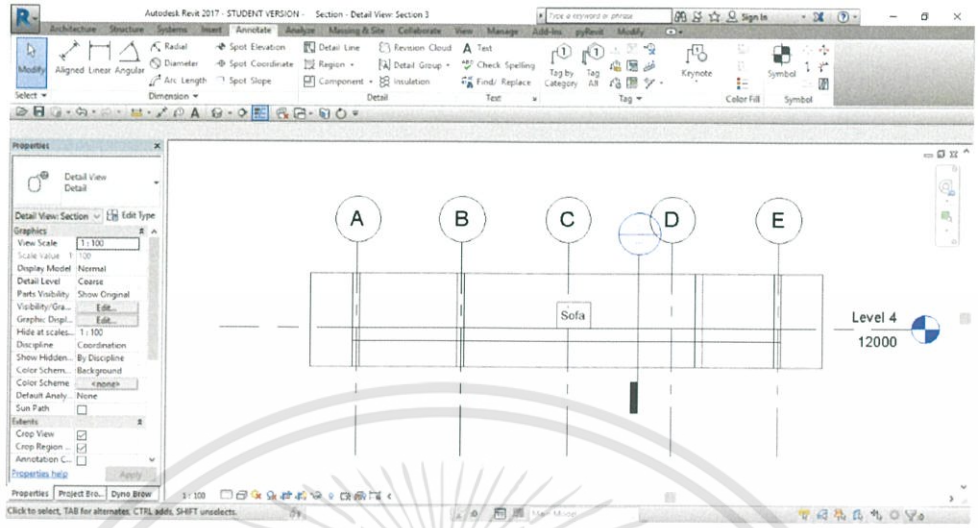
4) ใช้เครื่องมือ Detail View วางลงในชั้นระดับที่ต้องการในหน้า Section View จากขั้นตอนที่ 3 ดังแสดงในรูปที่ 3.5



รูปที่ 3.5 แสดงการใช้ Detial View

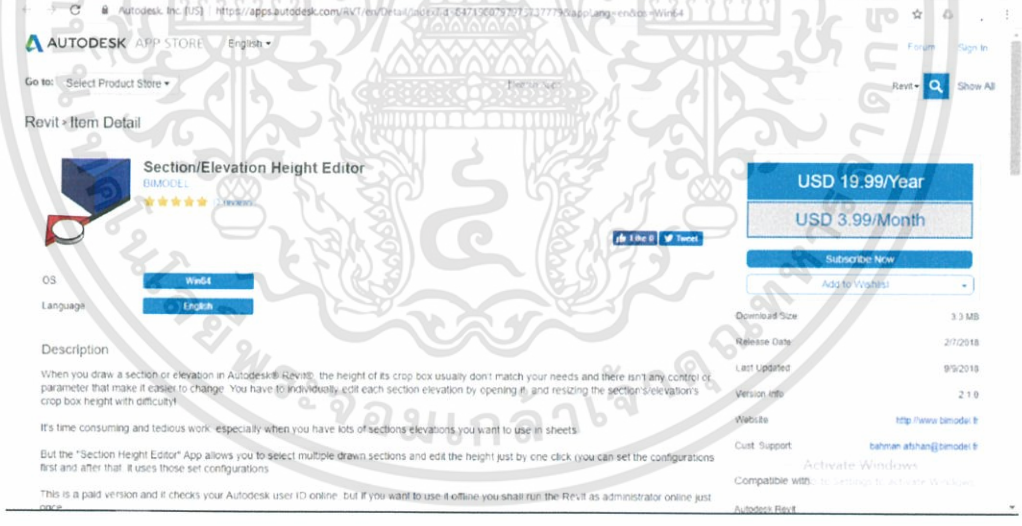
5) จากรูปที่ 3.6 แสดงให้เห็นว่า เมื่อเปิดหน้า Detail View ขึ้นมา จะพบว่าได้ระดับชั้นที่ถูกตัดออกมาได้ตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



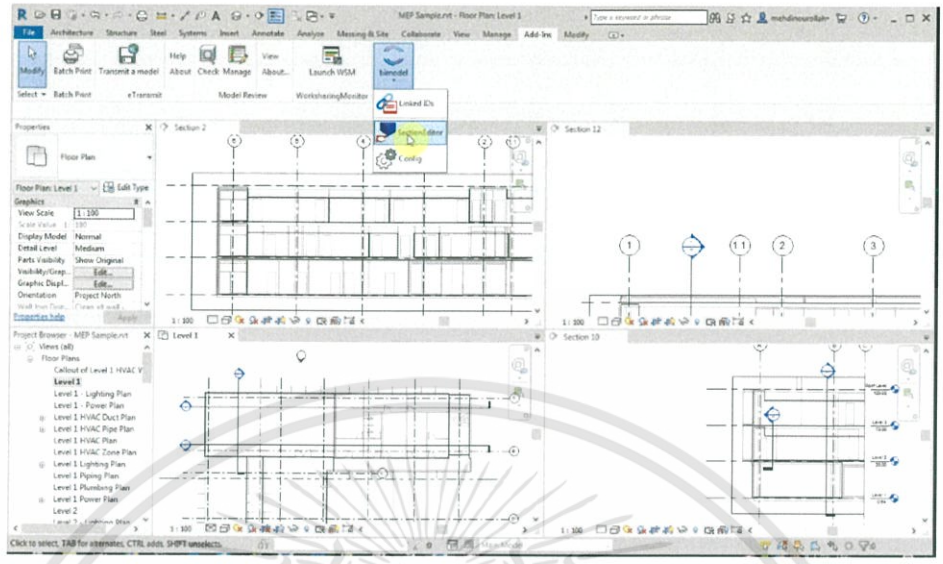
รูปที่ 3.6 แสดงชั้นระดับที่ได้หลังจากการใช้ Detail View

หมายเหตุ ทางบริษัท Autodesk ได้มีเครื่องมือ Section/Elevation Height Editor ดังแสดงในรูปที่ 3.7 ซึ่งเป็นเครื่องมือเสริมที่ช่วยในการปรับระยะของ Crop box ให้อยู่ในระยะที่ต้องการด้วยการตั้งค่าระยะที่ต้อง จากนั้นโปรแกรมจะทำการปรับระยะของ Section View โดยอัตโนมัติดังแสดงในรูปที่ 3.8 และ 3.9 ตามลำดับ แต่จะต้องเสียค่าใช้จ่ายตามที่บริษัทกำหนด

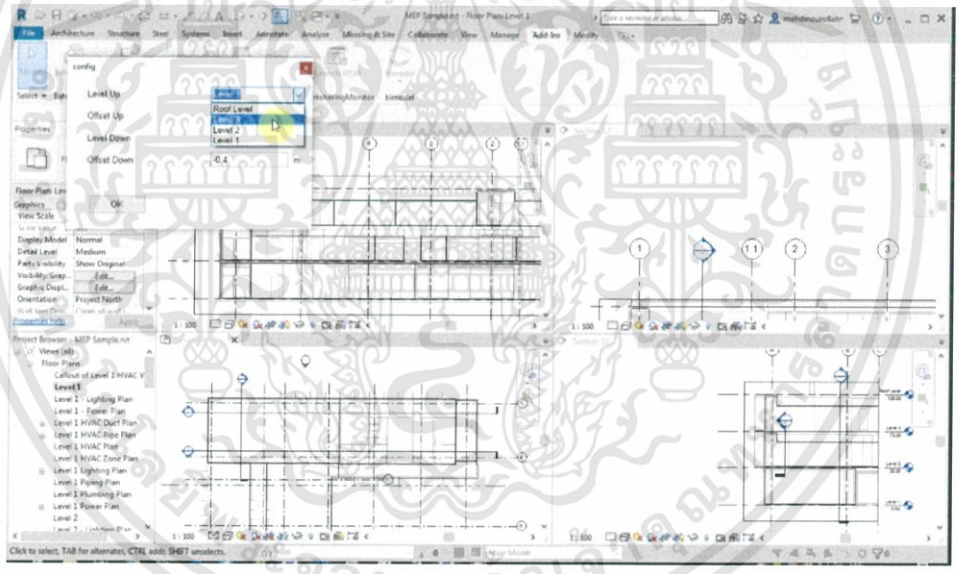


รูปที่ 3.7 แสดงเครื่องมือ Section/Elevation Height Editor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงการใช้งานเครื่องมือ Section/Elevation Height Editor



รูปที่ 3.9 แสดงการใช้งานเครื่องมือ Section/Elevation Height Editor

2. การเพิ่มพารามิเตอร์ Reference Level ของท่อใน Schedule

เนื่องจากพารามิเตอร์ ‘Reference Level’ เป็นพารามิเตอร์ที่บอกระดับชั้นของวัตถุที่ต้องการ ซึ่งพารามิเตอร์นี้จะปรากฏใน Properties Browser แต่ไม่ปรากฏใน Schedule ซึ่งพารามิเตอร์ตัวนี้มีหน้าที่บอกระยะของท่อจากจุดอ้างอิงมายังจุดที่ต้องการ ดังนั้นผู้ศึกษาจึงเลือกใช้ภาษา Python เข้ามาเพื่อเขียนโค้ดในการสร้างพารามิเตอร์ในโปรแกรม Autodesk Revit ซึ่งโค้ดที่จะกล่าวดังต่อไปนี้เป็นโค้ดที่ได้มาจากแหล่งข้อมูลที่อนุญาตให้ผู้ศึกษานำมาศึกษาและวิเคราะห์การทำงานของภาษา Python

สามารถอธิบายขั้นตอนการทำงานของโค้ดอย่างละเอียดได้ตามขั้นตอนต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 1 ในการเขียนโค้ดด้วยภาษา Python ส่วนแรกในการเขียนควรจะเป็นโค้ดที่ระบุชื่อของชิ้นงานและชื่อของผู้จัดทำ นั่นคือ

1.1) *__title__ = 'BOP\nMSL.'* แสดงถึงชื่อชิ้นงานที่ปรากฏใน Tooltip (คำอธิบายสั้นๆ เมื่อผู้ใช้นำเมาส์ไปชี้ที่เครื่องมือของโปรแกรม)

1.2) *__author__ = 'Yuttana Kongcarat'* แสดงถึงชื่อของผู้จัดทำที่ปรากฏใน Tooltip

ขั้นตอนที่ 2 เรียกเครื่องมืออื่นเข้ามาในชิ้นงาน โดยใช้โค้ด *import* ในกรณีนี้ชิ้นงานจำเป็นต้องใช้ข้อมูลที่มาจาก .NET library และต้องมีการดาวน์โหลด assemblies เข้ามา โดยการใช้ import clr (Common Language Runtime)

```
import clr
clr.AddReference("RevitAPI")
from Aurodesk.Revit.DB import *
clr.AddReference("RevitAPIUI")
from Aurodesk.Revit.UI import *
```

ทั้ง ReitAPI.dll และ RevitAPIUI.dll ต่างก็เป็น assemblies ที่จำเป็นเพื่อทำให้ Macro manager และ Revit macro IDE (Integrated Development Environment) สามารถสื่อสารเชื่อมโยงกันได้ อีกทั้ง Autodesk.RevitDB เป็นลิงก์ที่เรียกใช้ Class ทั้งหมดใน Autodesk.Revit.DB Namespace เนื่องจากเป็นที่ที่บรรจุ Class และ Method ทั้งหมดที่ถูกใช้งานในโปรแกรม Autodesk Revit นอกจากนี้ลิงก์ที่ใช้ในการเรียก Class อีกประเภทหนึ่งคือ Autodesk.Revit.UI ซึ่งเป็นลิงก์ที่รวบรวม Class ที่เกี่ยวข้องกับการปรับแต่งแถบเครื่องมือ (Ribbon) ในโปรแกรม Autodesk Revit ตามความเชี่ยวชาญและความเหมาะสมของผู้ใช้

```
From Autodesk.Revit.DB import FilteredElementCollector, BuiltInCategory, Transaction
```

จากลิงก์ Autodesk.Revit.DB มีการเรียก Class อื่นๆ เพิ่มเข้ามาเพื่อช่วยในการทำงานมากยิ่งขึ้น ได้แก่

FilteredElementCollector ใช้ในการค้นหาและกรองข้อมูลต่างๆ ที่ถูกนำเข้ามาในชิ้นงาน

BuiltInCategory รวบรวมวัตถุที่ถูกสร้างขึ้นมา เช่น กำแพง พื้น หน้าต่าง หลังคา เป็นต้น ซึ่งถูกจัดเก็บเป็น Category ในโปรแกรม Revit

Transaction เป็นบริบท (Context) ที่ป้องกันและควบคุมการเปลี่ยนแปลงที่ผู้ใช้กระทำต่อชิ้นงาน ในที่นี้คือแบบจำลอง 3 มิติในโปรแกรม Revit ซึ่งแบบจำลอง 3 มิติที่ถูกปรับเปลี่ยนแก้ไขได้เป็นเพราะเกิดการ Transaction ที่สมบูรณ์ หากเกิดการ Transaction ที่ไม่สมบูรณ์จะทำให้ระบบไม่บันทึกการเปลี่ยนแปลงที่เกิดขึ้นกับแบบจำลอง 3 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
doc = __revit__.ActiveUIDocument.Document
```

เนื่องจาก Revit UIApplication จะปฏิบัติการได้เมื่อเขียนโค้ด *__revit__* อีกทั้ง ActiveUIDocument เป็นคุณสมบัติในการยินยอมสิทธิเพื่อเข้าถึงวัตถุที่แสดงถึงชิ้นงานของผู้ใช้

ขั้นตอนที่ 3 สร้างตัวเก็บข้อมูล (Collector Instance) ของชิ้นงาน

```
pipe_collector = FilteredElementCollector(doc) .OfCategory(BuiltInCategory.OST_PipeCurves)\
                                               .WhereElementIsNotElementType()\
                                               .ToElements()
level_collector = FilteredElementCollector(doc) .OfCategory(BuiltInCategory.OST_Levels)\
                                               .WhereElementIsNotElementType()\
                                               .ToElements()
```

ในการสร้าง Collector Instance นี้ จัดทำขึ้นเพื่อเก็บข้อมูล 2 ชนิดด้วยคือท่อ (Pipes & Ducts) และชั้นระดับ (Level) โดยใช้ FilteredElementCollector เพื่อดึงค่าข้อมูลที่ออกมาจาก Revit Database เนื่องจากท่อและชั้นระดับถือว่าเป็น Category จึงใช้ Class BuiltinCategory ตามด้วยชื่อของ Category ที่ต้องการ (ในกรณีนี้เป็นชื่อของ Category ที่มีอยู่ใน Revit Database) เพื่อดึงข้อมูลที่ต้องการทั้งหมดออกมา อีกทั้งยังใช้ Method อีก 2 ชนิดคือ *WhereElementIsNotElement* เพื่อกรองข้อมูลที่อาจจะออกมาในรูปแบบอื่น เช่น Pipe Type เป็นต้น และ ToElements เพื่อส่งข้อมูลผ่าน Filter ไปยังพารามิเตอร์ที่กำหนด

ขั้นตอนที่ 4 สร้างพารามิเตอร์ใหม่เพื่อรับค่าจากพารามิเตอร์ที่มีอยู่เดิม

```
t= Transaction(doc, "Set Pipes ID_Level_Ref")
t.Start()
```

Start เป็นหนึ่งใน Method ของ Transaction class ซึ่งมีหน้าที่เริ่มทำการรัน Context

```
for get_pipe_ref in pipe_collector:
    get_pipe_ref_param = get_pipe_ref.LookupParameter("Reference Level")
    set_pipe_ref_param = get_pipe_ref.LookupParameter("Level_Ref")
```

เมื่อดึงข้อมูลใน Collector Instance ของท่อที่ผู้ศึกษาได้สร้างขึ้นแล้ว จากนั้นจึงใช้ LookupParameter Method เพื่อเรียกใช้พารามิเตอร์ที่ต้องการโดยการสร้างพารามิเตอร์ใหม่ขึ้นมารองรับข้อมูลที่ถูกดึงมาจาก pipe_collector ในที่นี้หมายถึงค่าข้อมูลจากพารามิเตอร์ 'Reference Level' และ 'Level_Ref'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 5 การหาระยะ offset ของท่อ

get_pipe_offset = get_pipe_ref.LookupParameter("Offset").AsDouble()
get_pipe_od = get_pipe_ref.LookupParameter("Outside Diameter").AsDouble()

ในขั้นตอนนี้จะมีความคล้ายคลึงกับการใช้ LookupParameter Method ใน Pipe เพื่อเรียกใช้พารามิเตอร์ 'Reference Level' และ 'Level_Ref' แต่ในที่นี้จะเรียกใช้พารามิเตอร์ 'Offset' และ 'Outside Diameter' อีกทั้งค่าที่ได้จะเป็นตัวเลขที่มีทศนิยม เนื่องจากกำหนดไว้จากโค้ด *AsDouble*

```
for get_level in level_collector:
    get_level_ref_parm = get_level.LookupParameter("Elevation").AsDouble()
```

ในทางเดียวกัน การเรียกใช้พารามิเตอร์ 'Elevation' มีขั้นตอนเช่นเดียวกับการเรียกใช้พารามิเตอร์ที่กล่าวมาข้างต้น แต่การดึงข้อมูลใน Collector Instance จะเป็นการดึงข้อมูลของชั้นระดับ (Elevation)

```
if get_pipe_ref_param.AsElementId().IntegerValue == get_level.Id.IntegerValue:
  set_level = get_level_ref_param
  else:
  continue
```

จากเงื่อนไขการใช้คำสั่ง if... else คือ เมื่อค่าของพารามิเตอร์ 'Reference Level' ตรงกับค่าของพารามิเตอร์ 'Elevation' แล้วค่าพารามิเตอร์ 'set_level' (พารามิเตอร์ใหม่) จะมีค่าเท่ากับค่าของพารามิเตอร์ 'Elevation' ด้วย ซึ่งทุกค่าที่ได้เป็นตัวเลขจำนวนเต็มเนื่องจากมีการกำหนดโดยใช้โค้ด. IntegerValue

pipe_bop = (get_pipe_offset – get_pipe_od /2) + set_level # get_level_ref_param

เขียนโค้ดสมการเพื่อคำนวณหาค่า BOP (Bottom Of Pipe) โดยการนำค่า offset ของท่อลบด้วยรัศมีของท่อ จากนั้นจึงนำไปรวมกับค่าของ set_level

ขั้นตอนที่ 6 สร้างพารามิเตอร์ 'BOP'

set_pipe_offset = get_pipe_ref.LookupParameter("BOP").Set(pipe_bop)

เมื่อคำนวณหา BOP (Bottom Of Pipe) ได้แล้ว ค่าที่ได้จะถูกจัดเก็บไว้ในพารามิเตอร์ 'BOP'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

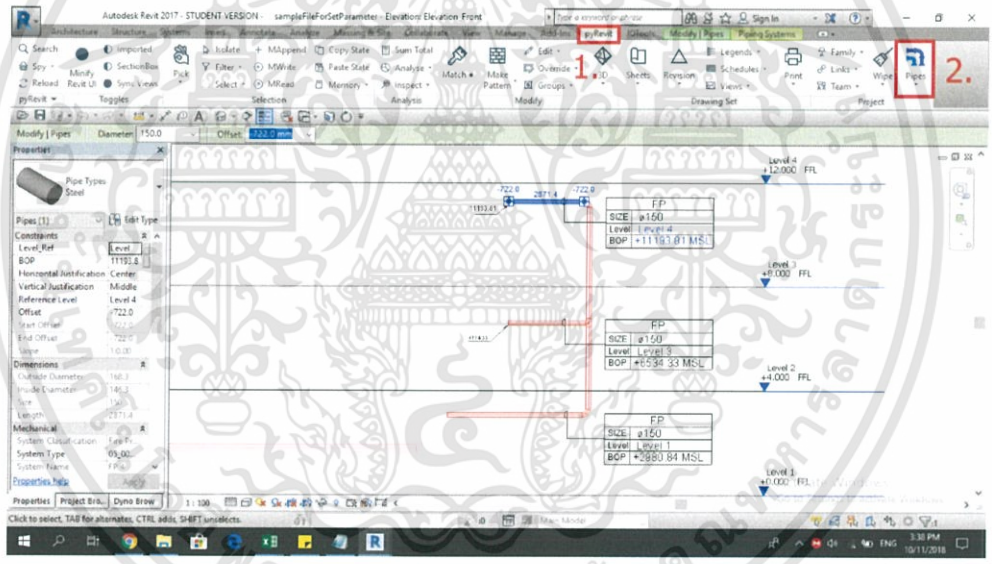
Set_pipe_ref_param.Set(get_pipe_ref_param.AsValueString())

เปลี่ยนประเภทของพารามิเตอร์ ‘Level_Ref’ จากตัวเลขจำนวนเต็ม (Integer) ให้กลายเป็นตัวอักษร (String)

t.Commit()

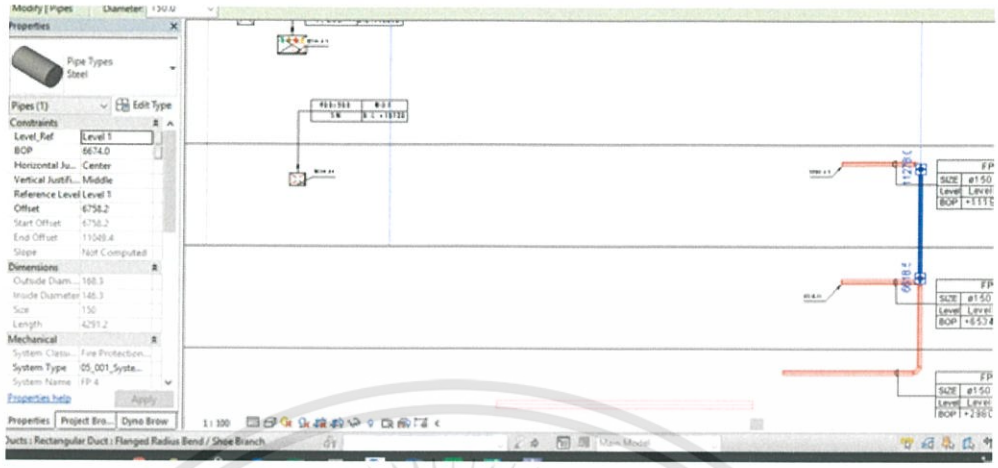
Commit เป็นหนึ่งใน Method ของ Transaction class ซึ่งมีหน้าที่เพื่อยุติการรัน Contextและบันทึกความเปลี่ยนแปลงที่เกิดขึ้นกับแบบจำลอง 3 มิติทั้งหมด

<u>ขั้นตอนที่ 7</u> ทำการรันโค้ดดังที่กล่าวมาจากข้างต้นผ่านแถบเครื่องมือ pyRevit ในโปรแกรม Autodesk Revit ถ้ารันโค้ดเสร็จสมบูรณ์แล้วจะเกิดเป็นเครื่องมือใน Ribbon ดังรูปที่ 3.10 โดยเครื่องมือนี้มีหน้าที่ในการปรับเปลี่ยนค่า BOP (Bottom Of Pipe) ในหน้าต่าง Properties ดังรูปที่ 3.11 เมื่อทำการกดใช้เครื่องมือ Pipes BOP จะเปลี่ยนค่าเมื่อระดับความสูงของท่อเปลี่ยนแปลงไป



รูปที่ 3.10 แสดงแถบเครื่องมือ pyRevit (หมายเลข 1) และเครื่องมือใน Ribbon เมื่อรันโค้ดเสร็จสมบูรณ์ (หมายเลข 2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แสดงค่า BOP (Bottom Of Pipe) ที่เปลี่ยนแปลงไปเมื่อทำการกดใช้เครื่องมือ Pipes

จากโค้ดที่กล่าวมาข้างต้นนี้ยังพบข้อผิดพลาดอยู่ประการหนึ่งคือชั้นระดับที่ระบุในพารามิเตอร์ ‘Reference Level’ และ ‘level_ref’ จะไม่เปลี่ยนแปลงไป เมื่อท่อถูกเคลื่อนย้ายไปยังชั้นระดับอื่น ตัวอย่างเช่น ในเริ่มแรกท่ออยู่ ณ ชั้นระดับที่ 4 เมื่อเคลื่อนย้ายท่อไปยังชั้นระดับที่ 3 พารามิเตอร์ ‘Reference Level’ และ ‘level_ref’ ยังคงระบุไว้ว่าท่อนั้นยังอยู่ที่ชั้นระดับที่ 4 ดังนั้นผู้ศึกษาจึงเขียนโค้ดในภาษา Python เพิ่มเติมเพื่อแก้ไขปัญหาดังกล่าวนี้

ก. การระบุชั้นระดับที่ถูกต้องของท่อชนิดกลม (Pipes)

สามารถอธิบายขั้นตอนการทำงานของโค้ดอย่างละเอียดได้ตามขั้นตอนต่อไปนี้

<u>ขั้นตอนที่ 1</u> เรียกเครื่องมืออื่นๆ เข้ามาในชิ้นงานโดยใช้โค้ด import เช่นเดียวกับขั้นตอนที่ 1 ในวิธีการเพิ่มพารามิเตอร์ Reference Level

```
import clr
Clr.AddReference(“RevitAPI”)
from Aurodesk.Revit.DB import *
clr.AddReference(“RevitAPIUI”)
from Aurodesk.Revit.UI import *
From Autodesk.Revit.DB import FilteredElementCollector, BuiltInCategory, Transaction
```

สร้างตัวเก็บข้อมูล (Collector Instance) ของชิ้นงานเช่นเดียวกับขั้นตอนที่ 2 ในวิธีการเพิ่มพารามิเตอร์ Reference Level

```
pipe_collector = FilteredElementCollector(doc) .OfCategory(BuiltInCategory.OST_PipeCurves)\
                                               .WhereElementIsNotElementType()\
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        .ToElements()
level_collector = FilteredElementCollector(doc) .OfCategory(BuiltInCategory.OST_Levels)\
                                        .WhereElementIsNotElementType()\
                                        .ToElements()
```

จากนั้นจึงใช้โค้ด t.Start() เพื่อเริ่มทำการรัน Context

```
t = Transaction(doc, "Test")
t.Start()
```

ขั้นตอนที่ 2 กำหนดฟังก์ชั่น frange

```
def frange(start, stop, step=0.01):
    i = start
    while i < stop:
```

เขียนโค้ดฟังก์ชั่น *frange* โดยโค้ดนี้ย่อมาจากคำว่า Float Range เพื่อใช้ระบุช่วงของค่าข้อมูลเลขทศนิยม (Float) ซึ่งโดยปกติแล้วช่วงของค่าข้อมูลนี้จะรับเพียงแค่ค่าข้อมูลเลขจำนวนเต็ม (Integer) เท่านั้น ดังนั้นจะต้องระบุฟังก์ชั่น frange ก่อนใช้ทุกครั้ง ในที่นี้ผู้ศึกษาต้องการให้ list เก็บค่าข้อมูลเลขทศนิยมสองตำแหน่งในช่วงที่กำหนดไว้

```
yield i
i += step
```

อีกทั้งยังมีโค้ด yield เพื่อทำให้ list ไม่จำเป็นต้องเก็บค่าข้อมูลตัวเลขทั้งหมด สามารถช่วยลดพื้นที่ความจำของอุปกรณ์

ขั้นตอนที่ 3 การเปลี่ยนข้อมูลเลขทศนิยม (float) เป็นข้อมูลตัวอักษร (string)

```
a = list(frange(0.0, 13.12, 0.01))
myRoundList_one = [str(round(elem_one, 2)) for elem_one in a]
print(myRoundList_one)
b= list(frange(13.13, 26.25, 0.01))
myRoundList_one = [str(round(elem_one, 2)) for elem_two in b]
print(myRoundList_two)
c= list(frange(26.26, 39.37, 0.01))
myRoundList_one = [str(round(elem_one, 2)) for elem_three in c]
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

print(myRoundList_three)
d= list(frange(39.38, 65.62, 0.01))
myRoundList_one = [str(round(elem_one, 2)) for elem_four in d]
print(myRoundList_four)

เนื่องจากเมื่อทำการรันโค้ดใน Interactive Python Shell ซึ่งเป็น Ribbon ที่อยู่ในแถบเครื่องมือ Add-ins ในโปรแกรม Autodesk Revit ค่าบางค่าใน List ของชุดของค่าข้อมูลตัวเลขตามที่ระบุไว้จะเปลี่ยนจากทศนิยมสองตำแหน่งเป็นทศนิยมแปดตำแหน่ง ดังนั้นผู้ศึกษาจึงเขียนโค้ด *round* ลงไปเพื่อเป็นการบังคับให้ค่าข้อมูลของตัวเลขแต่ละตัวมีทศนิยมสองตำแหน่งและเปลี่ยนประเภทของค่าข้อมูลของตัวเลขเหล่านี้จากค่าข้อมูลเลขทศนิยม (float) เป็นค่าข้อมูลตัวอักษร (string) นั่นคือค่าข้อมูลประเภทเลขจำนวนเต็ม (Integer) และเลขทศนิยม (float) ไม่สามารถทำซ้ำได้ใน Iterative Loop Statement

ขั้นตอนที่ 4 สร้างพารามิเตอร์ใหม่เพื่อรับค่าจากพารามิเตอร์ที่มีอยู่เดิม
ขั้นตอนนี้มีวิธีการทำเช่นเดียวกับขั้นตอนที่ 4 ในวิธีการเพิ่มพารามิเตอร์ Reference Level ผู้ศึกษาจึงขอละการอธิบายในขั้นตอนนี้

ขั้นตอนที่ 5 หาระยะ offset ของท่อ
ขั้นตอนนี้มีวิธีการทำเช่นเดียวกับขั้นตอนที่ 5 ในวิธีการเพิ่มพารามิเตอร์ Reference Level ผู้ศึกษาจึงขอละการอธิบายในขั้นตอนนี้

pipe_bop = (get_pipe_offset – (get_pipe_od /2))+ set _level # get_level_ref_param

จากนั้นจึงเขียนโค้ดสมการเพื่อคำนวณหาค่า BOP (Bottom Of Pipe) โดยการนำค่า offset ของท่อลบด้วยรัศมีของท่อ จากนั้นจึงนำไปรวมกับค่าของ set_level

ขั้นตอนที่ 6 สร้างพารามิเตอร์ 'BOP'
ขั้นตอนนี้มีวิธีการทำเช่นเดียวกับขั้นตอนที่ 6 ในวิธีการเพิ่มพารามิเตอร์ Reference Level ผู้ศึกษาจึงขอละการอธิบายในขั้นตอนนี้

ขั้นตอนที่ 7 List ค่าข้อมูล BOP ที่ได้จากการคำนวณ
เมื่อได้ค่า pipe_bop หรือค่า BOP ของท่อชนิดกลม (Pipes) ที่ได้มาจากการคำนวณในขั้นตอนที่ 5 จากนั้นนำมาใส่ลงใน list โดยใช้โค้ด *list_bop* และเปลี่ยนค่าข้อมูลดังกล่าวเป็นข้อมูลประเภทค่าข้อมูลตัวอักษร (string) โดยใช้โค้ด *round* เช่นเดียวกับขั้นตอนที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
print(pipe_bop)
list_bop = [pipe_bop]
print(list_bop)
myRoundList-bop = [str(round(elem_five, 2)) for elem_five in list_bop]
print(myRoundList_bop)
```

ขั้นตอนที่ 8 การแสดงผลของชั้นระดับ

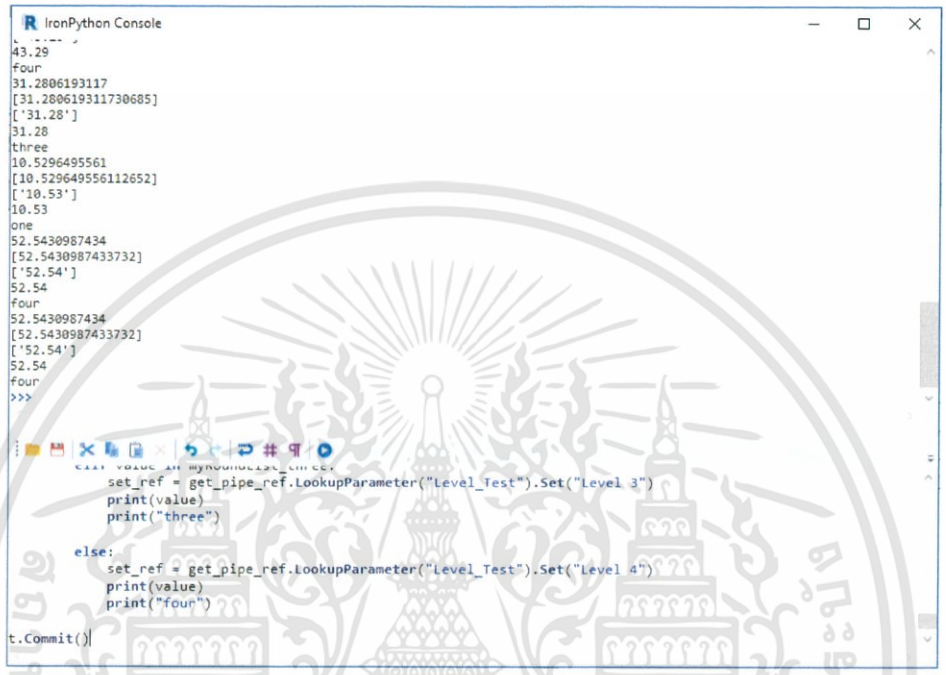
เมื่อได้ค่าข้อมูลจาก List ของ Range ของค่าข้อมูลตัวเลขและค่าข้อมูล BOP แล้ว จากนั้นจึงนำมาเข้า Iterative Loop Statement และ Conditional Statement และใช้เครื่องมือ LookupParameter เพื่อเปรียบเทียบค่าข้อมูลตัวอักษร (string) ของทั้ง 2 lists และถ้า condition ตรงกับค่าที่กำหนด โปรแกรมจะใส่ค่าข้อมูลของชั้นระดับที่กำหนดไว้ในแต่ละ condition ลงไปในพารามิเตอร์ 'Level_Test'

```
for value in myRoundList_bop:
    if value in myRoundList_one:
        set_ref = get_pipe_ref.LookupParameter("Level_Test").Set("Level 1")
        print(value)
        print("Why")
    elif value in myRoundList_two:
        set_ref = get_pipe_ref.LookupParameter("Level_Test").Set("Level 2")
        print(value)
        print("is this")
    elif value in myRoundList_three:
        set_ref = get_pipe_ref.LookupParameter("Level_Test").Set("Level 3")
        print(value)
        print("code")
    else:
        set_ref = get_pipe_ref.LookupParameter("Level_Test").Set("Level 4")
        print(value)
        print("not working!?")
t.Commit()
```

และจบการรันบริบท (Context) ด้วย t.Commit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 9 จากนั้นจึงรันโค้ดด้วย Interactive Python Shell เพื่อทดสอบว่าโค้ดนี้ใช้งานได้หรือไม่ ถ้าโค้ดนี้ไม่มีข้อผิดพลาด ค่าชั้นระดับของท่อชนิดกลม (pipes) จะปรากฏในพารามิเตอร์ 'Level Test'



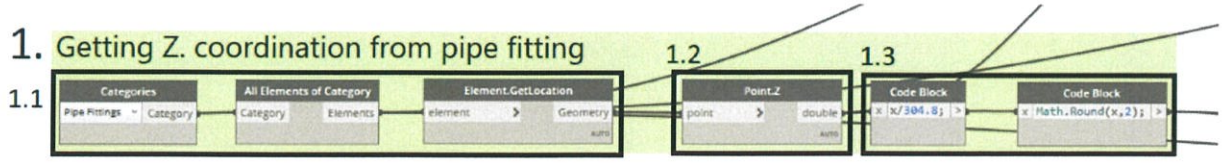
รูปที่ 3.12 แสดงวิธีการระบุชั้นระดับที่ถูกต้องของท่อชนิดกลม (Pipes) ในขั้นตอนที่ 9

ข. การระบุชั้นระดับที่ถูกต้องของท่อชนิดเหลี่ยม (Ducts)

วิธีการระบุชั้นระดับที่ถูกต้องของท่อชนิดเหลี่ยม (Ducts) สามารถทำได้เช่นเดียวกับการระบุชั้นระดับที่ถูกต้องของท่อชนิดกลม (Pipes) เพียงแต่เปลี่ยนพารามิเตอร์ให้เป็นของท่อชนิดเหลี่ยมทั้งหมดและเปลี่ยนชื่อค่าจาก BOP (Bottom Of Pipe) เป็น BOD (Bottom Of Duct)

จากนั้นจึงทำการรันโค้ดเหล่านี้ใน Interactive Python Shell เพื่อตรวจสอบว่าโค้ดนี้มีข้อผิดพลาดหรือไม่ ถ้าโค้ดนี้ไม่มีข้อผิดพลาด ค่าชั้นระดับของท่อชนิดเหลี่ยม (ducts) จะปรากฏในพารามิเตอร์ 'Level TestI'

ค. การระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อ (Pipe Fitting)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

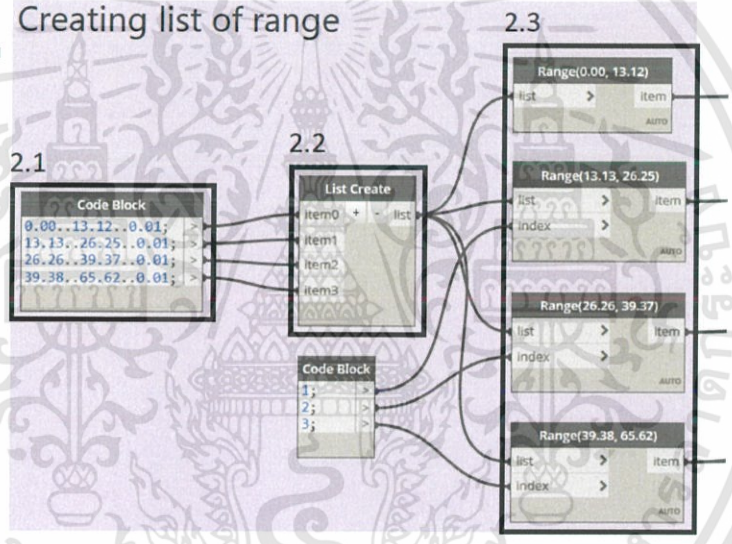
รูปที่ 3.13 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 1

ขั้นตอนที่ 1 เลือกค่าในแกน z จากจุดศูนย์กลางของอุปกรณ์ข้อต่อ (Pipe Fitting) ดังแสดงในรูปที่ 3.13

1.1) เลือกอุปกรณ์ข้อต่อทั้งหมดในชิ้นงานโดยใช้โหนด Categories และ All Elements Of Category

1.2) เลือกค่า z จากพิกัดจุดในแกน XYZ โดยใช้โหนด Point.Z เชื่อมต่อเข้ากับโหนด Element.Getlocation

1.3) แปลงหน่วยของค่า z จากมิลลิเมตร (mm.) เป็นหน่วยฟุต (ft.) และผลลัพธ์ที่ได้จะต้องทำให้มีทศนิยมเพียง 2 ตำแหน่ง



รูปที่ 3.14 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 2

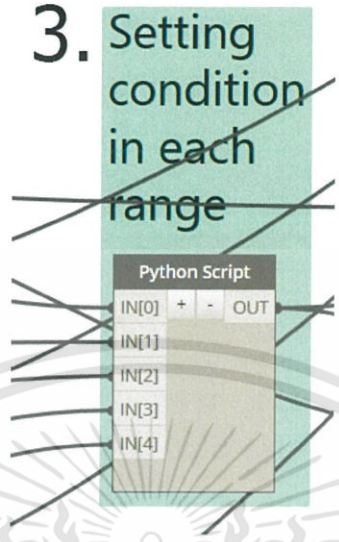
ขั้นตอนที่ 2 สร้างช่วงตัวเลขตามที่กำหนด ดังแสดงในรูปที่ 3.14

2.1) ใช้ Code block ในการเขียนโค้ด 0.00..13.12..0.01 หมายถึงตัวเลขเริ่มต้นจาก 0.00,0.01,...,13.12 เป็นต้น เพื่อสร้างช่วงลำดับตัวเลข ซึ่งค่าตัวเลขเหล่านี้คือค่าชั้นระดับ (Level) ที่ถูกแปลงหน่วยจากมิลลิเมตรให้กลายเป็นหน่วยฟุตแล้ว (เนื่องจากทำให้ช่วงตัวเลขมีขนาดเล็กลงและการใช้งานโค้ดจะเป็นไปได้อย่างรวดเร็วขึ้น)

2.2) นำช่วงตัวเลขทั้งหมดใส่ลงใน list

2.3) แยกช่วงแต่ละช่วงตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

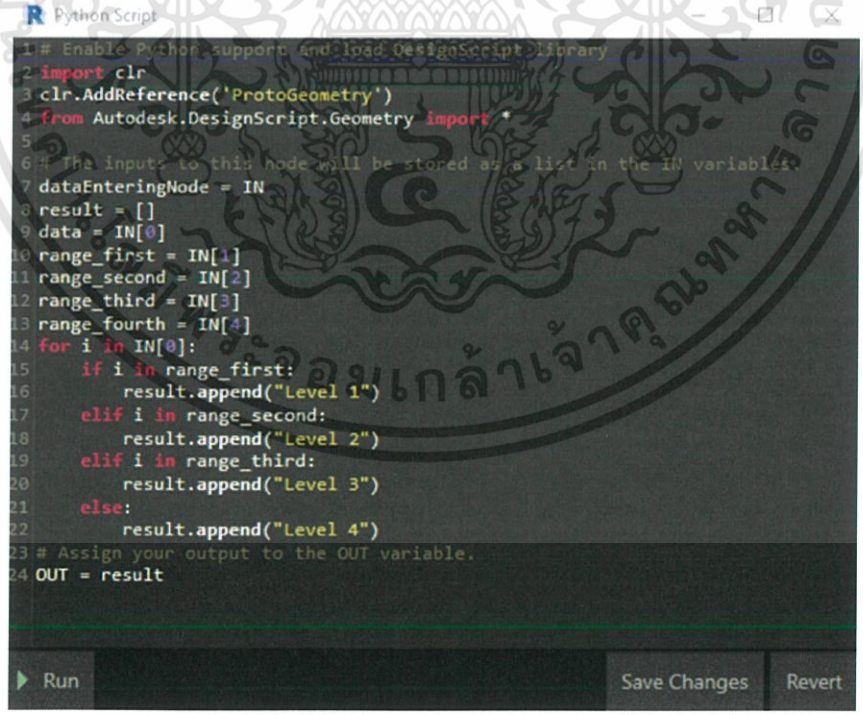


รูปที่ 3.15 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 3

ขั้นตอนที่ 3 เขียน Python Script เพื่อตั้งเงื่อนไขในการเปรียบเทียบ ดังแสดงในรูปที่ 3.15

3.1) เชื่อมโหนดผลลัพธ์ที่ได้จากขั้นตอนที่ 1.3 เข้ากับพอร์ต IN[0]

3.2) เชื่อมโหนดช่วงตัวเลขที่ได้จากขั้นตอนที่ 2.3 เข้ากับพอร์ต IN[1], IN[2], IN[3] และ IN[4] ตามลำดับ



รูปที่ 3.16 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 3.3

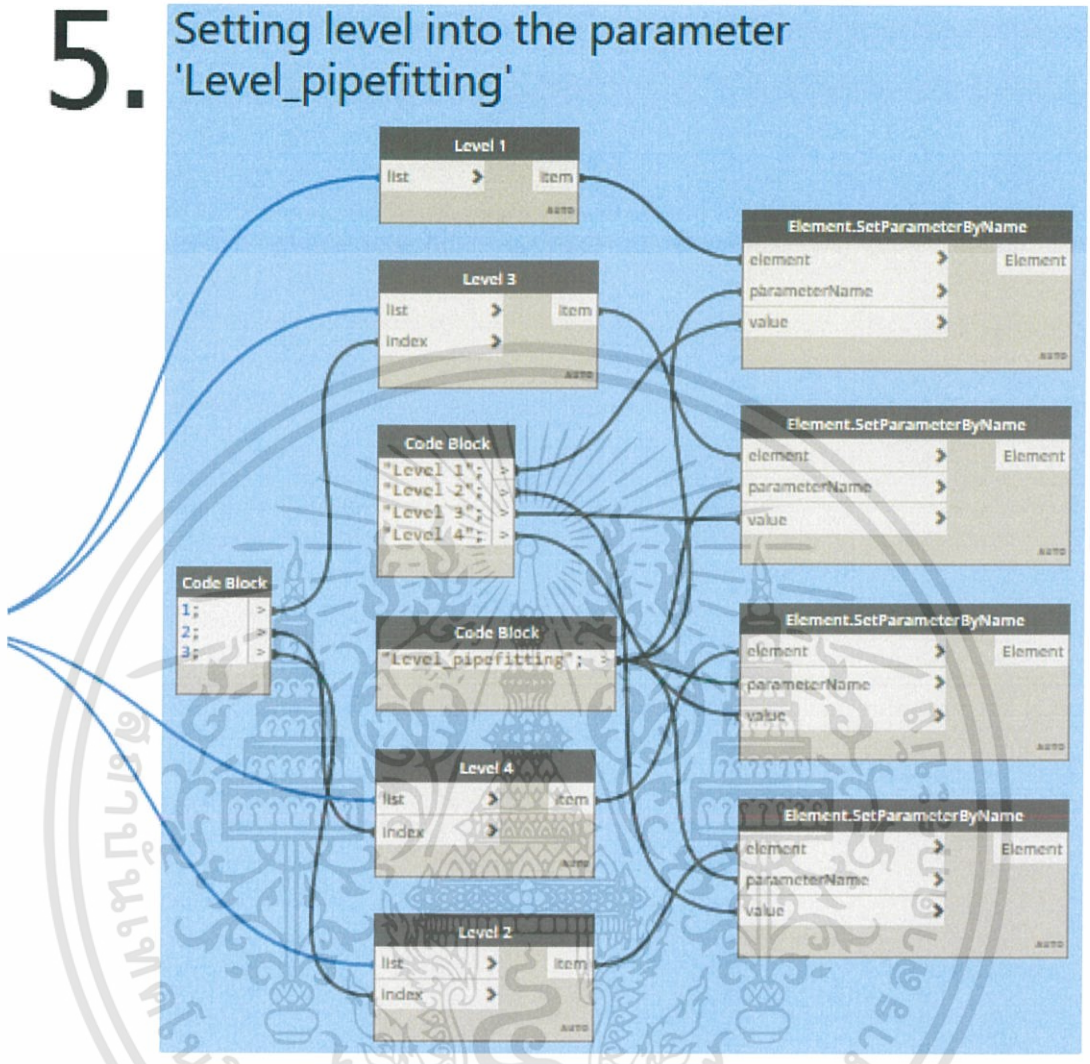
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3) จากการใช้ Python Script ดังรูปที่ 3.16 ถ้าผลลัพธ์จากขั้นตอนที่ 1.3 มีค่าตรงกับช่วงตัวเลขค่าชั้นระดับ (Level) ใดแล้ว ค่านั้นจะถูกใส่ลงใน list อีกครั้งหนึ่ง

รูปที่ 3.17 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 4

ขั้นตอนที่ 4 จัดกลุ่ม elements ที่ได้จากผลลัพธ์ในขั้นตอนที่ 3 เป็นปัจจัยในการจับกลุ่มดังรูปที่ 3.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 5

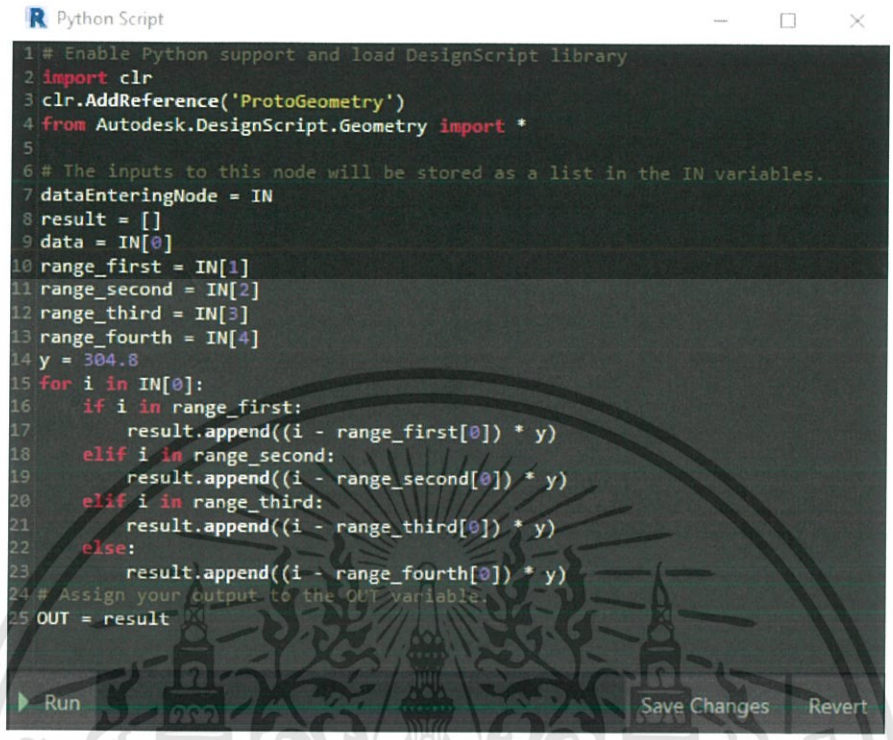
ขั้นตอนที่ 5 นำค่าที่ได้แยกเป็น list และนำค่าข้อมูลเหล่านี้ใส่ลงในพารามิเตอร์ ‘Level_pipefit-ting’ ตามลำดับของ key ดังรูปที่ 3.18

ขั้นตอนที่ 6 คำนวณหาค่า offset

6.1) นำค่าที่ได้จากขั้นตอนที่ 1 ใส่ลงในพารามิเตอร์ ‘Height’

6.2) เชื่อมเข้ากับพอร์ต IN[0] ของโหนด Python Script

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

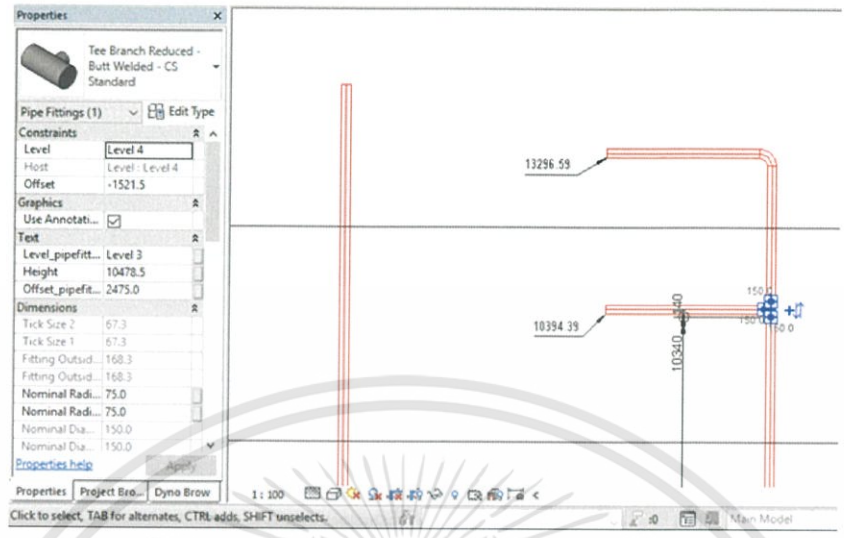


รูปที่ 3.19 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 6.3

6.3) จากการใช้ Python Script ดังรูปที่ 3.19 กำหนดช่วง range_first[0], range_second[0], range_third[0] และ range_fourth[0] โดยในแต่ละช่วงจะมีค่าเริ่มต้นของช่วงนั้นๆ คือ 0.00, 13.13, 26.16 และ 39.38 ตามลำดับ จากนั้นจึงหาค่า offset เทียบกับ กับ Reference Level ของอุปกรณ์ข้อต่อ (ค่า z ในที่นี้กำหนดให้เป็นตัวแปร i) โดยใช้การคำนวณดังนี้ *(i-ค่าเริ่มต้นของช่วง) * 304.8* ตัวอย่างเช่น ถ้าอุปกรณ์ข้อต่อมีค่า z ที่ 15.22 นั่นคือ range_second จากนั้นจึงนำไปคำนวณต่อไป และค่าที่ได้จะมีค่าเป็นบวกเสมอ

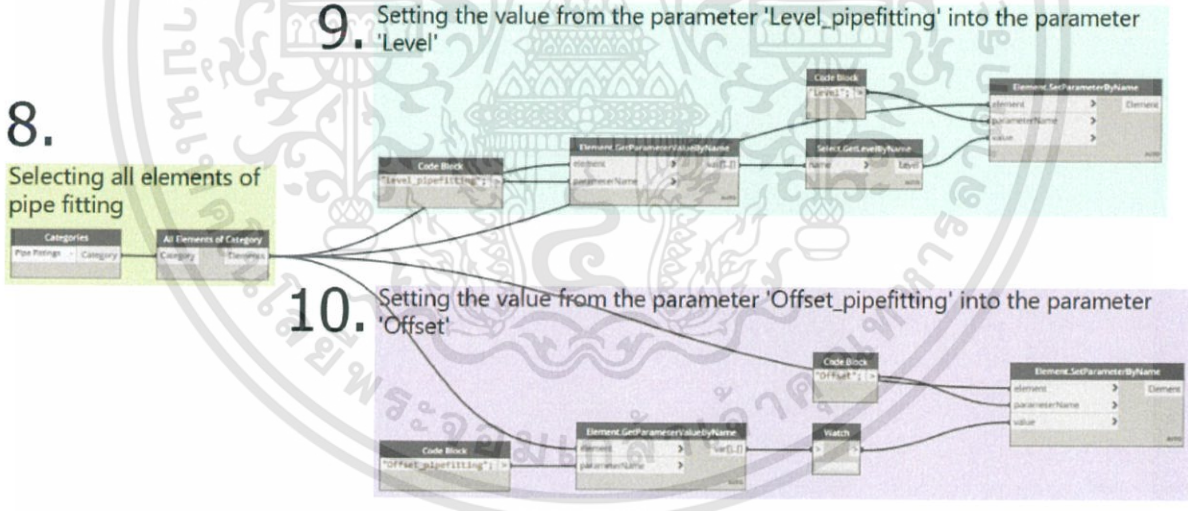
6.4) นำค่าที่ได้ใส่ลงในพารามิเตอร์ ‘Offset_pipefitting’

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 7

ขั้นตอนที่ 7 หลังจากนั้นจึงทำการรันโค้ดในซอฟท์แวร์ Dynamo จะได้ผลลัพธ์ดังรูปที่ 3.20 ต่อมาผู้ศึกษาได้ทำการปรับค่าพารามิเตอร์ ‘Level’ ให้ตรงกับพารามิเตอร์ ‘Level-pipefitting’ สามารถทำได้ตามขั้นตอนดังต่อไปนี้



รูปที่ 3.21 แสดงแผนภาพ Dynamo ในการระบุชั้นระดับที่ถูกต้องของอุปกรณ์ข้อต่อในขั้นตอนที่ 8 ถึง 10

ขั้นตอนที่ 8 เลือกอุปกรณ์ข้อต่อทั้งหมดในชิ้นงานด้วยโหนด Categories และ All elements in the categories

ขั้นตอนที่ 9 ถ่ายโอนค่าข้อมูลจากพารามิเตอร์ ‘Level_pipefitting’ ไปยังพารามิเตอร์ ‘Level’ ซึ่งพารามิเตอร์นี้เป็นพารามิเตอร์ที่มีอยู่แล้วในโปรแกรมและส่งผลต่อค่า Offset หลังจากย้ายค่าข้อมูลทั้งหมดแล้วชั้นระดับของอุปกรณ์ข้อต่อที่ได้จะมีค่าตรงกับชั้นระดับที่แท้จริงของอุปกรณ์ข้อต่อ

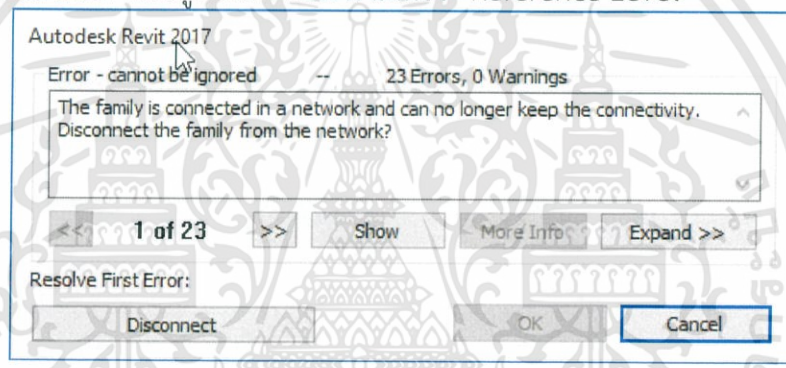
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั้นๆ โดยปกติแล้วอุปกรณ์ข้อต่อจะมี Reference Level ที่จุดเริ่มต้นของการสร้างท่อซึ่งมีค่าไม่แน่นอน

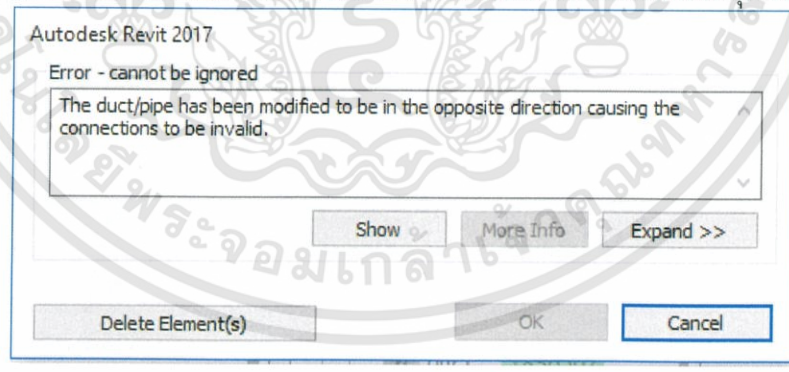
ขั้นตอนที่ 10 ถ่ายโอนค่าข้อมูลจากพารามิเตอร์ 'Offset_pipefitting' ไปยังพารามิเตอร์ 'Offset' ซึ่งพารามิเตอร์นี้เป็นพารามิเตอร์ที่มีอยู่แล้วในโปรแกรม

หมายเหตุ

1. การถ่ายโอนข้อมูลของพารามิเตอร์ทั้งสองตัวนี้จำเป็นต้องเปลี่ยนค่าพร้อมกัน หากเปลี่ยนตัวใดตัวหนึ่งก่อนจะทำให้เกิดข้อผิดพลาดและโปรแกรมจะแจ้งเตือนดังรูปที่ 3.22 และ 3.23 สาเหตุที่ต้องเปลี่ยนพารามิเตอร์ทั้งสองนี้พร้อมกัน เนื่องจากถ้าหากเปลี่ยนค่าในพารามิเตอร์ Reference Level แล้วค่าในพารามิเตอร์ Offset จะเปลี่ยนแปลงด้วยเช่นกัน ดังนั้นอุปกรณ์ข้อต่อในชิ้นงานจะเลื่อนระดับ อาจทำให้อุปกรณ์ข้อต่อและท่อหลุดการเชื่อมต่อกัน เพื่อป้องกันปัญหาที่เกิดขึ้นนี้จึงจำเป็นต้องบังคับค่าในพารามิเตอร์ Offset ให้มีค่าควบคู่ไปกับค่าในพารามิเตอร์ 'Reference Level'



รูปที่ 3.22 แสดงการแจ้งเตือนเมื่อโปรแกรมพบว่ามีท่อที่ไม่เชื่อมต่อกันและให้ยุติการเชื่อมต่อนั้น

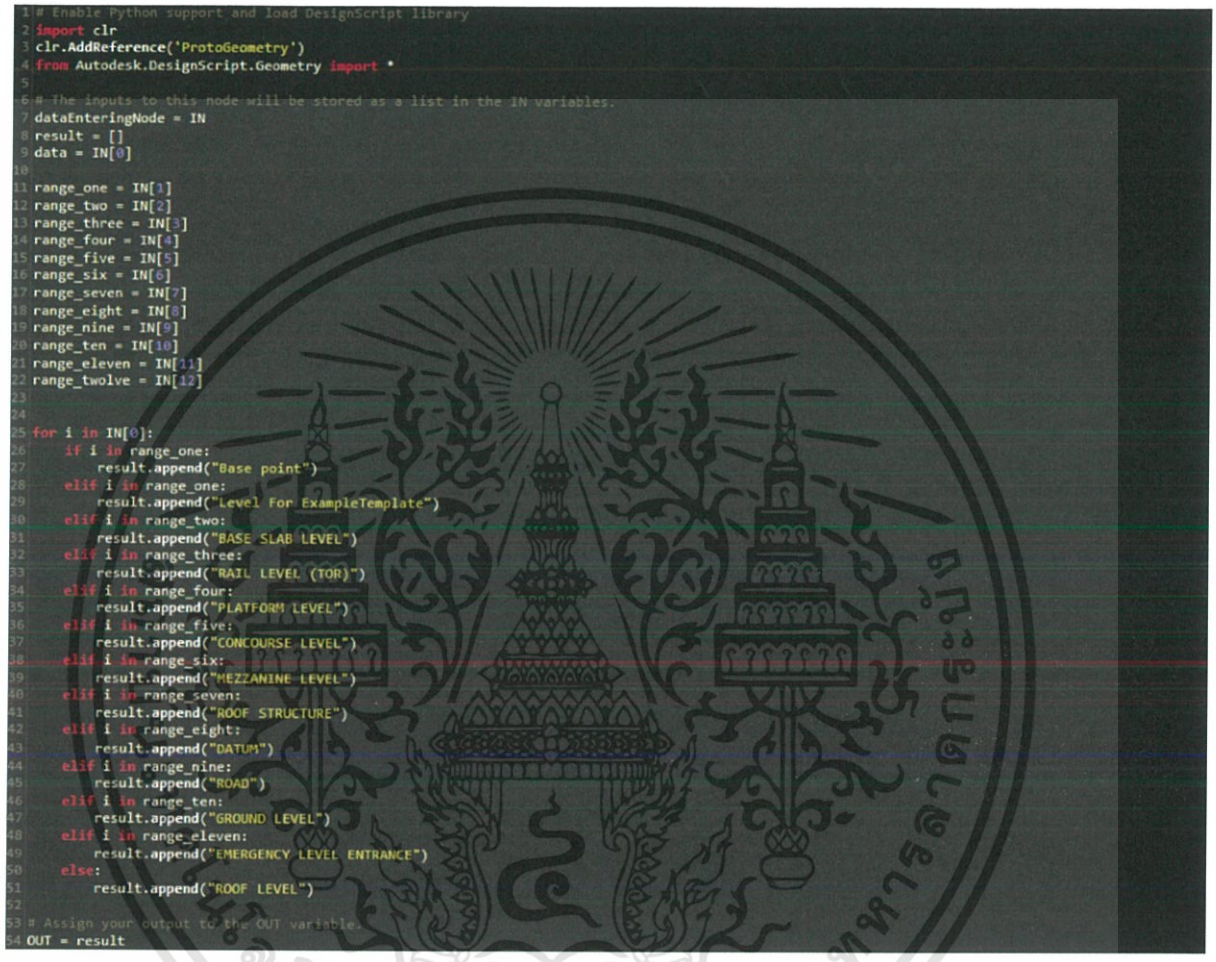


รูปที่ 3.23 แสดงการแจ้งเตือนเมื่อโปรแกรมพบว่ามีท่อที่ไม่เชื่อมต่อกันและให้ลบวัตถุ (ท่อ) นั้น

2. การเชื่อมโยงโหนดพารามิเตอร์ 'Level_pipefitting' และ พารามิเตอร์ 'Offset_pipefitting' จะต้องสมมาตรกัน โดยการเพิ่มโหนด Watch ในกระบวนการของพารามิเตอร์ 'Offset' เพื่อให้มีจำนวนโหนดเท่ากับกระบวนการของพารามิเตอร์ 'Level '

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

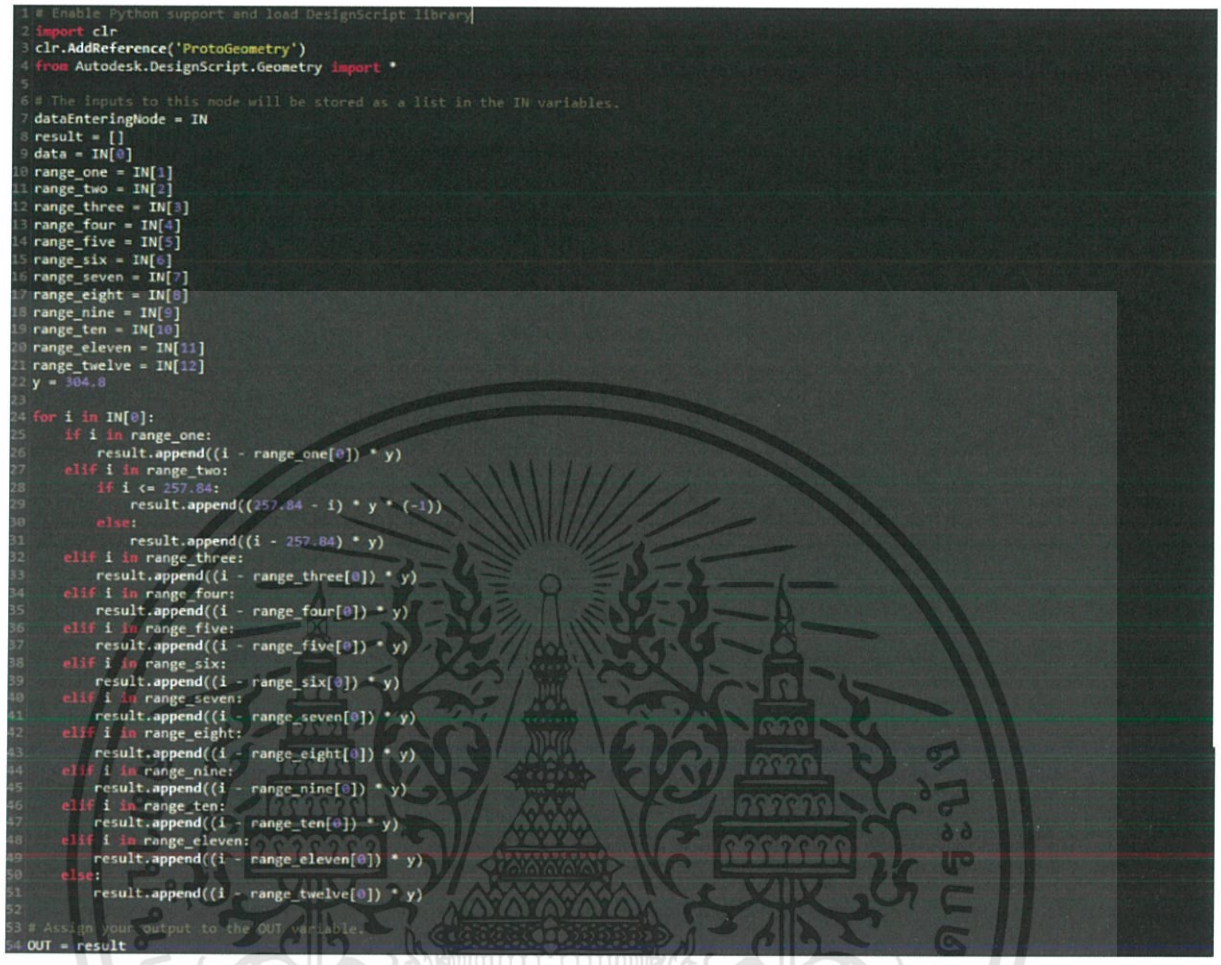
ทั้งนี้การนำโค้ดที่ได้จากซอฟท์แวร์ Dynamo จากการทดลองที่กล่าวมาข้างต้น สามารถนำมาใช้ได้ในการปฏิบัติงานจริงแต่จะต้องทำการปรับเปลี่ยน Python Script และ Dynamo Script เพื่อให้เหมาะสมกับชิ้นงานดังรูปที่ 3.24 ถึง 3.25



```python
# Enable Python support and load DesignScript library
import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *

# The inputs to this node will be stored as a list in the IN variables.
dataEnteringNode = IN
result = []
data = IN[0]

range_one = IN[1]
range_two = IN[2]
range_three = IN[3]
range_four = IN[4]
range_five = IN[5]
range_six = IN[6]
range_seven = IN[7]
range_eight = IN[8]
range_nine = IN[9]
range_ten = IN[10]
range_eleven = IN[11]
range_twolve = IN[12]


for i in IN[0]:
    if i in range_one:
        result.append("Base point")
    elif i in range_one:
        result.append("Level For ExampleTemplate")
    elif i in range_two:
        result.append("BASE SLAB LEVEL")
    elif i in range_three:
        result.append("RAIL LEVEL (TOR)")
    elif i in range_four:
        result.append("PLATFORM LEVEL")
    elif i in range_five:
        result.append("CONCOURSE LEVEL")
    elif i in range_six:
        result.append("MEZZANINE LEVEL")
    elif i in range_seven:
        result.append("ROOF STRUCTURE")
    elif i in range_eight:
        result.append("DATUM")
    elif i in range_nine:
        result.append("ROAD")
    elif i in range_ten:
        result.append("GROUND LEVEL")
    elif i in range_eleven:
        result.append("EMERGENCY LEVEL ENTRANCE")
    else:
        result.append("ROOF LEVEL")

# Assign your output to the OUT variable.
OUT = result
```

รูปที่ 3.24 แสดงแผนภาพ Python Script สำหรับการนำไปใช้ในการการสร้างแบบจำลอง 3 มิติของสถานีรถไฟฟ้าใต้ดินสายสีส้มในขั้นตอนการระบุชื่อของชั้นระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```python
# Enable Python support and load DesignScript library
import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *

# The inputs to this node will be stored as a list in the IN variables.
dataEnteringNode = IN
result = []
data = IN[0]
range_one = IN[1]
range_two = IN[2]
range_three = IN[3]
range_four = IN[4]
range_five = IN[5]
range_six = IN[6]
range_seven = IN[7]
range_eight = IN[8]
range_nine = IN[9]
range_ten = IN[10]
range_eleven = IN[11]
range_twelve = IN[12]
y = 304.8

for i in IN[0]:
    if i in range_one:
        result.append((i - range_one[0]) * y)
    elif i in range_two:
        if i <= 257.84:
            result.append((257.84 - i) * y * (-1))
        else:
            result.append((i - 257.84) * y)
    elif i in range_three:
        result.append((i - range_three[0]) * y)
    elif i in range_four:
        result.append((i - range_four[0]) * y)
    elif i in range_five:
        result.append((i - range_five[0]) * y)
    elif i in range_six:
        result.append((i - range_six[0]) * y)
    elif i in range_seven:
        result.append((i - range_seven[0]) * y)
    elif i in range_eight:
        result.append((i - range_eight[0]) * y)
    elif i in range_nine:
        result.append((i - range_nine[0]) * y)
    elif i in range_ten:
        result.append((i - range_ten[0]) * y)
    elif i in range_eleven:
        result.append((i - range_eleven[0]) * y)
    else:
        result.append((i - range_twelve[0]) * y)

# Assign your output to the OUT variable.
OUT = result
```

รูปที่ 3.25 แสดงแผนภาพ Python Script สำหรับการนำไปใช้ในการสร้างแบบจำลอง 3 มิติของสถานีรถไฟฟ้าใต้ดินสายสีส้มในขั้นตอนการคำนวณค่าพารามิเตอร์ ‘Offset_pipefitting’

<u>หมายเหตุ</u> โค้ดสำหรับการระบุชื่อของชั้นระดับใน Python Script ต้องมีการตรวจสอบว่าชื่อของชั้นระดับตรงกับชั้นระดับนั้นอย่างแน่นอน เนื่องจากในขั้นตอนการถ่ายโอนค่าข้อมูลไปยังพารามิเตอร์อีกตัวหนึ่ง ผู้ศึกษาจะต้องใช้โหนด Selection.GetLevelByName ซึ่งจะดึงค่าชั้นระดับมาจากข้อมูลตัวอักษร หากโปรแกรมไม่พบค่าข้อมูลเนื่องจากตัวอักษณ์พิมพ์เล็กและตัวอักษรพิมพ์ใหญ่ ผลลัพธ์ที่ได้คือไม่มีค่าหรือ null

3. การคำนวณหาขนาดของมอเตอร์ปั๊มน้ำในระบบประปาและสุขาภิบาลที่ซ้ำซ้อน

ในการหาขนาดมอเตอร์ปั๊มของน้ำ จะต้องคำนวณหาอัตราการไหลของของไหล (Flow rate), การสูญเสียเนื่องมาจากความฝืด (Friction loss), ความดันในท่อ (Pressure) และเฮดปั๊ม (Pump head) ของน้ำในท่อโดยผ่านการใช้โปรแกรม Autodesk Revit, โปรแกรม AutoCAD และโปรแกรม Microsoft Excel จากนั้นจึงนำกลับมาสร้างแบบจำลอง 3 มิติและเขียนข้อมูลเพื่อคำนวณด้วยเครื่องมือ Schedule ในโปรแกรม Autodesk Revit อีกครั้งหนึ่ง ซึ่งทั้งหมดนี้จะเป็นการทำงานที่ซ้ำซ้อนและอาจก่อให้เกิดข้อมูลที่ผิดพลาดได้ ผู้ศึกษาจึงได้คิดค้นวิธีการหาขนาดของมอเตอร์ปั๊มน้ำด้วยกัน 2 วิธี โดยวิธีแรกเป็นวิธีการที่มีการใช้ประโยชน์จาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสัมพันธ์ของโปรแกรม Autodesk Revit และโปรแกรม Microsoft Excel แต่ในส่วนของวิธีที่ 2 เป็นวิธีที่ทำกระบวนการต่างๆ โดยใช้ซอฟท์แวร์ Dynamo เป็นหลัก ซึ่งทั้ง 2 วิธีนี้เป็นวิธีที่มีประสิทธิภาพมากยิ่งขึ้น เนื่องจากลดขั้นตอนการทำงานและลดโปรแกรมที่ต้องใช้ ดังนี้

วิธีที่ 1

ขั้นตอนที่ 1 เนื่องจากพารามิเตอร์ 'Diameter' และ 'Length' ไม่สามารถนำมาคำนวณในสมการ Friction Loss ได้ ดังนั้นผู้ศึกษาจึงทำการย้ายข้อมูลของค่าดังกล่าวเข้าสู่พารามิเตอร์ตัวใหม่โดยการเขียนโค้ดในซอฟท์แวร์ Dynamo จากนั้นจึงรันโค้ดเพื่อให้ข้อมูลถูกย้ายจากโปรแกรม Autodesk Revit ไปยังโปรแกรม Microsoft Excel และย้ายข้อมูลกลับมายังโปรแกรม Autodesk Revit อีกครั้งหนึ่งเพื่อให้ได้พารามิเตอร์ที่ผู้ศึกษาสร้างขึ้นมาใหม่ นั่นก็คือ 'Diameter' และ 'Length' ซึ่งเป็นพารามิเตอร์ที่อยู่ใน Pipe Schedule

ขั้นตอนที่ 2 ย้ายข้อมูลค่าค่าพารามิเตอร์ 'Roughness coefficient' และ 'Flow rate' เข้ามายังโปรแกรม Auto-desk Revit เพื่อหาค่าความเร็วของของไหลภายในท่อ แต่พารามิเตอร์ 'Velocity' ที่อยู่ในรูปของสูตร (Formula) ไม่สามารถย้ายข้อมูลกลับไปยังโปรแกรม Microsoft Excel ได้ จึงต้องคำนวณหาค่าความเร็วของของไหลภายในท่อในซอฟท์แวร์ Dynamo โดยตรง โดยการนำสมการที่ถูกแปลงหน่วยจากมิลลิเมตรต่อวินาที (mm/s) เป็นเมตรต่อวินาที (m/s) แล้ว ใส่ลงในพารามิเตอร์ตัวใหม่ที่มีชื่อว่า 'Velocity_2' ดังแสดงในรูปที่ 3.26

<Pipe Schedule>

A	B	C	D	E	F	G
Family Type and Name	Diameter	Length	Roughness coeffic	Flow Rate	Velocity	Velocity_2
Pipe Types: SAN-CW-CD-SP-D	100	2025.26181	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	168	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	2434.175697	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	150	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	790	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	33.2121	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	4974.073882	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	187.614063	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	13223.290783	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	2533.648562	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	9358.647809	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	8846.611939	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	8894.576391	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	22325.944359	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	3280.17225	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	12141.50605	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	1006.841754	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	70	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	65.441559	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	1675	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	434.274353	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	431.037311	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	269.688336	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	3664.285621	120	4.1	0.521761	0.521761
Pipe Types: SAN-CW-CD-SP-D	100	3664.285621	120	4.1	0.521761	0.521761

รูปที่ 3.26 แสดงข้อมูลต่างๆ ใน Pipe Schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 3 หาค่า Friction Head ในโปรแกรม Microsoft Excel

เนื่องจากโปรแกรม Autodesk Revit ไม่สามารถคำนวณสมการซับซ้อนได้ ซึ่งแตกต่างจากโปรแกรม Microsoft Excel ดังนั้นเมื่อคำนวณค่า Friction Head และ Head Loss ทั้งหมดในโปรแกรม Microsoft Excel แล้วจึงย้ายข้อมูลกลับมายังโปรแกรม Autodesk Revit โดยการเขียนโค้ดในซอฟท์แวร์ Dynamo โดยใช้พารามิเตอร์ 'Friction' และ 'Head Loss' มีหน่วยเป็นเมตร (m) และฟุต (ft) ตามลำดับ (ในขณะนี้ผู้ศึกษาได้ Dynamic Head Loss ของท่อทุกเส้นในสถานีแล้ว)

ขั้นตอนที่ 4 หาค่า Static Head Loss

เริ่มจากการใช้พารามิเตอร์ 'Invert Elevation' ซึ่งพารามิเตอร์นี้มีหน้าที่บอกระดับความสูงของท่อแต่ละท่อ ในที่นี้ผู้ศึกษาจึงใช้แถบเครื่องมือ Filter และ Sort/Grouping ในหน้าต่าง Schedule Properties เพื่อหาระดับความสูงของท่อที่สูงที่สุดและต่ำที่สุดของเส้นทางทางเดินของท่อ ซึ่งใช้พารามิเตอร์ 'Mark' แสดงถึงเส้นทางทางเดินของท่อ และ 'Number' แสดงถึงลำดับของท่อในแต่ละเส้นทาง จากนั้นจึงได้กลุ่มของท่อจากแต่ละเส้นทางทางเดินของท่อที่มีการจัดลำดับของท่อจากระดับต่ำที่สุดไปจนถึงระดับสูงที่สุด โดยสร้างเป็น Schedule ใหม่ขึ้นมาที่มีชื่อว่า Pipe Selecting Data Schedule ดังรูปที่ 3.27 (ซึ่งทั้ง Pipe Schedule และ Pipe Selecting Data Schedule มีความเกี่ยวข้องกัน เนื่องจาก Pipe Selecting Data Schedule เป็น Schedule ที่ถูกทำซ้ำขึ้นมาจาก Pipe Schedule ฉะนั้นหากแก้ไขข้อมูลใน Pipe Schedule แล้วข้อมูลใน Pipe Selecting Data Schedule จะเปลี่ยนไปเช่นกัน) จากนั้นจึงสร้างพารามิเตอร์ที่ใช้สำหรับการคำนวณ (Calculated Parameter) ขึ้นมาโดยใช้ชื่อว่า 'Static Loss' และเพิ่มสมการเพื่อนำค่าของท่อระดับสูงลบกับท่อระดับต่ำที่สุดเข้าไปในซอฟท์แวร์ Dynamo จากนั้นจึงย้ายข้อมูลกลับมายังโปรแกรม Autodesk Revit ในขั้นตอนนี้จะได้ค่า Static Head Loss ของเส้นทางทางเดินของท่อที่เราต้องการจะอยู่ ณ ท่อระดับต่ำที่สุด

<Pipe Selecting Data Schedule>

B	C	D	E	F	G	H	I	J	K	L	M
Length_Test	Roughness coeffic	Flow Rate_Test	Velocity_Test	Velocity_2	Frction_Test	Head Loss_Test	Invert Elevation	Static Loss_Test	Total Head_Test	Total head	Mark
2025.26181	120	4.1	0.521761	0.521761	0.426514	0.009296	76545	34.502815	34.51211	34.51211	DSP-2 1
150	120	4.1	0.521761	0.521761	0.426514	0.000688	78755	32.294085	32.294773	32.294773	DSP-2 2
187.614063	120	4.1	0.521761	0.521761	0.426514	0.000134	79085	31.967481	31.967615	31.967615	DSP-2 3
790	120	4.1	0.521761	0.521761	0.426514	0.003626	79320	31.728717	31.732343	31.732343	DSP-2 4
168	120	4.1	0.521761	0.521761	0.426514	0.000771	79320	31.728717	31.729488	31.729488	DSP-2 5
2434.175697	120	4.1	0.521761	0.521761	0.426514	0.011172	79500	31.550867	31.562039	31.562039	DSP-2 6
3280.17225	120	4.1	0.521761	0.521761	0.426514	0.004391	82000	29.050847	29.055238	29.055238	DSP-2 7
13223.290783	120	4.1	0.521761	0.521761	0.426514	0.000861	82000	29.050847	29.051709	29.051709	DSP-2 8
33.2121	120	4.1	0.521761	0.521761	0.426514	0.004149	82000	29.050847	29.054996	29.054996	DSP-2 9
4974.073682	120	4.1	0.521761	0.521761	0.426514	0.000435	82165	28.883303	28.883738	28.883738	DSP-2 10
2533.648562	120	4.1	0.521761	0.521761	0.426514	0.005594	87205	23.843386	23.84898	23.84898	DSP-2 11
9358.647809	120	4.1	0.521761	0.521761	0.426514	0.014173	87205	23.843386	23.857558	23.857558	DSP-2 12
8846.611939	120	4.1	0.521761	0.521761	0.426514	0.011629	87205	23.843336	23.854965	23.854965	DSP-2 13
8894.576391	120	4.1	0.521761	0.521761	0.426514	0.042954	87205	23.843336	23.88629	23.88629	DSP-2 14
22325.944359	120	4.1	0.521761	0.521761	0.426514	0.040604	87205	23.843336	23.88394	23.88394	DSP-2 15
1006.841754	120	4.1	0.521761	0.521761	0.426514	0.055727	87205	23.843336	23.899062	23.899062	DSP-2 16
12141.50605	120	4.1	0.521761	0.521761	0.426514	0.000912	87375	23.675792	23.676704	23.676704	DSP-2 17
70	120	4.1	0.521761	0.521761	0.426514	0.004621	99590	11.460136	11.464757	11.464757	DSP-2 18
65.441559	120	4.1	0.521761	0.521761	0.426514	0.005405	99590	11.460136	11.465541	11.465541	DSP-2 19
431.037311	120	4.1	0.521761	0.521761	0.426514	0.015062	99590	11.460136	11.475197	11.475197	DSP-2 20
3664.285621	120	4.1	0.521761	0.521761	0.426514	0.002298	99765	11.284286	11.286584	11.286584	DSP-2 21
269.688336	120	4.1	0.521761	0.521761	0.426514	0.000112	103505	7.54585	7.545962	7.545962	DSP-2 22
3664.285621	120	4.1	0.521761	0.521761	0.426514	0.002298	99765	11.284286	11.286584	11.286584	DSP-2 23
434.274353	120	4.1	0.521761	0.521761	0.426514	0.000068	99590	11.460136	11.460201	11.460201	DSP-2 24
1675	120	4.1	0.521761	0.521761	0.426514	0.060828	99590	11.460136	11.520963	11.520963	DSP-2 25

รูปที่ 3.27 แสดงข้อมูลใน Pipe Selecting Data Schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 5 สร้างพารามิเตอร์ใหม่ขึ้นมาที่มีชื่อว่า ‘Total Head’

เพื่อรับค่าที่ได้จากการคำนวณในซอฟท์แวร์ Dynamo โดยการนำค่า ‘Static Loss’ และ ‘Head Loss’ มารวมกัน จากนั้นจึงนำค่าข้อมูลทั้งหมดเข้าไปในโปรแกรม Microsoft Excel อีกครั้งหนึ่ง (ไฟล์ที่ได้จะมีนามสกุล .txt ดังนั้นผู้ศึกษาจึงสามารถปรับรูปร่างของตารางให้ได้ตามที่ต้องการในภายหลัง)

ขั้นตอนที่ 6 สร้าง Schedule ใหม่ขึ้นมาโดยใช้ชื่อว่า ‘Pipe Fitting Schedule’ และ ‘Pipe Accessory Schedule’ ดังรูปที่ 3.28 และ 3.29 ตามลำดับ

โดยทั้งสอง Schedule นี้ ผู้ศึกษาได้ใส่ข้อมูลเช่นเดียวกับ Pipe Schedule ดังที่กล่าวไว้ข้างต้น (พารามิเตอร์ ‘Mark’ และ ‘Number’) เพื่อให้โปรแกรมเรียงลำดับท่อในแต่ละเส้นทางได้ จากนั้นจึงใช้เครื่องมือ Filter และ Sort/Grouping พารามิเตอร์ ‘System Name’ (เป็นพารามิเตอร์ที่มีอยู่ในโปรแกรมซึ่งใช้พารามิเตอร์นี้แสดงถึงชื่อและหมายเลขของบ่อพักน้ำที่ท่อเชื่อมต่อ) , ‘Mark’ และ ‘Number’ ดังนั้นจึงได้ Schedule ที่มีการเรียงลำดับของอุปกรณ์ข้อต่อ (Pipe Fitting) ในเส้นทางทางเดินของท่อแล้ว

<Pipe Fitting Schedule>

A	B	C	D	E	F	G	H
Family and Type	System Name	System Type	Type	Size	Mark	Number	Equipvalent Leng
TEE SAN With Groove: M_REDUCING TEE_	SW 27	P_Condensate(pu	M_REDUCING TEE_WELDED	100-100-100	DSP-2 5-6	2	13.8
M_Cap With Groove: Standard	SW 27	P_Condensate(pu	Standard	100			
TEE SAN With Groove: M_REDUCING TEE_	SP 6	P_Waste(pumped)	M_REDUCING TEE_WELDED	150-150-150			40.4
TEE SAN With Groove: M_REDUCING TEE_	SW 27	P_Condensate(pu	M_REDUCING TEE_WELDED	100-100-100	DSP-2 3-4	1	13.8
TEE SAN With Groove: M_REDUCING TEE_	SW 27	P_Condensate(pu	M_REDUCING TEE_WELDED	100-100-100			13.8
TEE SAN With Groove: M_REDUCING TEE_	SP 6	P_Waste(pumped)	M_REDUCING TEE_WELDED	150-150-150			40.4
M_Cap With Groove: Standard	SP 13	P_Waste(pumped)	Standard	150			
TEE SAN With Groove: M_REDUCING TEE_	SP 13	P_Waste(pumped)	M_REDUCING TEE_WELDED	150-150-150			40.4
Elbow - Butt Welded - CS.Irfa: Standard	V 7	P_Vent	Standard	80-80			7.7
Elbow - Butt Welded - CS.Irfa: Standard	V 7	P_Vent	Standard	80-80			7.7
Elbow - Butt Welded - CS.Irfa: Standard	V 8	P_Vent	Standard	80-80			7.7
Elbow - Butt Welded - CS.Irfa: Standard	V 8	P_Vent	Standard	80-80			7.7
Reducer Concentric - Butt Welded - CS: St	CW 83	P_Supply_Pipe	Standard	100-80			
Transition - Welded - Generic: Standard	CW 75	P_Supply_Pipe	Standard	100-80			
Reducer Concentric - Butt Welded - CS: St	CW 58	P_Supply_Pipe	Standard	100-80			
Transition - Welded - Generic: Standard	CW 47	P_Supply_Pipe	Standard	100-80			
Elbow SAN With Groove: Standard 2	CW 29	P_Supply_Pipe	Standard 2	100-100		1	5.4
Elbow SAN With Groove: Standard 2	CW 26	P_Supply_Pipe	Standard 2	100-100			5.4
TEE SAN With Groove: M_REDUCING TEE_	CW 112	P_Supply_Pipe	M_REDUCING TEE_WELDED	100-100-100			13.8
M_Elbow - GALVANIZED - Sch 40 I: Stand	CW 128	P_Supply_Pipe	Standard	100-100			10.1
Tee Branch Reduced - Plain - ABS: Standa	V 12	P_Vent	Standard	100-100-100			12.4
M_Elbow - GALVANIZED - Sch 40: M_Elbo	V 12	P_Vent	M_Elbow-Cast Iron	100-100			
M_Elbow - GALVANIZED - Sch 40: M_Elbo	V 12	P_Vent	M_Elbow-Cast Iron	100-100			
M_Elbow - GALVANIZED - Sch 40 I: Stand	W 77	P_Waste	Standard	150-150			12.6
M_Elbow - GALVANIZED - Sch 40 I: Stand	W 225	P_Waste	Standard	50-50			5.2
M_Cap With Groove: Standard	CW 164	P_Supply_Pipe	Standard	100			

รูปที่ 3.28 แสดงข้อมูลใน Pipe Fitting Schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<Pipe Accessory Schedule>

A	B	C	D	E	F	G
Type	Family	System Name	Equipvalent Length	Family Namel	Markl	No.
ø150 GATE VALVE	ø150 GATE VALVE	SP 6,SP 7	3.4	Ø150 GATE VALV		
ø150 GATE VALVE	ø150 GATE VALVE	SP 6,SP 12	3.4	Ø150 GATE VALV		
ø150 GATE VALVE	ø150 GATE VALVE	SP 6,SP 17	3.4	Ø150 GATE VALV		
ø150 GATE VALVE	ø150 GATE VALVE	SP 13,SP 14	3.4	Ø150 GATE VALV		
ø150 GATE VALVE	ø150 GATE VALVE	SP 13,SP 19	3.4	Ø150 GATE VALV		
ø150 GATE VALVE	ø150 GATE VALVE	SP 13,SP 23	3.4	Ø150 GATE VALV		
ø150 CHECK VALVE	ø150 CHECK VALVE	SP 5,SP 7	3.4	Ø150 CHECK VAL		
ø150 CHECK VALVE	ø150 CHECK VALVE	SP 5,SP 12	3.4	Ø150 CHECK VAL		
ø150 CHECK VALVE	ø150 CHECK VALVE	SP 1,SP 14	3.4	Ø150 CHECK VAL		
ø150 CHECK VALVE	ø150 CHECK VALVE	SP 9,SP 19	3.4	Ø150 CHECK VAL		
ø100 GATE VALVE	ø100 GATE VALVE	SW 27,SW 25	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	SW 27,SW 44	2.7	Ø100 GATE VALV	DSP-2	2
ø100 GATE VALVE	ø100 GATE VALVE	SW 27,SW 50	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	CW 108,CW 164	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	CW 102,CW 164	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE		2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE		2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	CD (PUMPED) 4,CD (PUMP	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	CD (PUMPED) 4,CD (PUMP	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	CD (PUMPED) 4,CD (PUMP	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	SW (PUMPED) 4,SW (PUM	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	SW (PUMPED) 8,SW (PUM	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	SW (PUMPED) 12,SW (PU	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	SW 8	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	SW 30	2.7	Ø100 GATE VALV		
ø100 GATE VALVE	ø100 GATE VALVE	SW 7	2.7	Ø100 GATE VALV		

รูปที่ 3.29 แสดงข้อมูลใน Pipe Accessory Schedule

ขั้นตอนที่ 7 เนื่องจากค่า Friction Loss ในอุปกรณ์ข้อต่อ (Pipe Fitting) ไม่สามารถคำนวณได้เช่นเดียวกับการคำนวณค่าต่างๆ ของท่อโดยใช้โปรแกรม Autodesk Revit แต่สามารถเทียบค่าได้จากตาราง Equivalent Length ดังนั้นผู้ศึกษาจึงใช้ Array 'Lookup Table' เพื่อเชื่อมโยงตารางนี้ในโปรแกรม Microsoft Excel และ Family ที่ต้องการในโปรแกรม Autodesk Revit (ในขั้นตอนนี้จะใช้ Family Pipe Fitting และ Pipe Accessory)

ขั้นตอนที่ 8 สร้างตาราง Equivalent Length

ตารางนี้จะมีทั้ง Equivalent Length ของอุปกรณ์ข้อต่อ (Pipe Fitting) เช่น Elbow, Reducing Tee เป็นต้น และ Pipe Accessory ในโปรแกรม Microsoft Excel โดยแถวที่ 1 นั้นจะต้องเว้นว่างเนื่องจากเป็นรูปแบบที่ทำให้โปรแกรม Autodesk Revit อ่านค่าจากตารางได้และในแถวถัดไปให้ใส่รูปแบบดังนี้

ชื่อพารามิเตอร์##ประเภทของพารามิเตอร์##หน่วยของพารามิเตอร์

โดยประเภทของพารามิเตอร์ต้องสอดคล้องกับพารามิเตอร์ที่ผู้ศึกษาสร้างขึ้นในโปรแกรม Autodesk Revit ตัวอย่างเช่น *DOPF##length##millimeters* ดังรูปที่ 3.30 โดย DOPF แสดงถึงชื่อพารามิเตอร์ของเส้นผ่านศูนย์กลางภายในของอุปกรณ์ข้อต่อ (Diameter of Pipe Fitting) ซึ่งเป็นความยาว (Length) และมีหน่วยเป็นมิลลิเมตร (mm.) ทั้งนี้ต้องบันทึกไฟล์โดยใช้นามสกุล .csv (Comma Delimited) เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	A	B	C	D	E	F	G
1		ND##length##millimeters	Ang##angle##degrees	SOD##length##inches	CtE##length##inches	SLgt##length##inches	DOPF##length##millimeters
2	0.5	15	90	1.094	1.25	0.75	1.6
3	0.75	20	90	1.281	1.281	0.719	2.1
4	1	25	90	1.594	1.563	0.875	2.6
5	1.25	32	90	2	2.125	1.25	3.5
6	1.5	40	90	2.25	2.313	1.313	4
7	2	50	90	2.688	2.406	1.156	5.2
8	2.5	65	90	3.344	3.25	1.75	6.2
9	3	80	90	3.969	3.813	2	7.7
10	4	100	90	5	4.375	2.063	10.1
11	6	150	90	7.25	7	3.5	12.6
12	8	200	90	9.375	8.531	4	15.2
13	0.5	15	45	1.094	1.063	0.75	0.8
14	0.75	20	45	1.313	1.031	0.719	1.1
15	1	25	45	1.656	1.438	1.125	1.4
16	1.25	32	45	2	1.75	1.25	1.8
17	1.5	40	45	2.25	1.906	1.313	2.2
18	2	50	45	2.75	2.125	1.375	2.8
19	2.5	65	45	3.313	2.469	1.781	3.3
20	3	80	45	3.969	2.844	2	4.1
21	4	100	45	5	3.156	2.063	5.4
22	6	150	45	7.25	5.25	3.5	6.7
23	8	200	45	9.375	6.844	4.5	8.1

รูปที่ 3.30 แสดงตาราง Equivalent Length ของอุปกรณ์ข้อต่อ (Elbow)

	A	B	C	D	E	F	G	H	I	J
1		ND##length##millimeters	Ang##angle##degrees	SOD##length##inches	CtE##length##inches	SLgt##length##inches	DOPF##length##millimeters	DOPA##length##millimeters	DOPAI##length##millimeters	DOPAII##length##millimeters
2	0.5	15	90	1.094	1.25	0.75	1.6	0.4	5.2	0
3	0.75	20	90	1.281	1.281	0.719	2.1	0.6	6.9	0
4	1	25	90	1.594	1.563	0.875	2.6	0.7	8.7	0
5	1.25	32	90	2	2.125	1.25	3.5	0.9	11.5	0
6	1.5	40	90	2.25	2.313	1.313	4	1.1	13.4	0
7	2	50	90	2.688	2.406	1.156	5.2	1.4	17.2	7.8
8	2.5	65	90	3.344	3.25	1.75	6.2	1.7	20.6	9.3
9	3	80	90	3.969	3.813	2	7.7	2	25.5	11.5
10	4	100	90	5	4.375	2.063	10.1	2.7	33.6	15.1
11	6	150	90	7.25	7	3.5	12.6	3.4	42.1	18.9
12	8	200	90	9.375	8.531	4	15.2	4	50.5	22.7
13	0.5	15	45	1.094	1.063	0.75	0.8			
14	0.75	20	45	1.313	1.031	0.719	1.1			
15	1	25	45	1.656	1.438	1.125	1.4			
16	1.25	32	45	2	1.75	1.25	1.8			
17	1.5	40	45	2.25	1.906	1.313	2.2			
18	2	50	45	2.75	2.125	1.375	2.8			
19	2.5	65	45	3.313	2.469	1.781	3.3			
20	3	80	45	3.969	2.844	2	4.1			
21	4	100	45	5	3.156	2.063	5.4			
22	6	150	45	7.25	5.25	3.5	6.7			
23	8	200	45	9.375	6.844	4.5	8.1			
24										

รูปที่ 3.31 แสดงตาราง Equivalent Length ของ Pipe Accessory

<u>หมายเหตุ</u> จากรูปที่ 3.31 DOPA หมายถึงลิ้นวาล์ว (Gate Valve), DOPAI หมายถึงวาล์วกันกลับ (Check Valve) และ DOPAII หมายถึงวาล์วปีกผีเสื้อ (Butterfly Valve)

<u>ขั้นตอนที่ 9</u> เลือกอุปกรณ์ข้อต่อที่ต้องการในโปรแกรม Autodesk Revit จากนั้นจึงสร้างพารามิเตอร์ ‘Equivalent Length’ และ ‘Lookup Table Name’ ขึ้นใหม่ โดย ‘Equivalent Length’ จะอยู่รูปแบบของ Type และเลือก Type of parameter เป็น Length และ ‘Lookup Table Name’ เลือกรูปแบบเป็น Instant และเลือก Type of parameter เป็น Text เท่านั้น ต่อมาจึงนำตารางเข้าไปในโปรแกรม Autodesk Revit โดยเลือกที่ Manage Lookup Table และใส่สูตร (Formula) ลงใน Equivalent Length โดยมีรูปแบบคือ

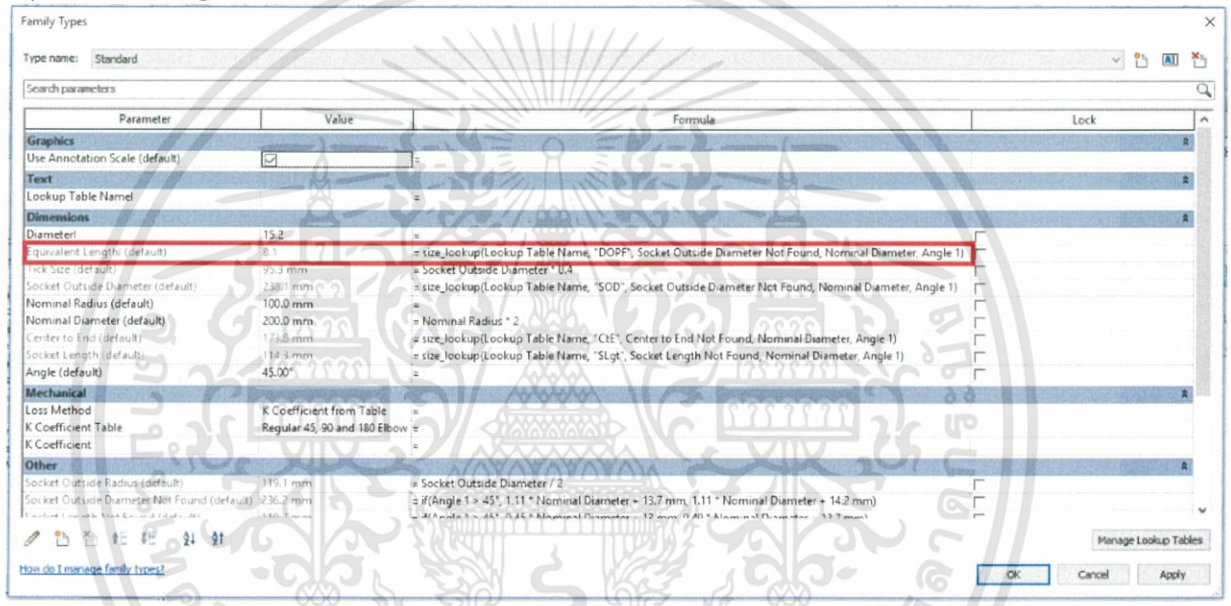
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

size_lookup(ชื่อพารามิเตอร์ของตารางที่ต้องการ, "ชื่อพารามิเตอร์ที่ต้องการค้นหาค่า",ชื่อพารามิเตอร์เมื่อหาค่าที่ต้องการไม่พบ,ชื่อพารามิเตอร์ที่เกี่ยวข้อง)

ตัวอย่างจากรูปที่ 3.32

size_lookup(Lookup Table Neme, "DOPF",R,Nominal Diameter1,Nominal Diameter 2)

หมายความว่าให้โปรแกรม Autodesk Revit หาข้อมูลจากไฟล์ตาราง Lookup Table Name ในคอลัมน์ 'DOPF' เมื่อโปรแกรมไม่สามารถอ่านค่าจากคอลัมน์นี้ได้หรือไม่พบค่าข้อมูล โปรแกรมจะอ่านค่าจากพารามิเตอร์ 'R' ทันทีและถ้าหากค่าของพารามิเตอร์ 'Nominal Diameter1' และ 'Nominal Diameter2' เปลี่ยนแล้วค่าที่โปรแกรมอ่านออกมาได้ย่อมเปลี่ยนไปด้วยเช่นกัน (หากใส่รูปแบบที่ถูกต้องค่า Equivalent Length จะเปลี่ยนค่าโดยอัตโนมัติเมื่อเปลี่ยนค่าที่เกี่ยวข้อง)



รูปที่ 3.32 แสดงหน้าต่าง Family Type

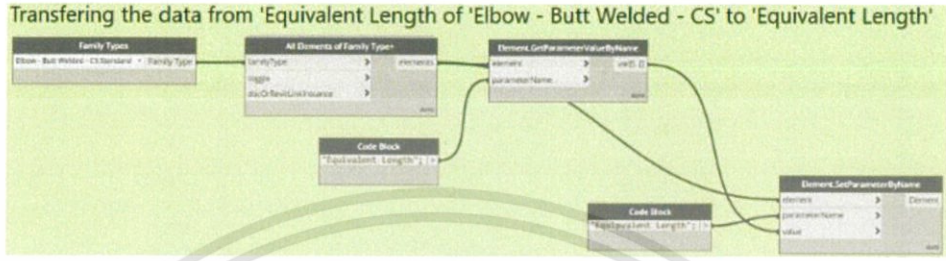
ขั้นตอนที่ 10 เชื่อมโยงตารางกับ Family ทุกตัวที่ต้องการ รวมถึงอุปกรณ์ข้อต่อ (Pipe Fitting) และ Pipe Accessory

ขั้นตอนที่ 11 เมื่อได้ค่า Equivalent Length ของอุปกรณ์ข้อต่อ (Pipe Fitting) และ Pipe Accessory แล้ว แต่ข้อมูลค่า Equivalent Length ไม่สามารถนำเข้าไปใน Schedule ได้ เนื่องจากพารามิเตอร์ 'Equivalent Length' ไม่ปรากฏใน Properties Browser ดังนั้นจึงต้องย้ายข้อมูล Equivalent Length เข้าไปในพารามิเตอร์ตัวใหม่โดยใช้ชื่อเดียวกันว่า 'Equivalent Length' โดนการเขียนและรันโค้ดผ่านซอฟท์แวร์ Dynamo อีกครั้งหนึ่ง

หมายเหตุ จากรูปที่ 3.33 เป็นการแสดงตัวอย่างการถ่ายโอนข้อมูลของตาราง Equivalent Length จากโปรแกรม Microsoft Excel ไปยังโปรแกรม Revit ของข้อต่อชนิด Butt Welded-CS ซึ่งในการใช้งานอาจมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนแปลงชนิดของข้อต่อหรือข้อต่อสามทางได้ เช่น ข้อต่อชนิด Galvanized-Cast Iron, ข้อต่อสามทางแบบมีร่อง (Tee Branch with groove), ข้อต่อสามทางลดชนิด Plain-ABS:standard เป็นต้น



รูปที่ 3.33 แสดงแผนภาพ Dynamo แสดงการถ่ายโอนข้อมูลของตาราง Equivalent Length จาก Excel ไปยัง Revit ของข้อต่อชนิด Butt-Welded-CS

ขั้นตอนที่ 12 สร้างพารามิเตอร์ ‘Mark’ และ ‘Number’

เพื่อทำการเรียงลำดับและแยกระบบ ในส่วนนี้เราจะได้ ‘Pipe Fitting Selecting Data Schedule’ และ ‘Pipe Accessory Selecting Data Schedule’ ดังรูปที่ 3.34 และ 3.35 ตามลำดับ

จากรูปที่ 3.34 ค่า Equivalent Length บางค่าอาจไม่พบค่าเนื่องจากอุปกรณ์ชิ้นนั้นไม่ใช่อุปกรณ์ข้อต่อ (Pipe Fitting) ที่อยู่ในระบบท่อ แต่ถูกจัดกลุ่มให้อยู่ใน Category Pipe Fitting

<Pipe Fitting Selecting Data Schedule>

A	B	C	D	E	F	G
System Name	System Type	Type	Size	Mark	Number	Equipvalent Length
SW 27	P_Condensate(pu	M_REDUCING TEE_	100-100-100	DSP-2 3-4	1	13.8
SW 27	P_Condensate(pu	M_REDUCING TEE_	100-100-100	DSP-2 5-6	2	13.8
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 6-7	3	5.4
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 7-8	4	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 8-9	5	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 9-10	6	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 10-11	7	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 11-12	8	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 12-13	9	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 13-14	10	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 14-15	11	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 15-16	12	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 16-17	13	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 17-18	14	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 18-19	15	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 19-20	16	5.4
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 20-21	17	5.4
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 21-22	18	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 22-23	19	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 23-24	20	10.1
SW 27	P_Condensate(pu	Standard	100-100	DSP-2 24-25	21	10.1

รูปที่ 3.34 แสดงข้อมูลใน Pipe Fitting Selecting Data Schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<Pipe Accessory Selecting Data Schedule>

A	B	C	D	E	F	G
Type	Family	System Type	System Name	Equipvalent Length	Mark	Number
ø100 CHECK VAL	ø100 CHECK VALV	P_Condensate(pu	SW 44,SW 21	33.6	DSP-2	1
ø100 GATE VALV	ø100 GATE VALVE	P_Condensate(pu	SW 27,SW 44	2.7	DSP-2	2

รูปที่ 3.35 แสดงข้อมูลใน Pipe Accessory Selecting Data Schedule

ขั้นตอนที่ 13 ในขั้นตอนของการนำข้อมูล Schedule ที่ได้ออกมาในรูปแบบของไฟล์ตาราง Excel ผู้ศึกษาจะแบ่งออกเป็น 2 กรณีคือ

13.1) กรณีของ Pipe Schedule, Pipe Fitting Schedule และ Pipe Accessory Schedule สามารถย้ายข้อมูลเข้าไปในโปรแกรม Microsoft Excel โดยผ่านการรันโค้ดในซอฟท์แวร์ Dynamo

13.2) กรณีของ Pipe Selecting Data Schedule, Pipe Fitting Selecting Data Schedule และ Pipe Accessory Selecting Data Schedule สามารถย้ายข้อมูลเข้าไปในโปรแกรม Microsoft Excel โดยการใช้คำสั่ง Export, Report และ Schedule ตามลำดับ

ขั้นตอนที่ 14 คำนวณหากำลังของปั๊มน้ำ

เมื่อนำค่าการสูญเสียพลังงาน ได้แก่ Friction Head Loss, Static Loss และ Head Loss ที่ได้จากการคำนวณในตารางรวมกับค่า Outer Pressure แล้ว จากนั้นจึงนำค่าที่ได้คำนวณหากำลังโดยใช้สมการที่ 2.3 โดยกำหนดให้มอเตอร์มีประสิทธิภาพ 80% และปั๊มมีประสิทธิภาพ 60% จากนั้นจึงนำกำลังที่ได้จากการคำนวณเลือกขนาดที่เหมาะสมของมอเตอร์ของปั๊มน้ำ

วิธีที่ 2

ขั้นตอนในวิธีที่สองนี้สามารถทำได้ในซอฟท์แวร์ Dynamo โดยไม่จำเป็นต้องโอนข้อมูลไป-กลับระหว่างโปรแกรม Microsoft Excel และโปรแกรม Autodesk Revit ซึ่งวิธีดังกล่าวนี้เป็นวิธีที่ทำให้เสียเวลาในการทำงานเป็นอย่างมากเนื่องจากเป็นการที่ซ้ำซ้อนกันระหว่างโปรแกรม โดยในวิธีที่สองนี้มีวิธีการทำที่เหมือนกัน นั่นคือการทำ Lookup Table ในอุปกรณ์ข้อต่อ (Pipe Fitting) และ Pipe Accessories ดังนั้นผู้ศึกษาจึงขอข้ามเนื้อหาวิธีการทำ Lookup Table และวิธีการลิงก์ Family กับ Lookup Table เพื่อหา Equivalent Length ต่อไปจะเป็นวิธีการโดยละเอียดของวิธีที่ 2

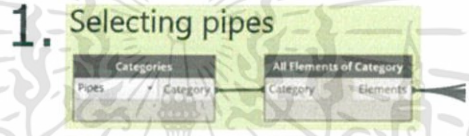
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก. วิธีการคำนวณในท่อ (Pipe)

ในที่นี้ผู้ศึกษาต้องสร้างพารามิเตอร์ใน Schedule โดยใช้ชื่อว่า Pipe Schedule เพื่อรับค่าที่ได้จากการคำนวณในซอฟท์แวร์ Dynamo ได้แก่พารามิเตอร์ 'Flow Rate', 'Roughness Coefficient', 'Velocity', 'Friction Head', 'Length1', 'Static Loss', 'Dynamic Loss' และ 'Total Head' โดยพารามิเตอร์ทั้งหมดที่กล่าวมานี้เป็นพารามิเตอร์รูปแบบ Instance ชนิด Text อีกทั้งผู้ศึกษาจะต้องสร้างพารามิเตอร์เช่นเดียวกับวิธีที่หนึ่ง นั่นคือพารามิเตอร์ 'Comment3' เพื่อแสดงประเภทของท่อและพารามิเตอร์ 'No' เพื่อแสดงลำดับของท่อ

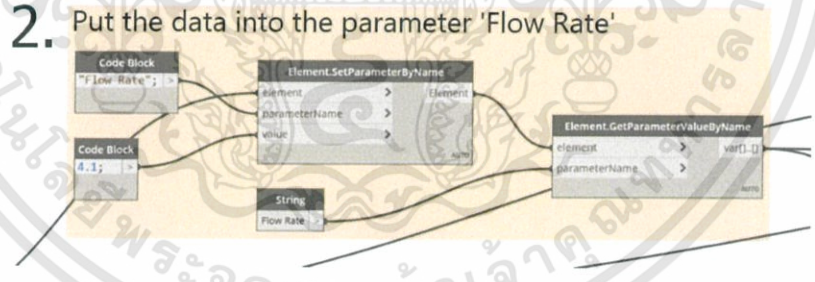
สามารถอธิบายแผนภาพ Dynamo อย่างละเอียดได้ดังนี้

ขั้นตอนที่ 1 เลือกท่อทั้งหมดที่อยู่ในระบบ



รูปที่ 3.36 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 1

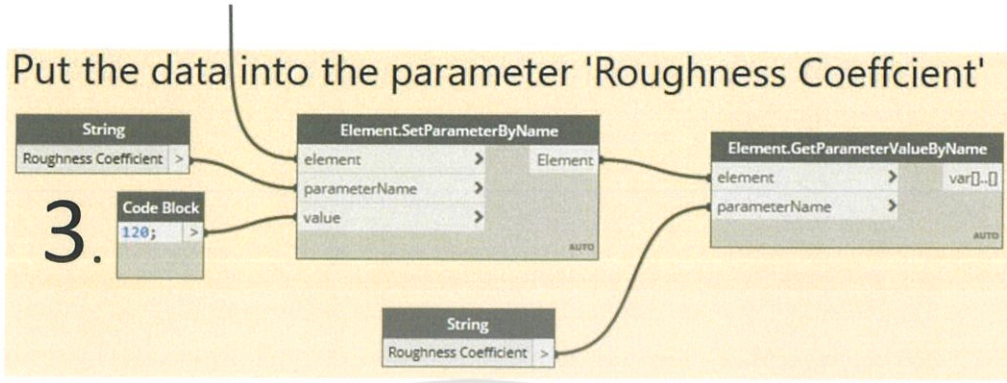
ขั้นตอนที่ 2 ใส่ข้อมูลลงไปในพารามิเตอร์ 'Flow Rate' ในกรณีนี้สมมติให้อัตราการไหลมีค่าเท่ากันในทุกท่อ นั่นคือ 4.1 ลิตรต่อวินาที (L/s)



รูปที่ 3.37 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 2

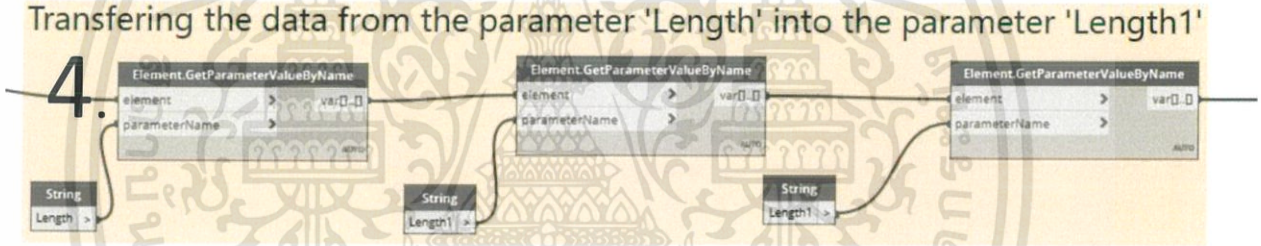
ขั้นตอนที่ 3 ใส่ข้อมูลลงไปในพารามิเตอร์ 'Roughness Coefficient' ซึ่งในกรณีนี้จะมีค่าเท่ากันหมด นั่นคือ 120

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



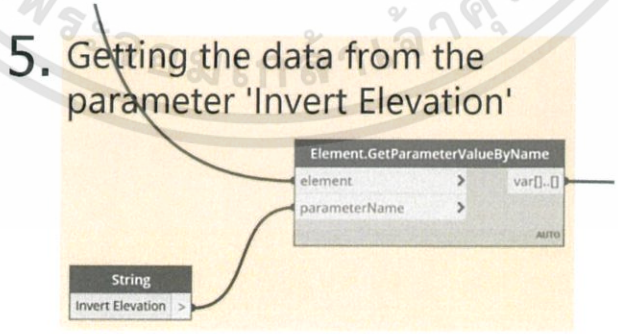
รูปที่ 3.38 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 3

<u>ขั้นตอนที่ 4</u> ย้ายข้อมูลจากพารามิเตอร์ ‘Length’ ไปยังพารามิเตอร์ ‘Length1’ ที่ถูกสร้างขึ้น เนื่องจากค่าในพารามิเตอร์ ‘Length’ ไม่สามารถนำมาคำนวณได้เนื่องจากมีหน่วยไม่ตรงกัน



รูปที่ 3.39 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 4

<u>ขั้นตอนที่ 5</u> ดึงข้อมูลจากพารามิเตอร์ ‘Invert Parameter’ เพื่อนำไปคำนวณหาค่า Static Loss



รูปที่ 3.40 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 5

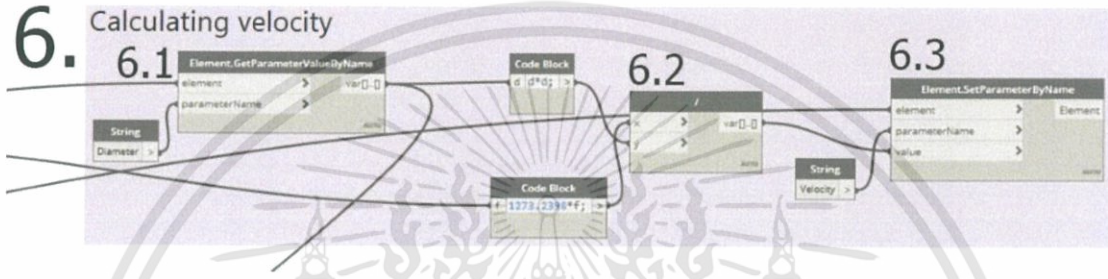
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 6 คำนวณหาค่าความเร็วและนำข้อมูลใส่ลงในพารามิเตอร์ ‘Velocity’

6.1) นำข้อมูลจากพารามิเตอร์ ‘Diameter’ ซึ่งเป็นข้อมูลของขนาดเส้นผ่านศูนย์กลางของท่อ

6.2) เนื่องจากค่าความเร็วของของไหลหาได้จากสมการ Q = V x A ดังนั้นผู้ศึกษาจึงต้องทำการเปลี่ยนหน่วยของขนาดเส้นผ่านศูนย์กลางจากหน่วยมิลลิเมตร (mm.) ให้กลายเป็นเมตร (m) และเปลี่ยนหน่วยของอัตราการไหลของของไหลจากเดิมคือลิตรต่อวินาที (L/s) เป็นลูกบาศก์เมตรต่อวินาที (m^3/s)

6.3) ได้ค่าความเร็วที่บรรจุอยู่ในพารามิเตอร์ ‘Velocity’

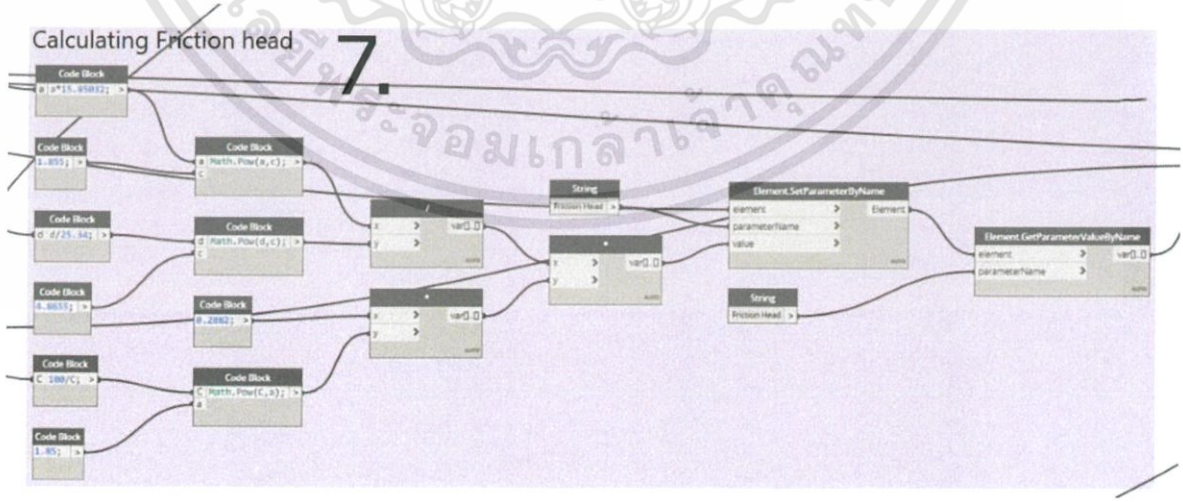


รูปที่ 3.41 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 6

ขั้นตอนที่ 7 คำนวณหาค่าปริมาณพลังงานสูญเสียของระบบท่อน้ำ (Friction Head) จากสมการ

$$f = 0.2082\left(\frac{100}{c}\right)^{1.85}\left(\frac{q^{1.85}}{d^{4.8655}}\right) \qquad (3.1)$$

ดังนั้นจึงต้องนำค่าข้อมูลที่อยู่ในพารามิเตอร์ ‘Diameter’, ‘Flow Rate’ และ ‘Roughness Coefficient’ มาเพื่อคำนวณ

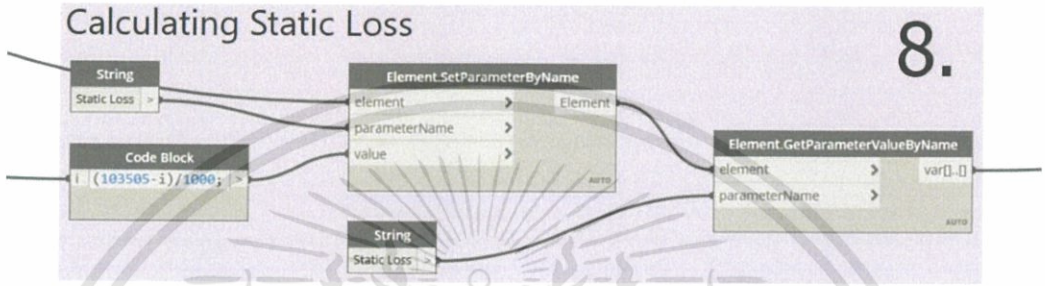


รูปที่ 3.42 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

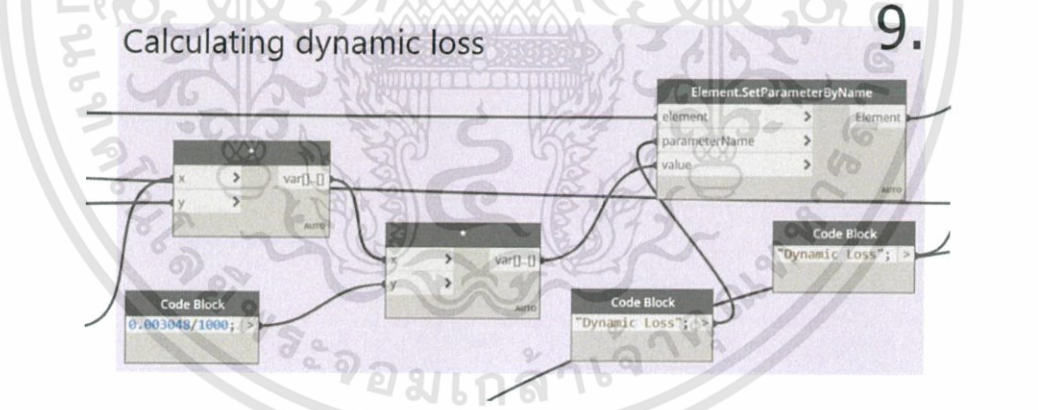
ขั้นตอนที่ 8 คำนวณหาค่า Static Loss โดยใช้ค่าที่ได้จากพารามิเตอร์ ‘Invert parameter’

ในขั้นตอนนี้จะต้องดูข้อมูลใน Schedule ในโปรแกรม Autodesk Revit และหาว่าท่อชิ้นไหนที่มีระดับสูงที่สุดในเส้นทางเดินของท่อ จากนั้นจึงนำค่านั้นลบกับค่าในพารามิเตอร์ ‘Invert parameter’ โดยที่ค่า Static Loss จะอยู่ที่ท่อระดับต่ำที่สุดของเส้นทางทางเดินของท่อ



รูปที่ 3.43 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 8

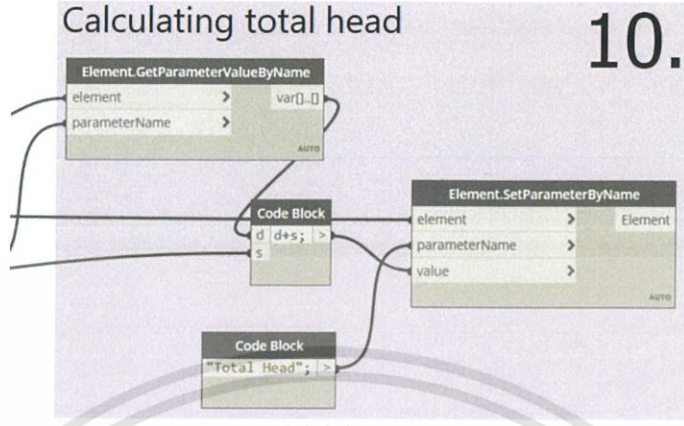
ขั้นตอนที่ 9 คำนวณหาค่า Dynamic Loss โดยการนำค่าของพารามิเตอร์ ‘Friction Head’ ที่ได้จากขั้นตอนที่ 7 คูณกับความยาวของท่อในหน่วยเมตร (m)



รูปที่ 3.44 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 9

ขั้นตอนที่ 10 เมื่อได้ค่าของพารามิเตอร์ ‘Static Loss’ และ ‘Dynamic Loss’ จากขั้นตอนที่ 8 และ 9 ตามลำดับแล้ว จึงนำมาคำนวณหา Total Head

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



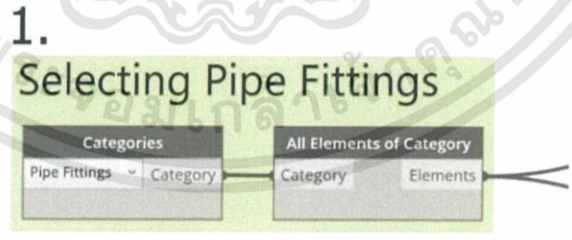
รูปที่ 3.45 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในท่อ (Pipe) ในขั้นตอนที่ 10

ข. วิธีการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting)

วิธีการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) มีวิธีการคำนวณคล้ายคลึงกับการคำนวณในท่อ (Pipe) แต่จะต้องเปลี่ยนชื่อพารามิเตอร์ทั้งหมดเพื่อไม่ให้ข้อมูลซ้อนทับกัน ดังนั้นผู้ศึกษาจึงต้องสร้างพารามิเตอร์ใหม่ขึ้นมาใน Schedule โดยใช้ชื่อว่า Pipe Fitting Schedule เพื่อรอค่าที่ได้จากการคำนวณในซอฟท์แวร์ Dynamo ได้แก่พารามิเตอร์ ‘Flow Rate’, ‘Roughness Coefficient’, ‘Velocity’, ‘Friction Head’ ‘Head Loss’ และ ‘Equivalent Length’ ซึ่งพารามิเตอร์ทั้งหมดที่กล่าวมานี้เป็นพารามิเตอร์ประเภท Number แบบ Instance อีกทั้งผู้ศึกษาจะต้องสร้างพารามิเตอร์เช่นเดียวกับวิธีที่หนึ่ง นั่นคือพารามิเตอร์ ‘Comment4’ เพื่อแสดงประเภทของระบบท่อและพารามิเตอร์ ‘No.’ เพื่อแสดงลำดับของท่อ

สามารถอธิบายแผนภาพ Dynamo ได้ดังนี้

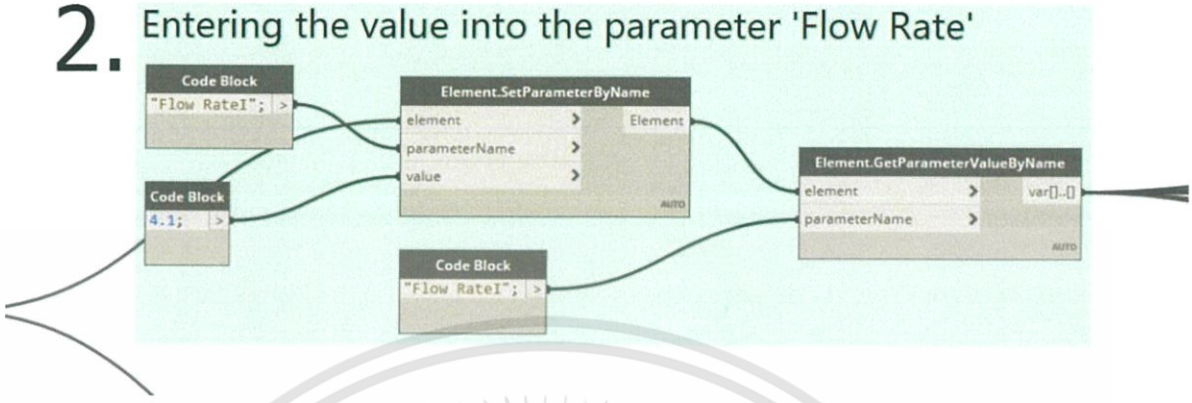
ขั้นตอนที่ 1 เลือกอุปกรณ์ข้อต่อ (Pipe Fitting) ทั้งหมดที่อยู่ในระบบ



รูปที่ 3.46 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) ในขั้นตอนที่ 1

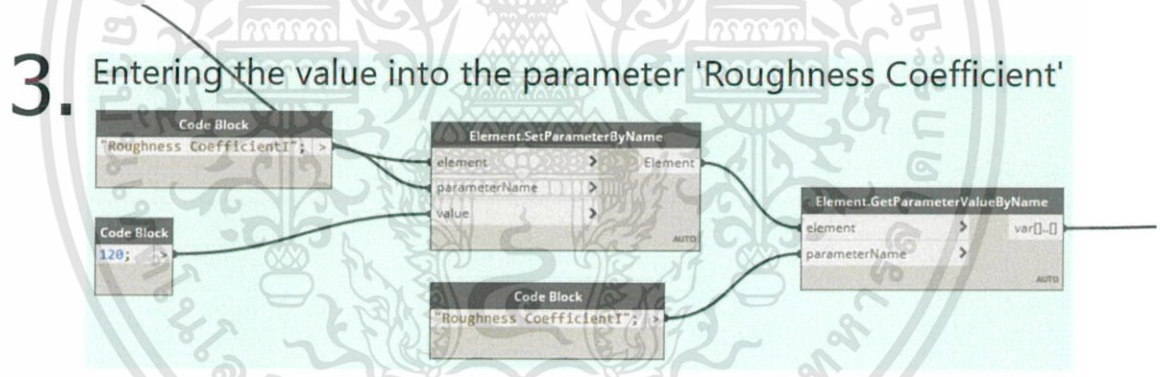
ขั้นตอนที่ 2 ใส่ข้อมูลลงไปในพารามิเตอร์ ‘Flow Rate’ ในกรณีนี้สมมติให้อัตราการไหลมีค่าเท่ากันในทุกท่อ นั่นคือ 4.1 ลิตรต่อวินาที (L/s)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.47 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) ในขั้นตอนที่ 2

<u>ขั้นตอนที่ 3</u> ใส่ข้อมูลลงในพารามิเตอร์ 'Roughness Coefficient' ซึ่งในกรณีนี้จะมีค่าเท่ากันทั้งหมด นั่นคือ 120

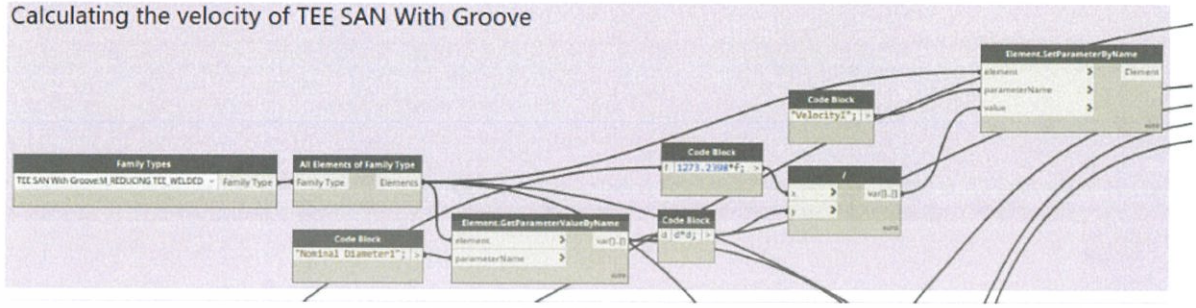


รูปที่ 3.48 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) ในขั้นตอนที่ 3

<u>ขั้นตอนที่ 4</u> คำนวณค่าความเร็วเพื่อใส่ลงในอุปกรณ์ข้อต่อ (Pipe Fitting) แต่ละชนิดจากสมการ Q = A x V โดยการดึงค่าข้อมูลเส้นผ่านศูนย์กลางของท่อจากพารามิเตอร์ 'Nominal Diameter' (ยกเว้นข้อต่อแบบ Tee เนื่องจากใช้พารามิเตอร์ 'Nominal Dameter1') และอัตราการไหลของของไหลจากขั้นตอนที่ 2 ผลลัพธ์ที่ได้จะมีหน่วยเป็นเมตรต่อวินาที (m/s)

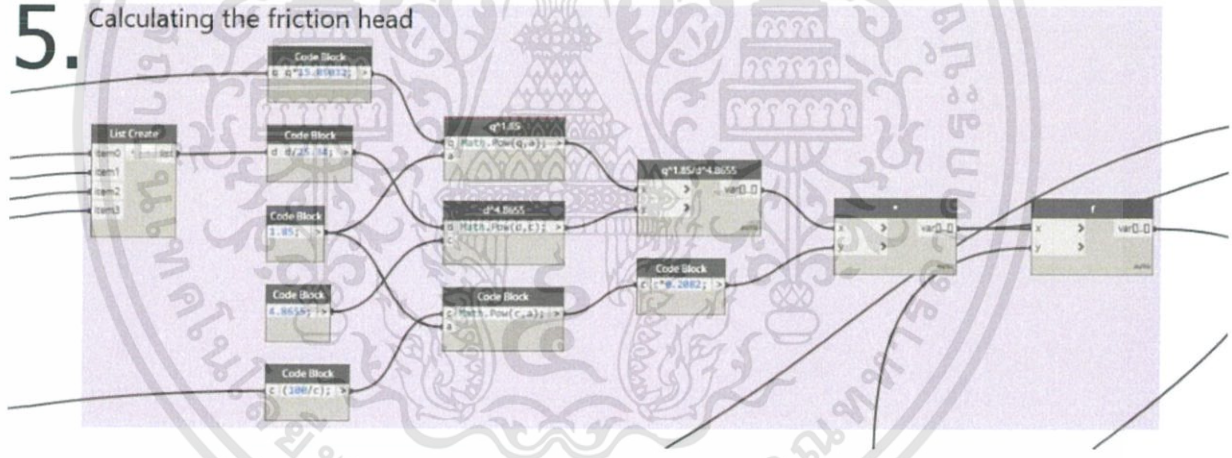
<u>หมายเหตุ</u> ในขั้นตอนนี้ผู้ศึกษาได้รวมข้อมูลเข้ากับ List เพื่อลดเส้นที่เชื่อมระหว่างโหนดแต่ไม่ประสบความสำเร็จเนื่องจากหากมีการคำนวณใน Array ที่เก็บข้อมูล ค่าที่ได้จะเปลี่ยนแปลงทันที ซึ่งในจุดนี้โปรแกรมจะเตือนขึ้นมาเป็นโหนด Error (สีเหลือง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.49 แสดงตัวอย่างภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) ชนิด Tee SAN With Groove ในขั้นตอนที่ 4

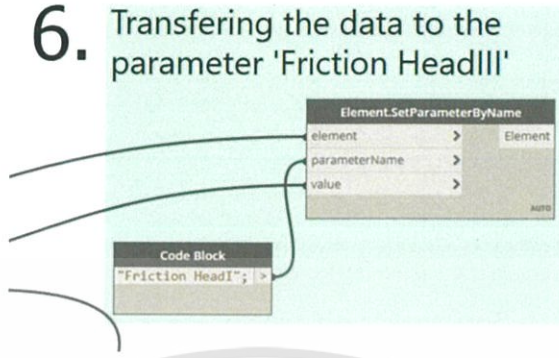
<u>ขั้นตอนที่ 5</u> นำข้อมูลที่ได้จากในขั้นตอนที่ 2,3 และ 4 คำนวณหาค่า Friction Head จากสมการที่ 3.1



รูปที่ 3.50 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) ในขั้นตอนที่ 5

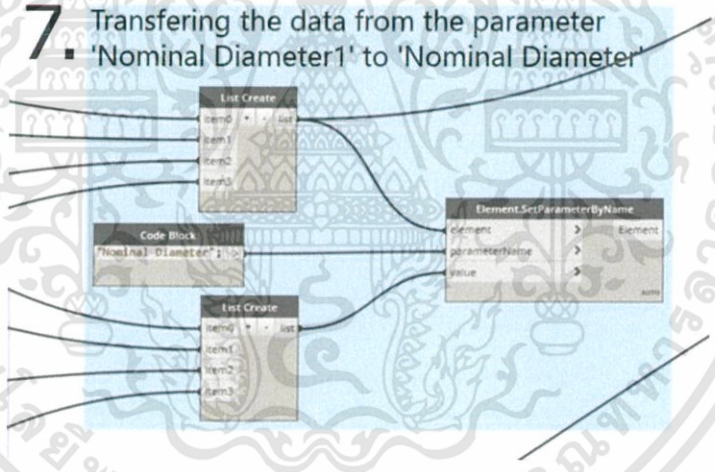
<u>ขั้นตอนที่ 6</u> จากนั้นจึงนำค่าข้อมูลที่ได้ใส่ลงในพารามิเตอร์ ‘Friction HeadIII’

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.51 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) ในขั้นตอนที่ 6

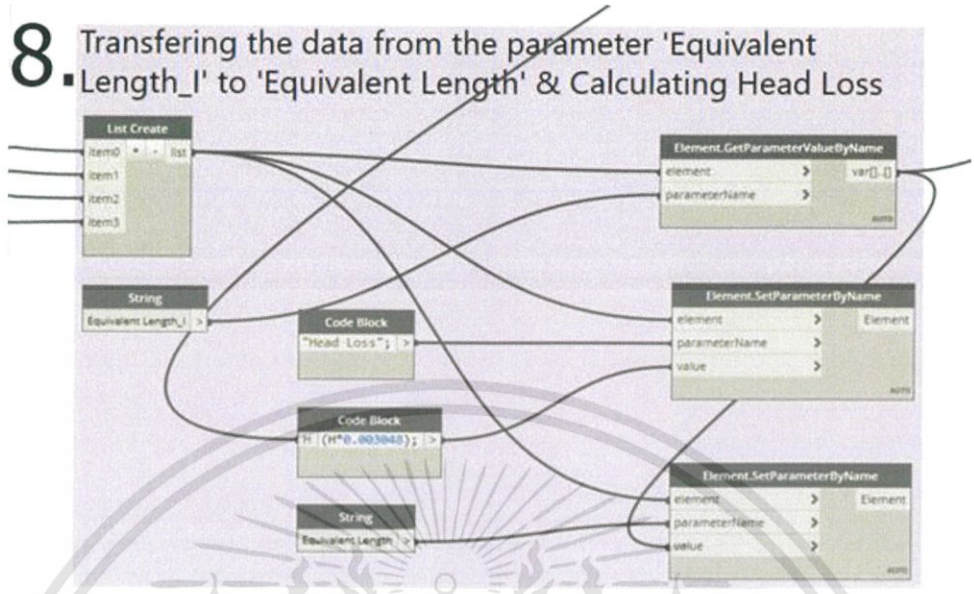
ขั้นตอนที่ 7 ย้ายค่าข้อมูลจากพารามิเตอร์ ‘Nominal Diameter1’ ของ Pipe Fiitting แต่ละชนิด (Family) ไปยังพารามิเตอร์ ‘Nominal Diameter’เพื่อทำให้ค่าข้อมูลเหล่านี้ปรากฏใน Schedule



รูปที่ 3.52 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) ในขั้นตอนที่ 7

ขั้นตอนที่ 8 คัดลอกข้อมูลจากพารามิเตอร์ ‘Equivalent Length_I’ ไปยังพารามิเตอร์ ‘Equivalent Length’ เพื่อทำให้ค่าข้อมูลนี้ใส่ลงใน Schedule ได้ จากนั้นจึงนำค่าข้อมูลของพารามิเตอร์ ‘Equivalent Length’ มาคำนวณเพื่อหาค่า Head Loss

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.53 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) ในขั้นตอนที่ 8

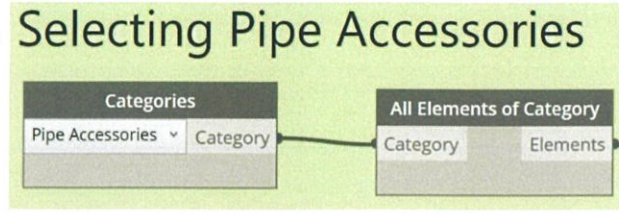
ค. วิธีการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories)

วิธีการคำนวณใน Pipe Accessories มีวิธีการคำนวณคล้ายคลึงกับการคำนวณในอุปกรณ์ข้อต่อ (Pipe Fitting) แต่จำเป็นต้องเปลี่ยนชื่อพารามิเตอร์ทั้งหมดเพื่อป้องกันการซ้อนทับกันของข้อมูล ดังนั้นผู้ศึกษาจึงต้องสร้างพารามิเตอร์ใหม่ขึ้นมาใน Schedule โดยใช้ชื่อว่า Pipe Accessories Schedule เพื่อรอรับค่าที่ได้จากการคำนวณในซอฟท์แวร์ Dynamo ได้แก่พารามิเตอร์ ‘Flow RateIII’, ‘Roughness CoefficientIII’, ‘VelocityIII’, ‘Friction HeadIII’, ‘Head LossIII’, ‘Equivalent LengthIII’ และ ‘Friction HeadIII’ ซึ่งพารามิเตอร์ทั้งหมดที่กล่าวมานี้เป็นพารามิเตอร์ประเภท Number แบบ Instance อีกทั้งผู้ศึกษาต้องสร้างพารามิเตอร์เช่นเดียวกับวิธีที่ 1 นั่นคือ ‘Comment4’ เพื่อแสดงประเภทของระบบท่อและพารามิเตอร์ ‘No.I’ เพื่อแสดงลำดับท่อ

สามารถอธิบายแผนภาพ Dynamo ได้ดังนี้

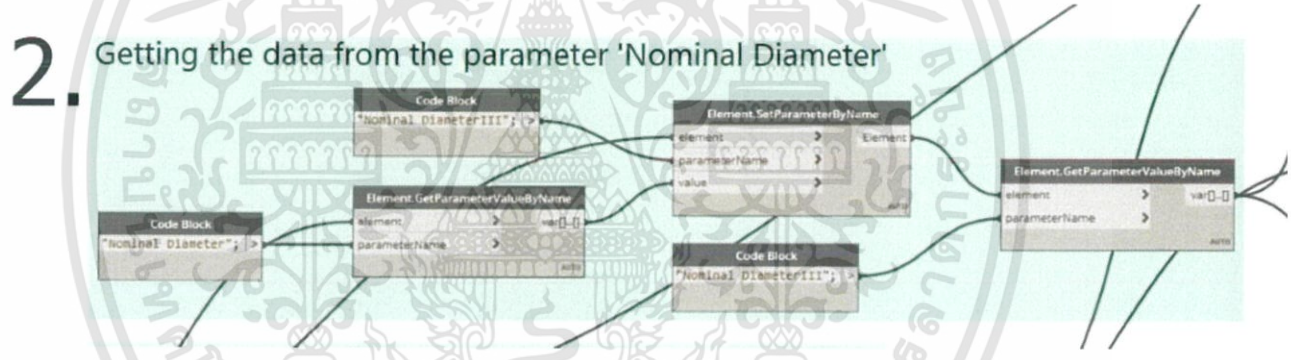
ขั้นตอนที่ 1 เลือกอุปกรณ์ข้อต่อ (Pipe Fitting) ทั้งหมดที่อยู่ในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.54 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories) ในขั้นตอนที่ 1

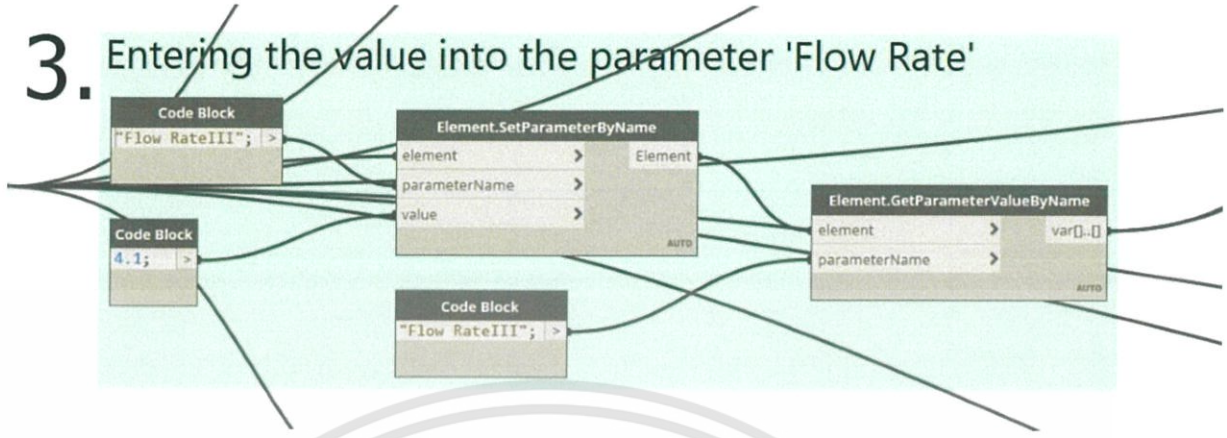
ขั้นตอนที่ 2 ดึงค่าข้อมูลจากพารามิเตอร์ 'Nominal Diameter' ใส่ลงในพารามิเตอร์ 'Nominal ParameterIII' เพื่อที่จะทำให้พารามิเตอร์นี้ปรากฏอยู่ใน Schedule ซึ่งค่าที่ได้คือขนาดของเส้นผ่านศูนย์กลางของวาล์ว



รูปที่ 3.55 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories) ในขั้นตอนที่ 2

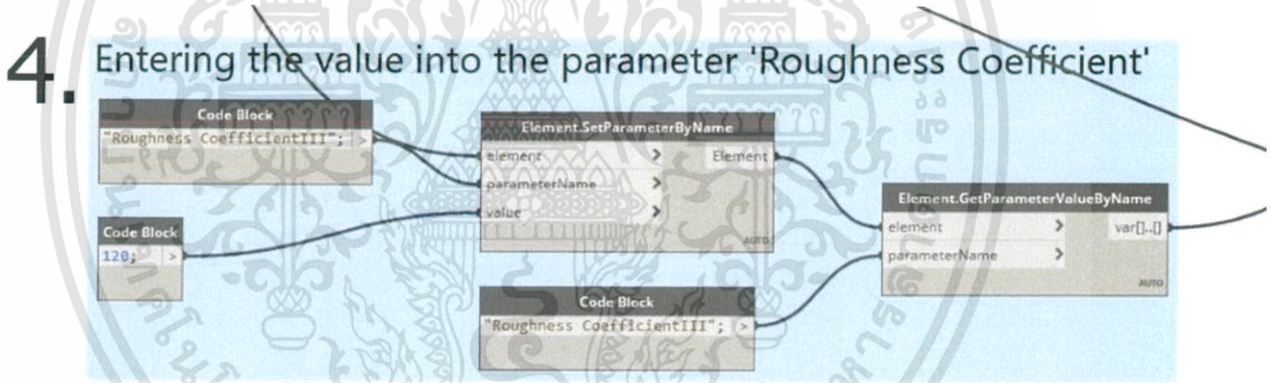
ขั้นตอนที่ 3 ใส่ข้อมูลลงไปในพารามิเตอร์ 'Flow Rate' ในกรณีนี้สมมติให้อัตราการไหลมีค่าเท่ากันในทุกท่อ นั่นคือ 4.1 ลิตรต่อวินาที (L/s)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.56 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories) ในขั้นตอนที่ 3

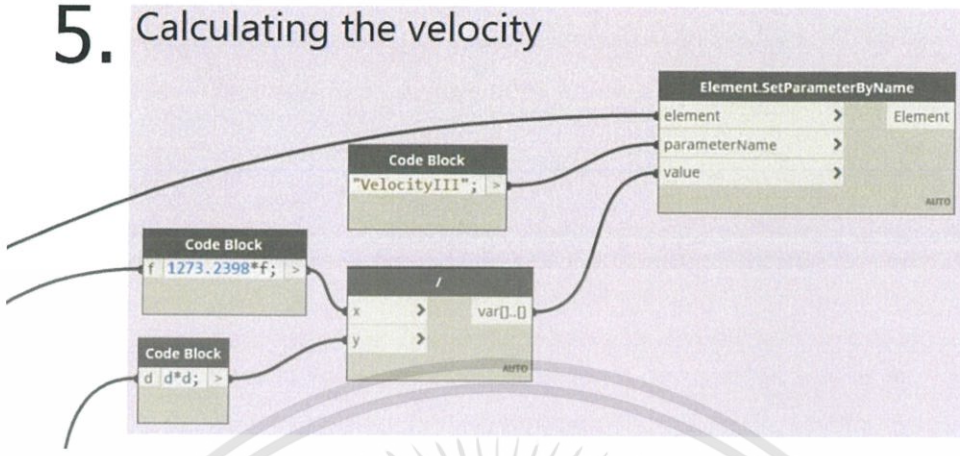
ขั้นตอนที่ 4 ใส่ข้อมูลลงไปในพารามิเตอร์ ‘Roughness Coefficient’ ซึ่งในกรณีนี้จะมีค่าเท่ากันหมด นั่นคือ 120



รูปที่ 3.57 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories) ในขั้นตอนที่ 4

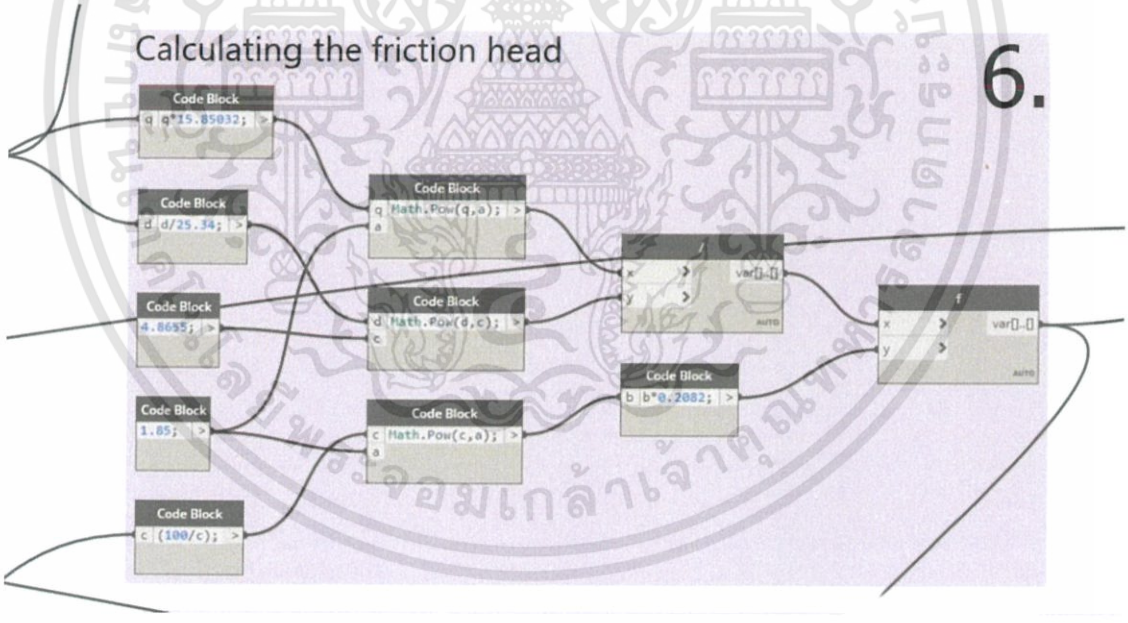
ขั้นตอนที่ 5 คำนวณหาค่าความเร็วจากสมการ Q = A x V จากนั้นจึงนำผลลัพธ์ที่ได้ใส่ลงในพารามิเตอร์ ‘VelocityIII’

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.58 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories) ในขั้นตอนที่ 5

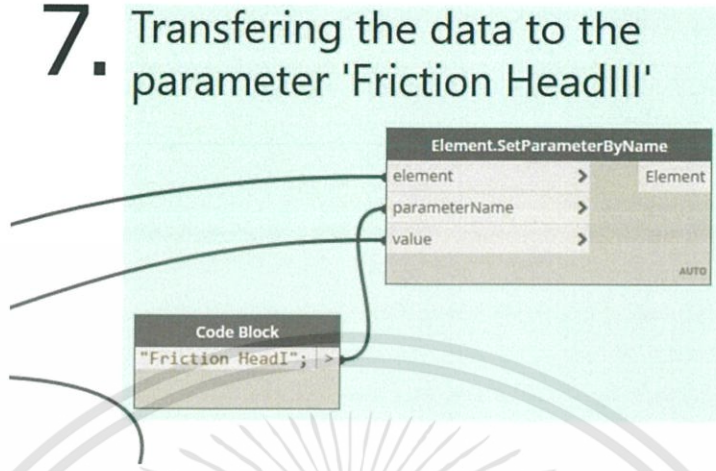
ขั้นตอนที่ 6 นำข้อมูลที่ได้จากในขั้นตอนที่ 2,3 และ 4 มาคำนวณหาค่า Friction Head จากสมการที่ 3.1



รูปที่ 3.59 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories) ในขั้นตอนที่ 6

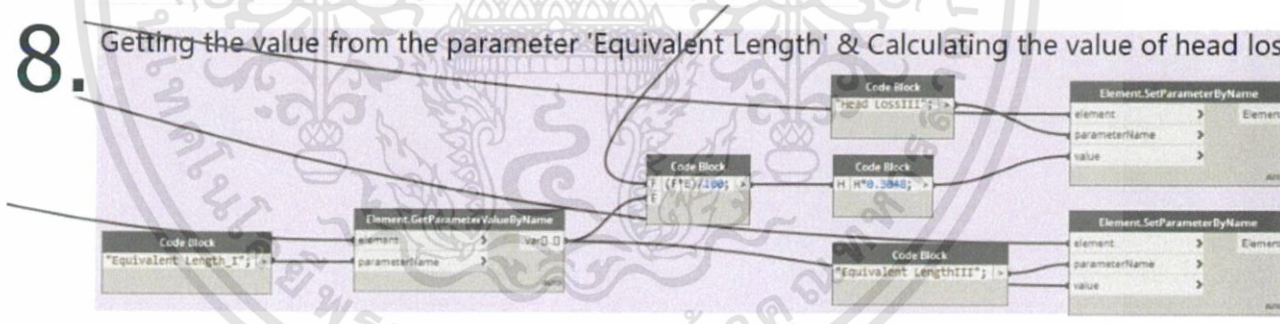
ขั้นตอนที่ 7 นำค่าข้อมูลที่ได้ใส่ลงในพารามิเตอร์ ‘Friction HeadIII’

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.60 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories) ในขั้นตอนที่ 7

ขั้นตอนที่ 8 คำนวณหาค่า Head Loss โดยการนำค่า Friction Head ที่ได้จากขั้นตอนที่ 6 คูณกับค่าของพารามิเตอร์ 'Equivalent Length_I' ผลลัพธ์ที่ได้มีหน่วยเป็นเมตร (m) และนำข้อมูลใส่ลงในพารามิเตอร์ 'Head LossIII'



รูปที่ 3.61 แสดงภาพขยายแผนภาพ Dynamo ของการคำนวณในอุปกรณ์จำพวกวาล์ว (Pipe Accessories) ในขั้นตอนที่ 8

เมื่อได้ข้อมูล Head Loss ของ Pipe, อุปกรณ์ข้อต่อ (Pipe Fitting) และ Pipe Accessories แล้ว จากนั้นจึงนำข้อมูลทั้งหมดเข้าไปใน Excel โดยใช้ไฟล์สกุล .txt ดังแสดงในรูปที่ 3.62 3.63 และ 3.64 ตามลำดับเพื่อนำมาเรียงลำดับและจัดรูปแบบของข้อมูลอีกครั้งหนึ่งเช่นเดียวกับวิธีการทำในวิธีที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pipe Schedule											
System Name	Diameter	Comment3	No.	Length1	Roughness Coefficient	Flow Rate	Velocity	Friction Head	Static Loss	Dynamic Loss	Total Head
SW 21	100	CSP2	1	2025.26181	120	4.1	0.522028	0.430528	26.957815	0.002658	26.960473
SW 44	100	CSP2	2	150	120	4.1	0.522028	0.430528	24.749085	0.000197	24.749282
SW 27	100	CSP2	3	187.614063	120	4.1	0.522028	0.430528	24.422481	0.000246	24.422727
SW 27	100	CSP2	4	790	120	4.1	0.522028	0.430528	24.183717	0.001037	24.184753
SW 27	100	CSP2	5	168	120	4.1	0.522028	0.430528	24.183717	0.00022	24.183937
SW 27	100	CSP2	6	2434.175697	120	4.1	0.522028	0.430528	24.005867	0.003194	24.009061
SW 27	100	CSP2	7	3280.17225	120	4.1	0.522028	0.430528	21.505847	0.004304	21.510151
SW 27	100	CSP2	8	13223.29078	120	4.1	0.522028	0.430528	21.505847	0.017352	21.5232
SW 27	100	CSP2	9	33.2121	120	4.1	0.522028	0.430528	21.505847	0.000044	21.505891
SW 27	100	CSP2	10	4974.073882	120	4.1	0.522028	0.430528	21.338303	0.006527	21.344831
SW 27	100	CSP2	11	2533.648562	120	4.1	0.522028	0.430528	16.298386	0.003325	16.30171
SW 27	100	CSP2	12	9358.647809	120	4.1	0.522028	0.430528	16.298336	0.012281	16.310617
SW 27	100	CSP2	13	8846.611939	120	4.1	0.522028	0.430528	16.298336	0.011609	16.309945
SW 27	100	CSP2	14	8894.576391	120	4.1	0.522028	0.430528	16.298336	0.011672	16.310008
SW 27	100	CSP2	15	22325.94436	120	4.1	0.522028	0.430528	16.298336	0.029297	16.327633
SW 27	100	CSP2	16	1006.841754	120	4.1	0.522028	0.430528	16.298336	0.001321	16.299657
SW 27	100	CSP2	17	12141.50605	120	4.1	0.522028	0.430528	16.130792	0.015933	16.146724
SW 27	100	CSP2	18	70	120	4.1	0.522028	0.430528	3.915136	0.000092	3.915227
SW 27	100	CSP2	19	65.441559	120	4.1	0.522028	0.430528	3.915136	0.000086	3.915221
SW 27	100	CSP2	20	431.037311	120	4.1	0.522028	0.430528	3.915136	0.000566	3.915701

รูปที่ 3.62 แสดงตารางข้อมูลต่าง ๆ ของ Pipe Schedule ในโปรแกรม Microsoft Excel

Pipe Fitting Schedule											
Type	System Type	System Name	Size	Comment4	No.I	Equivalent Length	Nominal Diameter	Friction HeadI	Flow RateI	Roughness CoefficientI	Head Loss
M_REDUCING TEE_WELDED	P_Condensate(pumped)	SW 27	100-100-100	CSP2	3.5	7.6	100	0.421635	4.1	120	0.009767
M_REDUCING TEE_WELDED	P_Condensate(pumped)	SW 27	100-100-100	CSP2	4.5	7.6	100	0.421635	4.1	120	0.009767
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	5.5	10.1	100	0.421635	4.1	120	0.01298
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	6.5	10.1	100	0.421635	4.1	120	0.01298
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	7.5	10.1	100	0.421635	4.1	120	0.01298
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	8.5	10.1	100	0.421635	4.1	120	0.01298
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	9.5	10.1	100	0.421635	4.1	120	0.01298
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	10.5	10.1	100	0.421635	4.1	120	0.01298
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	11.5	10.1	100	0.421635	4.1	120	0.01298
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	12.5	10.1	100	0.421635	4.1	120	0.01298
Standard	P_Condensate(pumped)	SW 27	100-100	CSP2	13.5	10.1	100	0.421635	4.1	120	0.01298

รูปที่ 3.63 แสดงตารางข้อมูลต่าง ๆ ของ Pipe Fitting Schedule ในโปรแกรม Microsoft Excel

Pipe Accessory Schedule									
Type	System Name	System Type	No.II	Nominal DiameterIII	Friction HeadIII	Flow RateIII	Roughness CoefficientIII	Equivalent LengthIII	Head LossIII	VelocityIII
100 CHECK VALV	SW 44,SW 21	ondensate(pum	1.5	100	0.421635	4.1	120	33.6	0.043181	0.522028
100 GATE VALV	SW 27,SW 44	ondensate(pum	2.5	100	0.421635	4.1	120	2.7	0.00347	0.522028

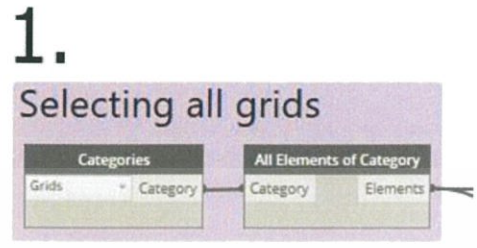
รูปที่ 3.64 แสดงตารางข้อมูลต่าง ๆ ของ Pipe Accessories Schedule ในโปรแกรม Microsoft Excel

4. การสร้าง Grid Line ในรูปแบบ 3 มิติ

เนื่องจากในโปรแกรม Autodesk Revit ยังไม่มีเครื่องมือใดที่สามารถทำให้ Grid Lines แสดงออกมาได้เมื่อเปิดแบบจำลอง 3 มิติ จึงต้องมีการนำซอฟท์แวร์ Dynamo เวอร์ชั่น 2.0.1 เข้ามาเพื่อเขียนโค้ดเพิ่มเติม สามารถอธิบายเป็นขั้นตอนอย่างละเอียดได้ดังนี้

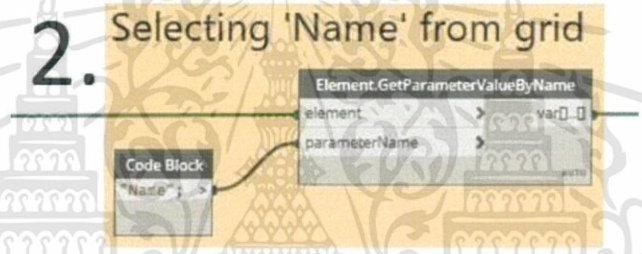
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 1 เลือก Grid Line ที่ต้องการในชิ้นงานโดยการใช้โหนด Category และ All Elements of Category



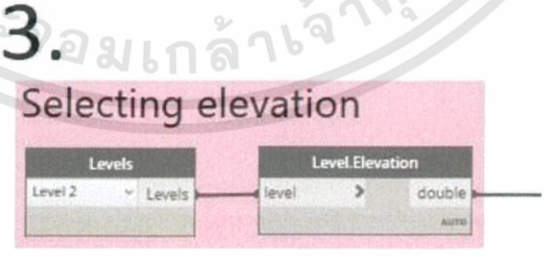
รูปที่ 3.65 แสดงภาพขยายแผนภาพ Dynamo ในการสร้าง Grid Line ในมุมมองสามมิติในขั้นตอนที่ 1

ขั้นตอนที่ 2 ใช้โหนด Element.GetParameterValueByName เพื่อเลือกพารามิเตอร์ ‘Name’



รูปที่ 3.66 แสดงภาพขยายแผนภาพ Dynamo ในการสร้าง Grid Line ในมุมมองสามมิติในขั้นตอนที่ 2

ขั้นตอนที่ 3 เลือกระดับชั้นของชิ้นงาน (ในกรณีที่ชิ้นงานมีระดับชั้นมากกว่า 1 ระดับชั้น) และเชื่อมต่อเข้ากับโหนด Level.Elevation ซึ่งโหนดนี้ทำหน้าที่คำนวณหาระยะห่างของ Grids เมื่อเทียบกับระดับชั้นพื้น อาจแสดงผลเป็นตัวเลขจำนวนเต็มหรือตัวเลขที่มีทศนิยม (Double)



รูปที่ 3.67 แสดงภาพขยายแผนภาพ Dynamo ในการสร้าง Grid Line ในมุมมองสามมิติในขั้นตอนที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 การแสดง Grid Line ในมุมมอง 3 มิติ

4.1) เปลี่ยน Elements ทั้งหมดเป็นเส้นโค้งโดยใช้โหนด Grid.curve

4.2) นำค่าที่ได้จากขั้นตอนที่ 3 มาเชื่อมต่อกับ Code Block ซึ่งภายในจะเขียนโค้ดเพื่อสร้างระนาบ xz โดยกำหนดให้พารามิเตอร์ h แสดงถึงค่าในแนวแกน z และพารามิเตอร์ a แสดงถึงคู่อันดับ (x,y,z) โดยใช้โค้ดPoint.ByCoordinates(0,0,h) ทำให้เกิดจุดบนแกน z และเขียนโค้ด Plance.ByOriginNormal(a.Vector.ZAxis()) เป็นคำสั่งในการเขียนระนาบ xz โดยใช้คู่อันดับ (x,y,z) ของพารามิเตอร์ a เป็นจุดศูนย์กลางของระนาย xz ที่ต้องการ จากนั้นจึงเชื่อมโค้ดทั้งหมดนี้เข้ากับ Port baseGeometry ในโหนด Curve.Project ในขณะเดียวกันจะต้องเขียนโค้ด Vector.ZAxis() เชื่อมต่อ Port projectDirection ในโหนด Curve.Project เพื่อรับค่าของเวกเตอร์

4.3) จากโหนด Curve.Project ทำการเชื่อม Port Geometry เข้ากับ Port List ในโหนด List.Flatten เพื่อรวบรวมข้อมูลที่ถูกแยกออกจากกันเป็นลำดับให้กลายเป็นข้อมูลชุดเดียวกันและเชื่อมเข้ากับโหนด ModelCurve.Bycurve เพื่อสร้าง Element ตามเส้นโค้งที่กำหนด

4.4) เขียนโค้ดใน Code Block โดยกำหนดให้ a=crv.ExtendStart(offset) ในที่นี้ crv คือพารามิเตอร์ใหม่ที่สร้างขึ้นมาเพื่อสร้าง Element (ในที่นี้หมายถึงจุดเล็กๆที่ประกอบกันเป็นเส้นตรงขแง Grid Line)จาก ExtendStart ไปจนถึง ExtendEnd ตามค่า offset ที่กำหนด

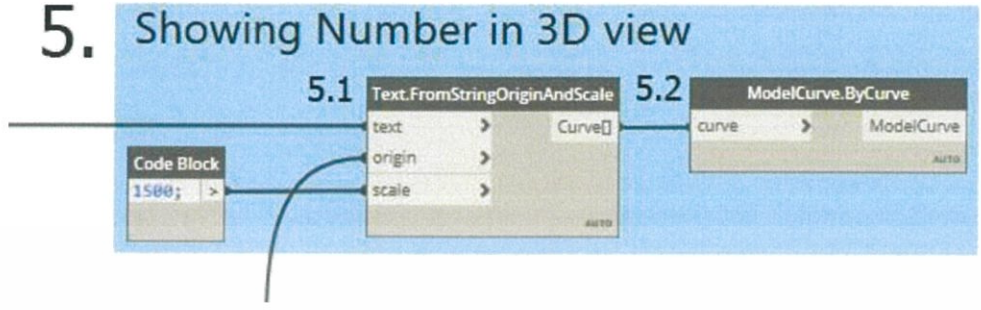
4.5) จากขั้นตอนที่ 4.4 ทำการเขียนโค้ดใน Code Block เพื่อกำหนดจุดเริ่มต้นและจุดสิ้นสุดของระยะเส้นโค้งของพารามิเตอร์ ‘crv’

ขั้นตอนที่ 5 การแสดงตัวเลขลำดับของ Grid Line ในมุมมอง 3 มิติ

5.1) นำค่าที่ได้จากขั้นตอนที่ 2 เชื่อมเข้ากับ Port Text ในโหนด Text.FromStringOriginAndScale (ทำหน้าที่แสดงตัวเลขหรือตัวอักษรในมุมมองสามมิติ) เพื่อให้ค่าที่รับเข้ามาเป็นตัวอักษร ในส่วนของ Port Origin จะรับค่าพารามิเตอร์ ‘crv’ ที่ได้จากขั้นตอนที่ 8 เพื่อรับค่าเข้ามาเป็นแบบจุด (Point) และสร้าง Code Block ใหม่มาเชื่อมต่อกับ Port Scale เพื่อบอกขนาดของตัวเลขที่แสดงลำดับของ Grid ในมุมมองสามมิติ

5.2) เชื่อมโหนด Text.FromStringOriginAndScale เข้ากับ ModelCurve.ByCurve เพื่อสร้าง Element ขึ้นมาตามเส้นโค้งที่ประกอบกันเป็นตัวเลขหรือตัวอักษรตามที่ต้องการ ซึ่งในกรณีนี้เป็นการสร้างเส้นโค้งที่ประกอบกันเป็นตัวเลขเพื่อบอกตัวเลขลำดับของ Grid Line

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.68 แสดงภาพขยายแผนภาพ Dynamo ในการสร้าง Grid Line ในมุมมองสามมิติในขั้นตอนที่ 5

5. การระบุตำแหน่งของวัตถุที่ต้องการในแบบจำลอง 3 มิติด้วย Grid Line

ในการสร้างแบบจำลอง 3 มิติของชิ้นงานหนึ่ง จะประกอบไปด้วยระบบที่หลากหลาย เช่น ระบบป้องกันอัคคีภัย (Fire Protection System), ระบบสุขาภิบาลและการประปา (Sanitary and Plumbing System), ระบบปรับความดันบรรยากาศ (Environment Control System) เป็นต้น ซึ่งเมื่อนำระบบเหล่านี้มารวมกัน อุปกรณ์ในแต่ละระบบอาจเกิดการชนกัน (Clash) แต่ในโปรแกรม Autodesk Revit ไม่สามารถตรวจสอบและระบุตำแหน่งของวัตถุที่เกิดการชนกันได้ภายในเวลาอันรวดเร็ว ดังนั้นผู้ศึกษาจึงใช้ซอฟท์แวร์ Dynamo เวอร์ชั่น 1.3.1 เพื่อระบุ Grid Lines ที่ใกล้ที่สุดของวัตถุที่ต้องการในชิ้นงาน

สามารถอธิบายเป็นขั้นตอนอย่างละเอียดได้ดังนี้

ขั้นตอนที่ 1 ระบุพิกัดของจุดที่ Grid ตัดกันและนำตัวเลขในพารามิเตอร์ 'Name' มาใช้

1.1) เลือก Grid Line ทั้งหมดโดยการใช้โหนด Categoryและ All Elements of Category

1. 2) เชื่อมโหนด All Elements of Categoryเข้ากับ Port list ของโหนด List.GroupBykey เพื่อเลือก element และเชื่อมเข้ากับโหนด grid.curve ตามลำดับ เพื่อเปลี่ยน grid จากเส้นโค้งให้กลายเป็นค่าพิกัดจุด (x,y,z) โดยค่าที่ระบุจะแสดงเพียงแค่ค่าจุดเริ่มต้นและจุดสิ้นสุด

1.3) จากโหนด Grid.Curve ทำการเชื่อมเข้ากับโหนด Line.Direction เพื่อเปลี่ยนจุดพิกัด (x,y,z) ให้เป็นเวกเตอร์ สังเกตได้ว่าค่าที่ได้เป็นเวกเตอร์ที่มีทิศทางไปในแนวแกน X และ แกน Y จากนั้นจึงเชื่อมต่อโหนดไปยัง List.Firstitem เพื่อเก็บค่าแรกของโหนด Line.Direction

1.4) นำค่าจากโหนด Line.Direction เข้าไปยัง Port vector ของโหนด Vector.IsParallel และนำค่าจากโหนด List.FirstItem เข้าไปยัง Port other ของโหนด Vector.IsParallel เพื่อทดสอบว่าเวกเตอร์ใดที่ขนานกับเวกเตอร์ในโหนด List.FirstItem ส่งผลออกมาเป็น Boolean ที่ให้ค่า True หรือ False

1.5) นำค่าที่ได้จากโหนด Vector.IsParallel เก็บไว้ในโหนด List.GroupByKey ผ่านทาง Port keys ซึ่งโหนดนี้มีหน้าที่จัด list ที่มีค่าเดียวกับ key ไว้ด้วยกัน

1.6) ในส่วนของโหนด List.GroupByKey โหนดแรก ที่ได้รับค่าจากขั้นตอนที่ 1.2 ซึ่งโหนดนี้จะเก็บค่าของทุก Grid ไว้ โดยค่าที่ปรากฏใน List นี้เป็นตัวเลข ID ของ Grid Line แต่ละเส้นจากนั้นจึงเชื่อมต่อPort groupsของ List.GroupByKeyเข้ากับElement.GetParameterValueByName

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อดึงค่าตัวอักษรและตัวเลขลำดับของ Grid ออกมาโดยในขั้นตอนนี้ข้อมูลจะถูกแยกออกเป็น 2 ชุด ซึ่งในชุดที่หนึ่งคือตัวเลข เช่น 1 2 3 เป็นต้น และชุดที่สองคือตัวอักษรภาษาอังกฤษ เช่น A B C เป็นต้น ต่อมาจึงนำโหนด Element.GetParameterValueByName เชื่อมเข้ากับโหนด List.FirstItem สำหรับเก็บข้อมูลชุดที่หนึ่ง (ข้อมูลตัวอักษรภาษาอังกฤษ) และเชื่อมเข้ากับโหนด List.LastItem สำหรับเก็บข้อมูลชุดที่สอง (ข้อมูลตัวเลข)

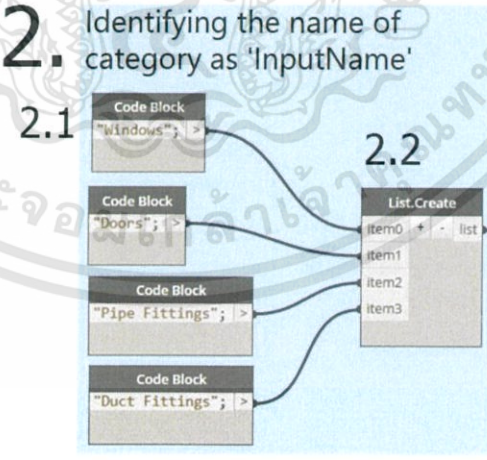
1.7) เชื่อมทั้งโหนด List.FirstItem และ List.LastItem เข้ากับ Code Block ภายในจะเขียนโค้ด *Transpose(a+"/"+b<1>)* ในที่นี้การ Transpose คือการสลับค่าในแต่ละแถวและคอลัมน์ ดังนั้นค่าที่ได้จะออกมาในรูปแบบของ ตัวอักษรภาษาอังกฤษ/ตัวเลข ตัวอย่างเช่น A/1 เป็นต้น แต่ข้อมูลที่ถูกเก็บไว้จะแบ่งออกเป็น 5 ชุด จากนั้นจึงเชื่อมต่อ Code Block เข้ากับ List.Flatten เพื่อจัดข้อมูลเหล่านั้นให้อยู่ใน List เดียวกัน

1.8) ในส่วนของโหนด List.GroupByKey โหนดที่สองนำไปเชื่อมกับ List.FirstItem และ List.LastItem เพื่อแยกข้อมูลออกจากกัน จากนั้นจึงเชื่อม List.FirstItem และ List.LastItem เข้ากับ Port other และ Geometry ของโหนด Geometry.Intersect ในขั้นตอนนี้เองที่ทำให้ข้อมูลจากโหนด List.FirstItem และ List.LastItem จับคู่กัน จากนั้นจึงนำไปเชื่อมกับโหนด Flatten เพื่อรวบรวมข้อมูลทั้งหมดให้อยู่ใน List เดียวกัน

ขั้นตอนที่ 2 ระบุ Category ที่ต้องการทราบว่าอยู่ ณ Grid Line ใด

2.1) เขียนโค้ดเพื่อระบุชื่อ Category ใน Code Block ดังภาพที่ 10

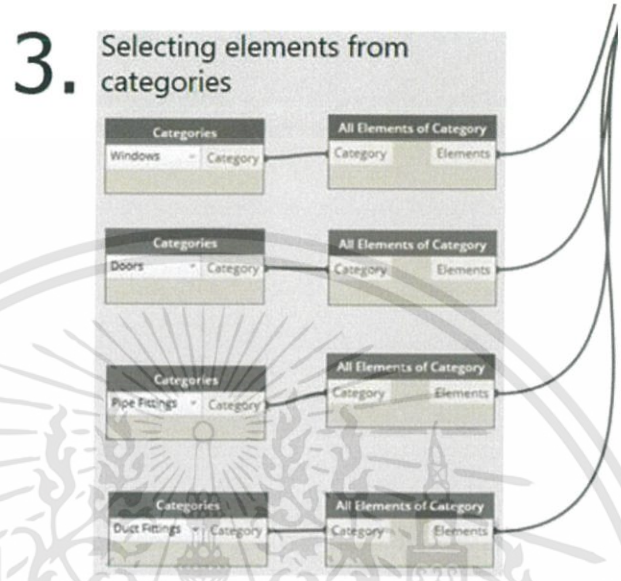
2.2) เชื่อมต่อ Code Block ที่ได้จากขั้นตอนที่ 2.1 เข้ากับโหนด List.create เพื่อจัดเก็บข้อมูลให้อยู่ในรูปของรายการของชื่อ Category ที่ได้จากขั้นตอนที่ 2.1



รูปที่ 3.69 แสดงภาพขยายแผนภาพ Dynamo ในการระบุ Grid Line ที่ใกล้ที่สุดของวัตถุที่ต้องการในขั้นตอนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 3 เลือก element ใน category จากโหนด Categories และ
โหนด All Elements of Category



รูปที่ 3.70 แสดงภาพขยายแผนภาพ Dynamo ในการระบุ Grid Line ที่ใกล้ที่สุดของวัตถุที่ต้องการในขั้นตอนที่ 3

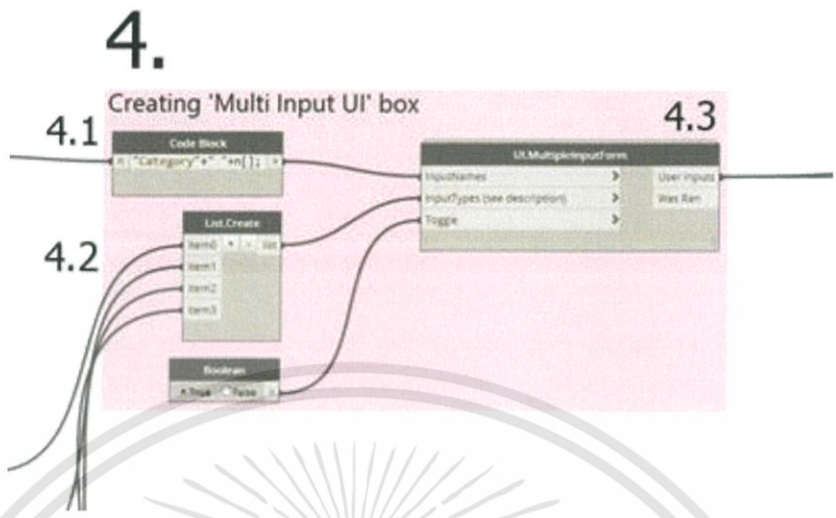
ขั้นตอนที่ 4 ส่งข้อมูลทั้งหมดเข้าไปยังโหนด UI.MultipleInputForm

4.1) นำโหนด List.Create ที่ได้จากขั้นตอนที่ 2.2 เชื่อมเข้ากับ Code Block ภายในเขียนโค้ด *"category"+" "+n[];* เพื่อกำหนดให้ข้อมูลที่อยู่ใน list ทั้งหมดมีค่าเป็นตัวแปร ดังนั้นข้อมูลที่ได้จะอยู่ในรูปแบบของ Category+ ชื่อของ Category เช่น Category Door, Category Pipe Fitting เป็นต้น ดังนั้นในขั้นตอนนี้จะเป็นการกำหนดให้แสดงรายชื่อในหน้าต่าง Multi Input UI

4.2) นำโหนด All Elements of Category จากขั้นตอนที่ 3 เชื่อมต่อกับโหนด List.Create เพื่อจัดเก็บข้อมูลให้อยู่รูปของรายการ

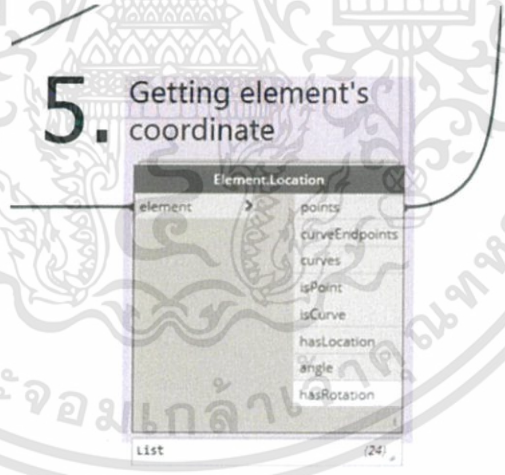
4.3) นำ Code Block ที่ได้จากขั้นตอนที่ 4.1 และ List.Create ที่ได้จากขั้นตอนที่ 4.2 เชื่อมต่อเข้ากับ Port InputNames และ Port InputTypes ของโหนด UI.MultipleInputForm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.71 แสดงภาพขยายแผนภาพ Dynamo ในการระบุ Grid Line ที่ใกล้ที่สุดของวัตถุที่ต้องการในขั้นตอนที่ 4

ขั้นตอนที่ 5 เชื่อมต่อ Port User inputs ของโหนด UI.MultipleInputFrom เข้ากับโหนด Element.Location เพื่อรับค่าของจุดในระนาบ xyz หรือตำแหน่งของวัตถุนั้น



รูปที่ 3.72 แสดงภาพขยายแผนภาพ Dynamo ในการระบุ Grid Line ที่ใกล้ที่สุดของวัตถุที่ต้องการในขั้นตอนที่ 5

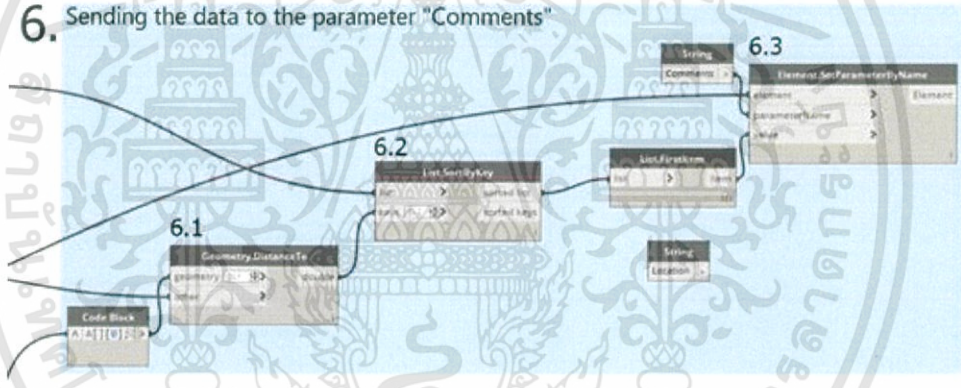
ขั้นตอนที่ 6 นำข้อมูลที่ได้มาทั้งหมดใส่ไว้ในพารามิเตอร์ ‘Comment’

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1) เขียนโค้ด *A[][0]* ใน Code Block เพื่อกำหนดให้ตัวแปร A เก็บค่าพิกัดจุดไว้ใน Array จากนั้นจึงเชื่อมต่อเข้ากับโหนด Geometry.DistanceTo เพื่อได้ค่าระยะห่างระหว่างวัตถุกับพิกัดจุดที่ Grid Line ตัดกัน

6.2) เชื่อมต่อโหนด Flatten ที่ได้จากในขั้นตอนที่ 1.8 เข้ากับ Port list และเชื่อมต่อโหนด Geometry.DistanceTo เข้ากับ Port key ของโหนด List.sortByKey ตามลำดับ โดย Port key จะทำหน้าที่เป็นตัวเปรียบเทียบข้อมูล กล่าวคือหากมีข้อมูลที่มีค่าตรงกับข้อมูลที่อยู่ใน key แล้วโหนด List.SortByKey จะเก็บข้อมูลเหล่านั้นลงใน list จากนั้นจึงเชื่อม Port sorted list ของโหนดดังกล่าวเข้ากับโหนด List.FirstItem เพื่อดึงค่าข้อมูลตัวแรกของ List

6.3) จากโหนด Element.SetParameterByName เชื่อม Port element เข้ากับ Port User inputs ของโหนด UI.MultipleInputForm จากนั้นจึงเชื่อมโหนด List.Flatten เข้ากับพอร์ต Value และเชื่อมโหนด String ซึ่งภายนเขียนโค้ด Comments เข้ากับ Port parameterName เพื่อระบุชื่อพารามิเตอร์ ในที่นี้คือพารามิเตอร์ ‘comments’ แสดงข้อมูลสำหรับระบุตำแหน่ง Grid Line ที่ใกล้ที่สุด



ภาพที่ 3.73 แสดงภาพขยายแผนภาพ Dynamo ในการระบุ Grid Line ที่ใกล้ที่สุดของวัตถุที่ต้องการ ในขั้นตอนที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการวิจัย

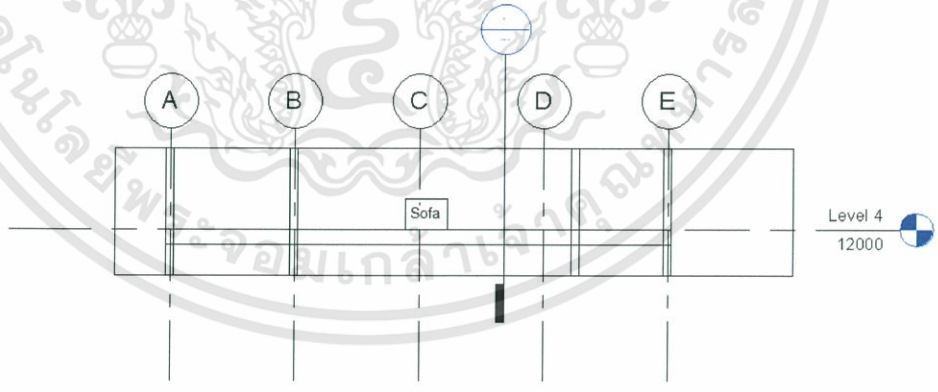
โครงงานสหกิจศึกษาเรื่องการปรับปรุงและพัฒนาการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคารภายในสถานีรถไฟฟ้าใต้ดินสายสีส้ม มีวัตถุประสงค์เพื่อศึกษากระบวนการแบบจำลองสารสนเทศอาคาร (Building Information Modeling) โดยใช้โปรแกรม Autodesk Revit ควบคู่กับซอฟท์แวร์ Dynamo, โปรแกรม Microsoft Excel และภาษา Python ช่วยลดระยะเวลาและหลีกเลี่ยงความซ้ำซ้อนในการปฏิบัติงานซึ่งอาจก่อให้เกิดข้อมูลที่ผิดพลาด อีกทั้งยังช่วยลดค่าใช้จ่ายในส่วนของแรงงานในระหว่างการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคารภายในรถไฟฟ้าใต้ดินสายสีส้มในสัญญาที่ 1 และ 2

4.1 ผลการวิจัย

การแก้ไขปัญหาที่เกิดขึ้นในระหว่างการปฏิบัติงานด้วยกระบวนการแบบจำลองสารสนเทศอาคารภายในรถไฟฟ้าสายสีส้มในสัญญาที่ 1 และ 2 สามารถสรุปผลได้ดังต่อไปนี้

1.การแสดงมุมมองของส่วนที่เฉพาะเจาะจงของแบบจำลอง 3 มิติ โดยใช้เครื่องมือ Section Views

ในการทำ Section เพื่อแสดงมุมมองของ Element ในชั้นระดับที่ต้องการ สามารถทำได้โดยการใช้เครื่องมือ Section ในโปรแกรม Autodesk Revit สองครั้ง นั่นคือ Section View และ Detail View ตามลำดับหรือการใช้ Section View 2 ครั้ง



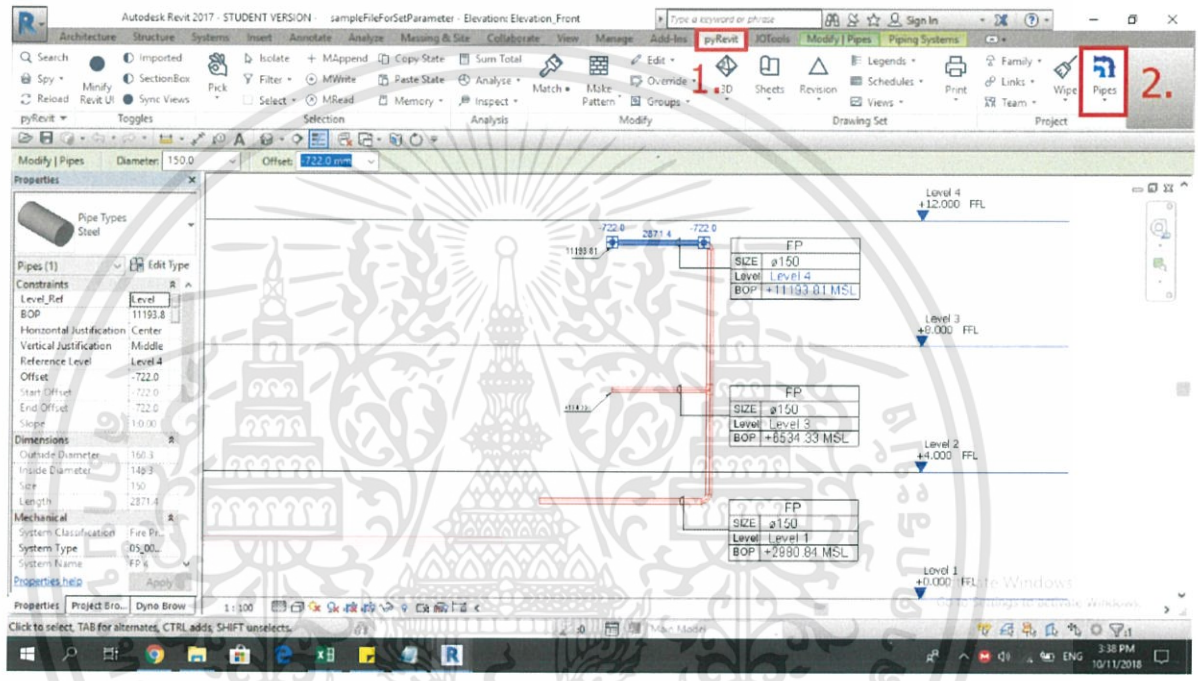
รูปที่ 4.1 แสดงมุมมองของวัตถุในชั้นระดับที่ต้องการ

เมื่อผู้ศึกษาทำการใช้ Detail View (หรืออาจใช้เพียงแค่ Section View ก็สามารถให้ผลลัพธ์เช่นเดียวกับการใช้ Detail View) จะทำให้ได้มุมมองของวัตถุในชั้นระดับ ดังรูปที่ 4.1 (ในที่นี้กำหนดให้วัตถุนั้นคือ Sofa)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

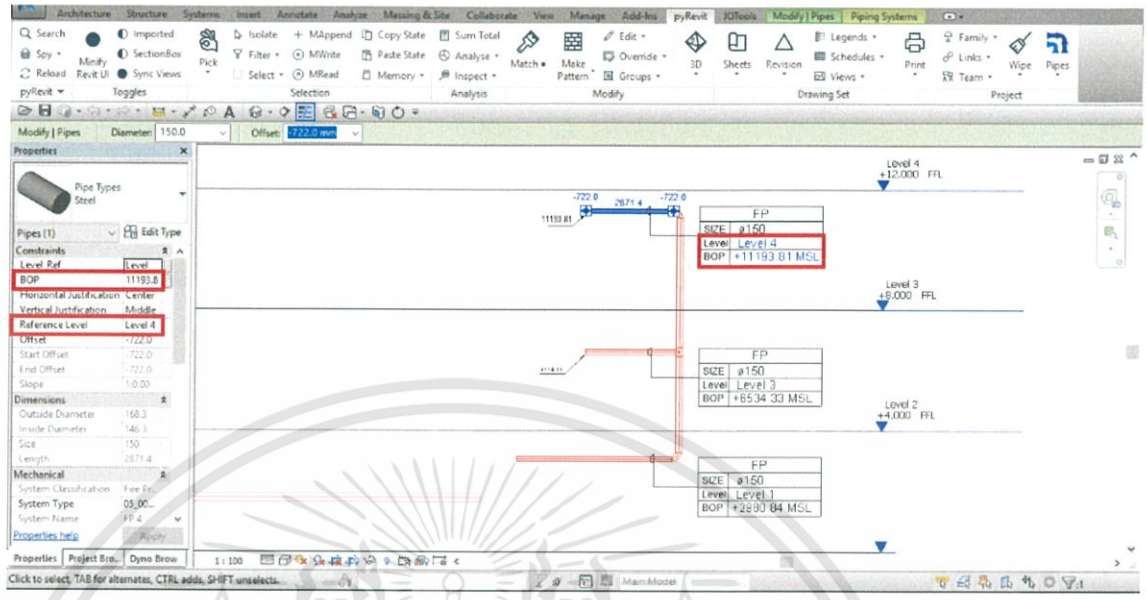
2. การเพิ่มพารามิเตอร์ Reference Level ของท่อใน Schedule

การเพิ่มพารามิเตอร์ Reference Level ของท่อทั้งท่อชนิดกลม (Pipes) และท่อชนิดเหลี่ยม (Ducts) ใน Schedule สามารถทำได้โดยการเขียนโค้ดในภาษา Python ผลลัพธ์ที่ได้จะปรากฏ Ribbon 'Project' ในแถบเครื่องมือ pyRevit ดังแสดงในรูปที่ 4.2 ซึ่งภายใน Ribbon นี้จะมีไอคอน Pipes ซึ่งเป็นเครื่องมือที่ใช้ในการวัดระดับ BOP ของท่อโดยอัตโนมัติเมื่อเคลื่อนย้ายท่อไปยังชั้นระดับต่างๆ ดังแสดงในรูปที่ 4.3



รูปที่ 4.2 แสดงแถบเครื่องมือ pyRevit (หมายเลข 1) และเครื่องมือ Pipes ใน Ribbon (หมายเลข 2)

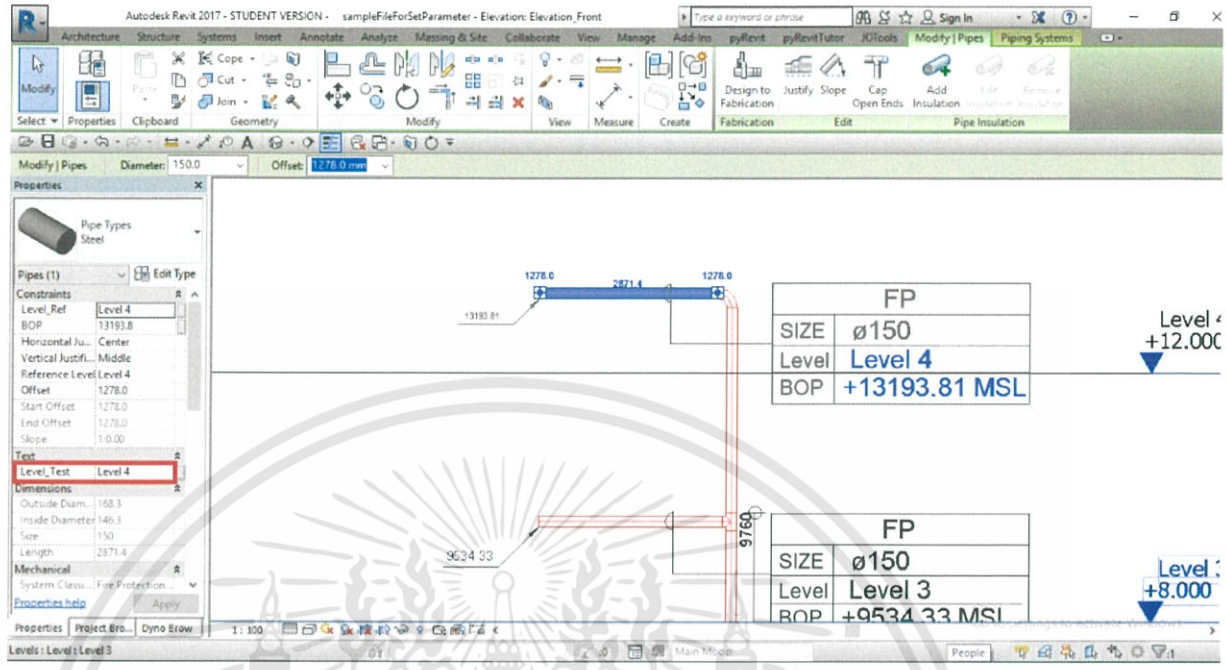
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



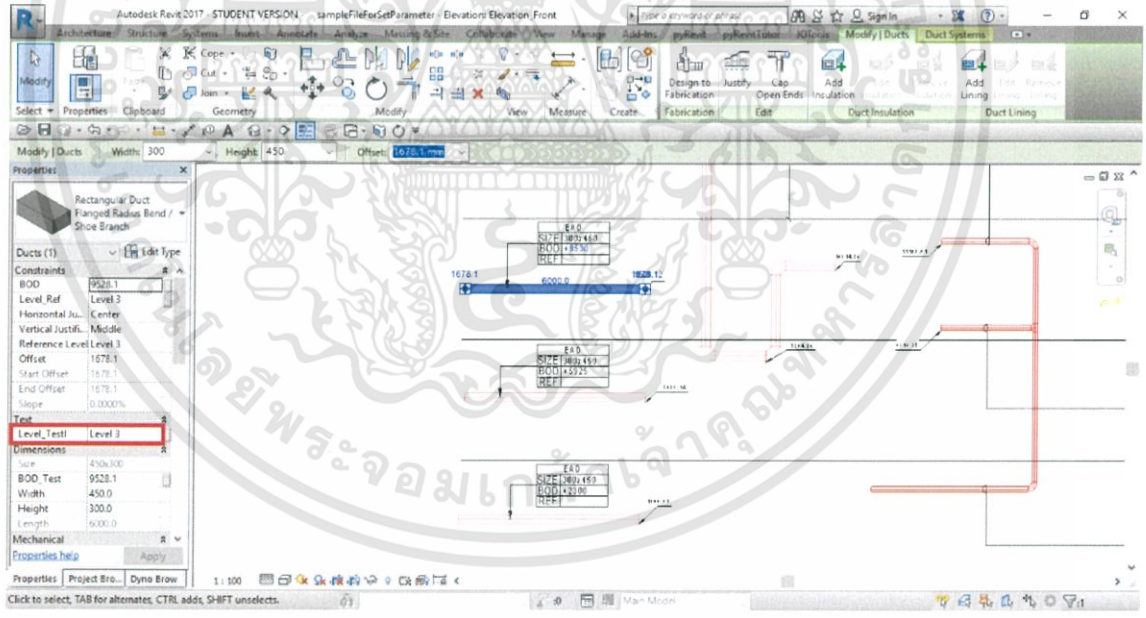
รูปที่ 4.3 แสดงชั้นระดับ (Level) และค่า BOP ของท่อที่ต้องการ (กรอบสีแดง)

นอกจากนี้ผู้ศึกษาได้เขียนโค้ดในภาษา Python เพื่อระบุชั้นระดับที่ถูกต้องของท่อชนิดกลม (Pipes) และท่อชนิดเหลี่ยม (Ducts) เนื่องจากเมื่อเคลื่อนย้ายท่อไปยังชั้นระดับอื่น พารามิเตอร์ ‘Reference Level’ และ ‘Level_Ref’ จะระบุชั้นระดับไม่ตรงกับชั้นระดับที่ท่ออยู่ โดยโค้ดนี้ได้รับการพัฒนาต่อยอดมาจากการเขียนโค้ดในวิธีการเพิ่มพารามิเตอร์ของท่อ ซึ่งผลลัพธ์ที่ได้คือเมื่อเคลื่อนย้ายท่อไปยังชั้นระดับชั้นต่างๆ แล้ว ชั้นระดับที่ถูกต้องจะปรากฏในพารามิเตอร์ ‘Level_Test’ สำหรับท่อชนิดกลม (Pipes) และพารามิเตอร์ ‘Level_Testl’ สำหรับท่อชนิดเหลี่ยม (Ducts) ดังแสดงในรูปที่ 4.4 และ 4.5 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงการระบุชั้นระดับของท่อชนิดกลม (Pipes) ในกรอบสีแดง



รูปที่ 4.5 แสดงการระบุชั้นระดับของท่อชนิดเหลี่ยม (Ducts) ในกรอบสีแดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.6 จะสังเกตได้ว่าค่าของ Reference Level ของทั้งพารามิเตอร์ ‘Level_Test’ และ ‘Level_Ref’ มีค่าที่ไม่เท่ากันเนื่องจากวิธีการดังกล่าวไม่สามารถใช้ได้กับท่อที่มีการวางในลักษณะของแนวตั้ง ดังนั้นพารามิเตอร์ ‘Level_Test’ จะระบุได้เพียงจุดเริ่มต้นของท่อเท่านั้น อีกทั้งผู้ศึกษาไม่ได้โอนถ่ายข้อมูลกลับไปในพารามิเตอร์ ‘Level_Ref’ เนื่องจากป้องกันการสับสนของโปรแกรม

<Pipe Schedule>

A	B	C	D	E
Type	BOP_Test	Level_Test	Level_Ref	Offset_Test
SAN-CW-CD-SP-D	84218.019	PLATFORM LEVEL	CONCOURSE LEVEL	3965.448
SAN-CW-CD-SP-D	84481.961	PLATFORM LEVEL	CONCOURSE LEVEL	4227.576
SAN-CW-CD-SP-D	84335.031	PLATFORM LEVEL	CONCOURSE LEVEL	4081.272
SAN-CW-CD-SP-D	84475.290	PLATFORM LEVEL	CONCOURSE LEVEL	4221.480
SAN-CW-CD-SP-D	84534.835	PLATFORM LEVEL	CONCOURSE LEVEL	4282.440
SAN-CW-CD-SP-D	84388.526	PLATFORM LEVEL	CONCOURSE LEVEL	4136.136
SAN-CW-CD-SP-D	84391.300	PLATFORM LEVEL	CONCOURSE LEVEL	4136.136
SAN-CW-CD-SP-D	84537.242	PLATFORM LEVEL	CONCOURSE LEVEL	4282.440
SAN-CW-CD-SP-D	84388.186	PLATFORM LEVEL	CONCOURSE LEVEL	4133.088
SAN-CW-CD-SP-D	84528.445	PLATFORM LEVEL	CONCOURSE LEVEL	4273.296
SAN-CW-CD-SP-D	84260.276	PLATFORM LEVEL	CONCOURSE LEVEL	4005.072
SAN-CW-CD-SP-D	84526.269	PLATFORM LEVEL	CONCOURSE LEVEL	4273.296
SAN-CW-CD-SP-D	84377.265	PLATFORM LEVEL	CONCOURSE LEVEL	4123.944
SAN-CW-CD-SP-D	84517.523	PLATFORM LEVEL	CONCOURSE LEVEL	4264.152
SAN-CW-CD-SP-D	84539.410	PLATFORM LEVEL	CONCOURSE LEVEL	4285.488
SAN-CW-CD-SP-D	84388.511	PLATFORM LEVEL	CONCOURSE LEVEL	4136.136
SAN-CW-CD-SP-D	84267.568	PLATFORM LEVEL	CONCOURSE LEVEL	4014.216
SAN-CW-CD-SP-D	84533.563	PLATFORM LEVEL	CONCOURSE LEVEL	4279.392
SAN-CW-CD-SP-D	84384.556	PLATFORM LEVEL	CONCOURSE LEVEL	4130.040
SAN-CW-CD-SP-D	84524.814	PLATFORM LEVEL	CONCOURSE LEVEL	4270.248
SAN-CW-CD-SP-D	84285.297	PLATFORM LEVEL	CONCOURSE LEVEL	4032.504
SAN-CW-CD-SP-D	84402.259	PLATFORM LEVEL	CONCOURSE LEVEL	4148.328
SAN-CW-CD-SP-D	84261.569	PLATFORM LEVEL	CONCOURSE LEVEL	4008.120
SAN-CW-CD-SP-D	84529.669	PLATFORM LEVEL	CONCOURSE LEVEL	4276.344
SAN-CW-CD-SP-D	84518.796	PLATFORM LEVEL	CONCOURSE LEVEL	4264.152
SAN-CW-CD-SP-D	84378.537	PLATFORM LEVEL	CONCOURSE LEVEL	4123.944

รูปที่ 4.6 แสดง Pipe Schedule ที่ได้จากการปฏิบัติงานจริงของสถานีรถไฟฟ้าใต้ดินสายสีส้ม

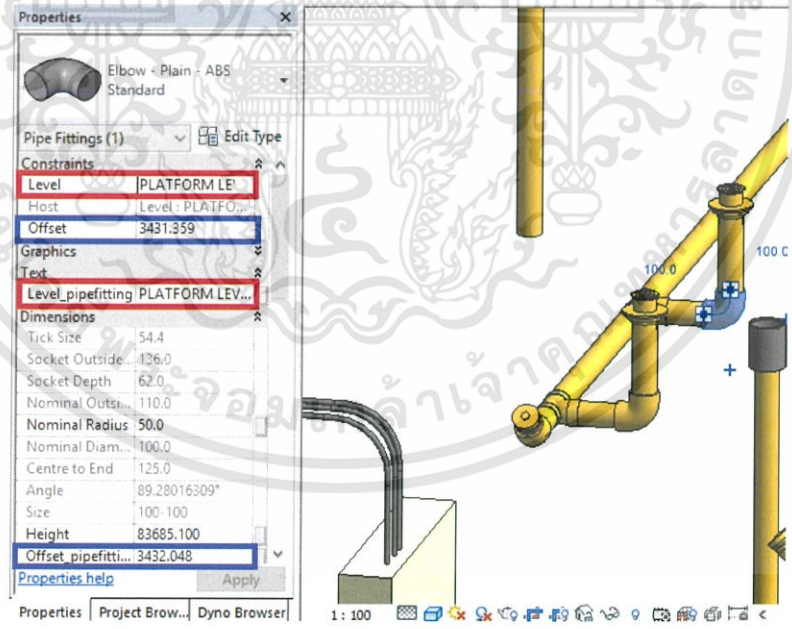
ในขณะเดียวกันอุปกรณ์ข้อต่อสามารถระบุตำแหน่งได้ในทันที ดังแสดงในรูปที่ 4.7 เนื่องจากอุปกรณ์ข้อต่อมีขนาดเล็ก ทำให้ทราบถึงชั้นระดับที่อุปกรณ์ข้อต่ออยู่ได้ ดังนั้นจึงสามารถโอนถ่ายข้อมูลจากพารามิเตอร์ ‘Level_pipefitting’ ไปยัง ‘Level’ ‘Offset_pipefitting’ และ ‘Offset’ ตามลำดับ โดยไม่ต้องคำนึงถึงความสับสนของโปรแกรมเช่นเดียวกับในกรณีของท่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<Pipe Fitting Schedule>

A	B	C	D
Family and Type	Height	Offset_pip	Level_pipefitting
M_Elbow - GALVANIZED - Sch 40: Standard	84387.920	4133.088	PLATFORM LEVEL
Tee Sanitary - PVC - Sch 40 - DWV: Standard	84389.180	4136.136	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: Standard	84442.790	4187.952	PLATFORM LEVEL
Tee Sanitary - PVC - Sch 40 - DWV: Standard	84444.050	4191.000	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: Standard	84439.740	4184.904	PLATFORM LEVEL
Tee Sanitary - PVC - Sch 40 - DWV: Standard	84441.000	4187.952	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: Standard	84430.600	4175.760	PLATFORM LEVEL
Tee Sanitary - PVC - Sch 40 - DWV: Standard	84431.860	4178.808	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: Standard	84442.810	4187.952	PLATFORM LEVEL
Tee Sanitary - PVC - Sch 40 - DWV: Standard	84444.050	4191.000	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: Standard	84436.690	4181.856	PLATFORM LEVEL
Tee Sanitary - PVC - Sch 40 - DWV: Standard	84437.950	4184.904	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: Standard	84455.010	4200.144	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: Standard	84430.620	4175.760	PLATFORM LEVEL
Tee Sanitary - PVC - Sch 40 - DWV: Standard	84437.950	4184.904	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: Standard	84421.450	4166.616	PLATFORM LEVEL
Tee Sanitary - PVC - Sch 40 - DWV: Standard	84422.710	4169.664	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	83807.020	3553.968	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	84200.210	3947.160	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	84037.660	3782.568	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	84194.110	3941.064	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	84037.860	3782.568	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	84194.110	3941.064	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	83961.680	3706.368	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	84197.160	3944.112	PLATFORM LEVEL
M_Elbow - GALVANIZED - Sch 40: M_Elbow-Ca	83992.940	3739.896	PLATFORM LEVEL

รูปที่ 4.7 แสดง Pipe Fitting Schedule ที่ได้จากการปฏิบัติงานจริงของสถานีรถไฟฟ้าใต้ดินสายสีส้ม



รูปที่ 4.8 แสดงชั้นระดับ (กรอบสีแดง) และค่า Offset (กรอบสีน้ำเงิน) ในหน้าต่าง Properties ของอุปกรณ์ข้อต่อ

จากรูปที่ 4.8 จะสังเกตได้ว่าค่าในพารามิเตอร์ ‘Level’ และ ‘Offset’ มีค่าตรงกันหรือใกล้เคียงกับพารามิเตอร์ ‘Level_pipefiting’ และ ‘Offset_pipefitting’ ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การคำนวณหาขนาดของมอเตอร์ปั๊มน้ำในระบบประปาและสุขาภิบาลที่ซ้ำซ้อน

จากการใช้วิธีที่ 1 และวิธีที่ 2 ในการคำนวณหาขนาดมอเตอร์ปั๊มน้ำ ผลลัพธ์ที่ได้เป็นเช่นเดียวกัน นั่นคือตารางแสดงคุณสมบัติต่างๆ ของท่อ อุปกรณ์ข้อต่อและ Pipe Accessory ทั้งหมดในระบบดังรูปที่ 4.9 โดยภายในตารางจะแสดงข้อมูลต่างๆ ดังนี้

Section Ref.	Type	Length(mm)	Diameter(mm)	Roughness Coefficient	Flow Rate(l/s)	Velocity(m/s)	Friction Head(ft/100)	Equivalent Length(ft)	Dynamic Loss(m)
1	Straight pipe	2025.26181	100	120	4.1	0.522028	0.430528		0.002658
1.5	Check valve		100	120	4.1	0.522028	0.421635	33.6	0.043181
2	Straight pipe	150	100	120	4.1	0.522028	0.430528		0.000197
2.5	Gate valve		100	120	4.1	0.522028	0.421635	2.7	0.00347
3	Straight pipe	187.614063	100	120	4.1	0.522028	0.430528		0.000246
3.5	Tee		100	120	4.1	0.522028	0.421635	7.6	0.009767
4	Straight pipe	790	100	120	4.1	0.522028	0.430528		0.001037
4.5	Tee		100	120	4.1	0.522028	0.421635	7.6	0.009767
5	Straight pipe	168	100	120	4.1	0.522028	0.430528		0.00022
5.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
6	Straight pipe	2434.175697	100	120	4.1	0.522028	0.430528		0.003194
6.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
7	Straight pipe	3280.17225	100	120	4.1	0.522028	0.430528		0.004304
7.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
8	Straight pipe	13223.29078	100	120	4.1	0.522028	0.430528		0.017352
8.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
9	Straight pipe	33.2121	100	120	4.1	0.522028	0.430528		0.000044
9.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
10	Straight pipe	4974.073882	100	120	4.1	0.522028	0.430528		0.006527
10.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
11	Straight pipe	2533.648562	100	120	4.1	0.522028	0.430528		0.003325
11.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
12	Straight pipe	9358.647809	100	120	4.1	0.522028	0.430528		0.012281
12.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
13	Straight pipe	8846.611939	100	120	4.1	0.522028	0.430528		0.011609
13.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
14	Straight pipe	8894.576391	100	120	4.1	0.522028	0.430528		0.011672
14.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
15	Straight pipe	22325.94436	100	120	4.1	0.522028	0.430528		0.029297
15.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
16	Straight pipe	1006.841754	100	120	4.1	0.522028	0.430528		0.001321
16.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
17	Straight pipe	12141.50605	100	120	4.1	0.522028	0.430528		0.015933
17.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
18	Straight pipe	70	100	120	4.1	0.522028	0.430528		0.000092
18.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
19	Straight pipe	65.441559	100	120	4.1	0.522028	0.430528		0.000086
19.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
20	Straight pipe	431.037311	100	120	4.1	0.522028	0.430528		0.000566
20.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
21	Straight pipe	3664.285621	100	120	4.1	0.522028	0.430528		0.004808
21.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
22	Straight pipe	269.688336	100	120	4.1	0.522028	0.430528		0.000354
22.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
23	Straight pipe	3664.285621	100	120	4.1	0.522028	0.430528		0.004808
23.5	Elbow		100	120	4.1	0.522028	0.421635	10.1	0.01298
24	Straight pipe	434.274353	100	120	4.1	0.522028	0.430528		0.00057
25	Straight pipe	1675	100	120	4.1	0.522028	0.430528		0.002198
								Total friction Head Loss(m)	0.447504
								Static Loss(m)	26.957815
								Total Head Loss(m)	27.405319
								Outlet pressure(m)	2.19
									29.595319
								(say)	30

Power	
Assumed pump efficiency	60%
Assumed motor efficiency	80%
Pump motor size(KW)	2.5138125

รูปที่ 9 แสดงตารางคุณสมบัติต่างๆท่อ อุปกรณ์ข้อต่อและ Pipe Accessory ทั้งหมดในระบบประปาและสุขาภิบาล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อคำนวณขนาดกำลังที่ถูกใช้งานในระบบแล้ว โดยกำหนดให้มอเตอร์มีประสิทธิภาพ 80% และปั๊มมีประสิทธิภาพ 60% จากนั้นจึงนำค่าที่ได้เลือกขนาดมอเตอร์ที่เหมาะสมของปั๊มน้ำ

4. การสร้าง Grid Line ในรูปแบบ 3 มิติ

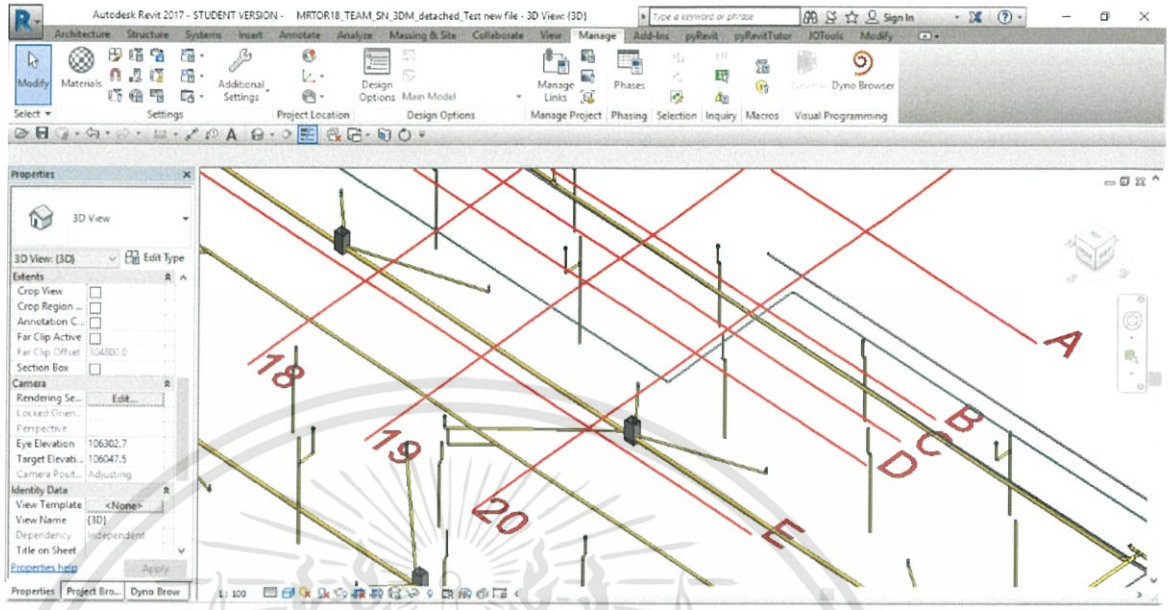
เนื่องจากในการเปิดแบบจำลอง 3 มิติ Grid Lines จะไม่สามารถแสดงผลออกมาได้ ทำให้เกิดความล่าช้าในการปฏิบัติงานเมื่อต้องการทราบตำแหน่งของอุปกรณ์ภายในแบบ ผู้ศึกษาจึงนำซอฟท์แวร์ Dynamo เข้ามาเพื่อเขียนโค้ดเพิ่มเติมในการสร้าง Grid Line ในรูปแบบ 3 มิติ

เมื่อทำการรันโค้ดในซอฟท์แวร์ Dynamo ผ่านโปรแกรม Autodesk Revit แล้ว Grid Lines จะปรากฏเมื่อเปิดแบบจำลอง 3 มิติของชิ้นงานซึ่งประกอบไปด้วยทั้ง Grid Lines ในแนวนอนและแนวตั้ง ดังแสดงในรูปที่ 4.10

รูปที่ 4.10 แสดง Grid Lines ในรูปแบบ 3 มิติ

จากนั้นจึงนำโค้ดในซอฟท์แวร์ Dynamo ที่ได้จากการทดลองมาใช้ในการปฏิบัติงานจริงในสถานีรถไฟฟ้าใต้ดินสายสีส้มในระบบสุขาภิบาลและการประปา ได้ผลลัพธ์เช่นเดียวกันกับการทดลองกล่าวคือชิ้นงานพร้อมด้วย Grid Lines ทั้งในแนวตั้งและแนวนอนจะปรากฏเมื่อเปิดชิ้นงานในรูปแบบ 3 มิติดังแสดงในรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



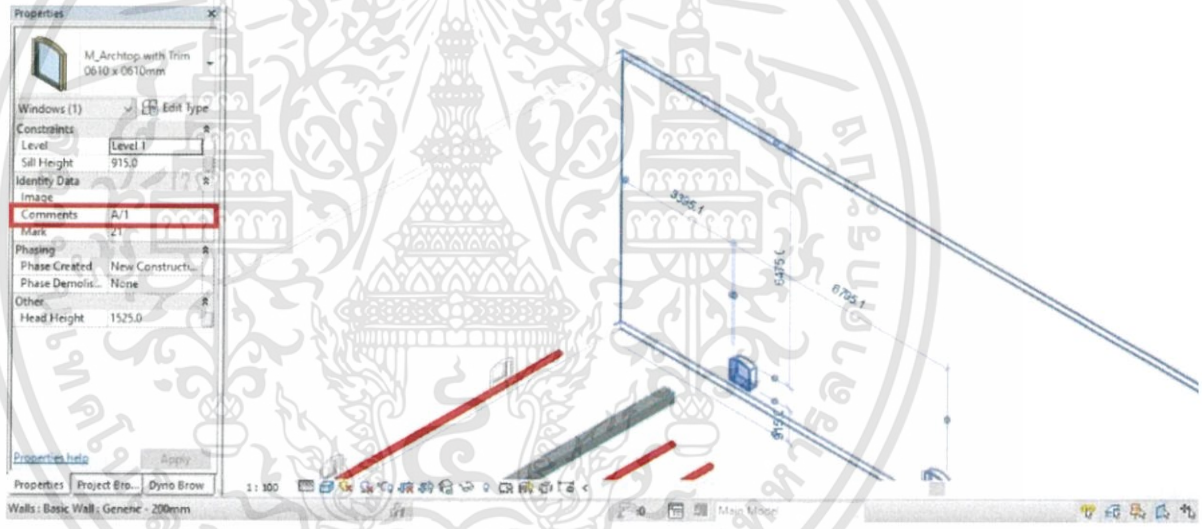
รูปที่ 4.11 แสดง Grid Lines ในรูปแบบ 3 มิติเมื่อนำไปใช้ในการปฏิบัติงานจริงในสถานีรถไฟฟ้าใต้ดินสายสีส้มในระบบสุขาภิบาลและการประปา

5. การระบุตำแหน่งของวัตถุที่ต้องการในแบบจำลอง 3 มิติด้วย Grid Line

ปัญหาที่เกิดขึ้นนี้สามารถแก้ไขได้โดยการเขียนโค้ดในซอฟท์แวร์ Dynamo ผลลัพธ์ที่ได้จะทำให้สามารถทราบตำแหน่งของวัตถุที่ต้องการในแต่ละ Category ของชิ้นงานโดยระบุเป็น Grid Line ที่ใกล้เคียงที่สุดของวัตถุนั้นๆ ซึ่งตำแหน่งของ Grid Line ที่ได้จะปรากฏอยู่ในพารามิเตอร์ 'Comments' ดังรูปที่ 4.12 และ 4.13 แต่ในที่นี้การระบุตำแหน่งของวัตถุยังไม่สามารถใช้ได้กับ Category ประเภท Pipe และ Duct หรือสามารถบอกได้อีกนัยหนึ่งว่าการเขียนโค้ดในซอฟท์แวร์ Dynamo นี้ยังไม่สามาถให้ผลได้กับวัตถุของ Category ที่มีความกว้างและความยาว (ซ้อนทับกันบน Grid Line หลายเส้น) เนื่องจากโค้ดไม่สามารถระบุว่าควรใช้จุดใดบนวัตถุเพื่อระบุตำแหน่ง ทำให้เกิดค่าข้อมูลที่ผิดพลาด (Error) ในที่นี้ทางผู้ศึกษาได้ใช้หน้าต่างชนิด M_Archtop with trim เป็น Element ในการทำการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.12 แสดงตำแหน่งของ Element ในมุมมอง 2 มิติ (กรอบสีแดง)

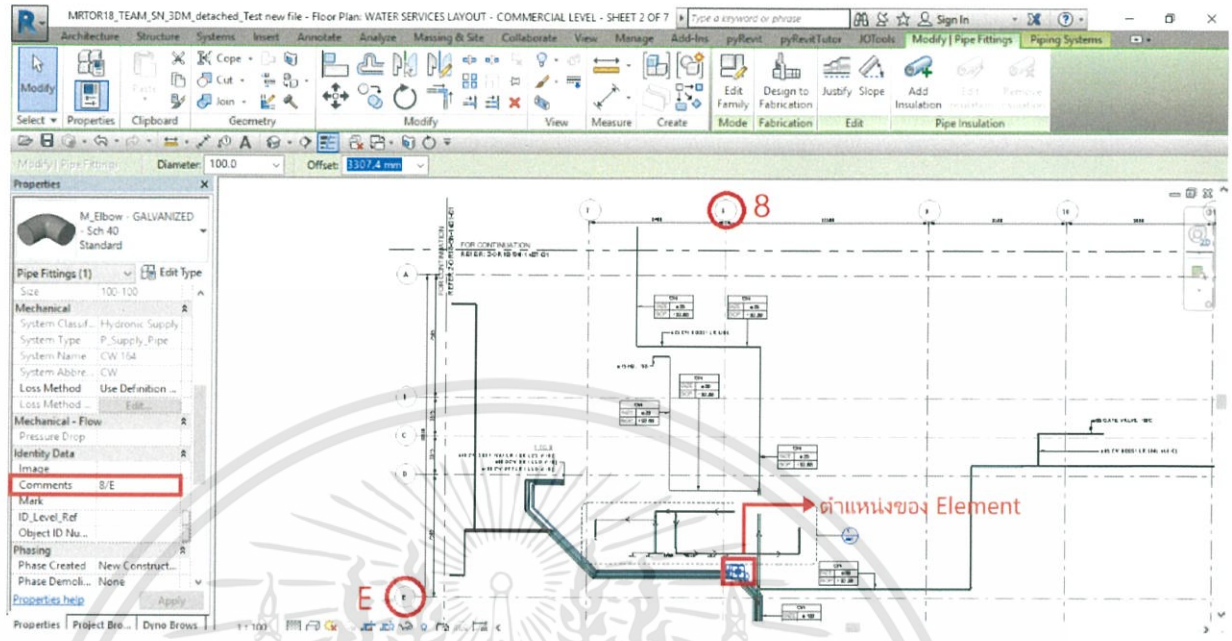


รูปที่ 4.13 แสดงตำแหน่งของ Element ในมุมมอง 3 มิติ (กรอบสีแดง)

จากแบบแปลนของแบบจำลองในรูปที่ 4.12 จะสังเกตได้ว่าหน้าต่างอยู่ใกล้เคียงกับตำแหน่งของ Grid Line A ในแนวนอนและ Grid Line 1 ในแนวตั้ง เมื่อรันโค้ดในซอฟท์แวร์ Dynamo แล้ว ตำแหน่งของหน้าต่างจะปรากฏอยู่ในพารามิเตอร์ ‘Comments’ ในหน้าต่าง Properties ของวัตถุที่ต้องการโดยจะปรากฏอยู่ในรูปแบบของ Grid Line แนวนอน/Grid Line แนวตั้ง

ตัวอย่างเช่น A/1 หมายถึงวัตถุที่ต้องการทราบตำแหน่งอยู่ใกล้กับ Grid Line A ในแนวนอน และ Grid Line 1 ในแนวตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 แสดงตำแหน่งของวัตถุในมุมมอง 3 มิติเมื่อนำไปใช้ในการปฏิบัติงานจริงในสถานีรถไฟฟ้าใต้ดินสายสีส้มในระบบสุขาภิบาลและการประปา

เมื่อนำโค้ดจากซอฟท์แวร์ Dynamo ที่ได้จากการทดลองมาใช้ในการปฏิบัติงานจริงในสถานีรถไฟฟ้าใต้ดินสายสีส้มในระบบสุขาภิบาลและการประปา จะได้ผลลัพธ์เช่นเดียวกันกับการทดลอง กล่าวคือ เมื่อเลือกวัตถุที่ต้องการทราบตำแหน่งแล้ว ตำแหน่ง Grid Line ที่ใกล้ที่สุดของวัตถุจะปรากฏในพารามิเตอร์ ‘Comment’ ในหน้าต่าง Properties ในที่นี้ผู้ศึกษาได้เลือก M_Elbow – Galvanized – Sch – 40 ซึ่งอยู่ในตำแหน่งที่ใกล้กับ Grid Line 8 ในแนวตั้งและ Grid Line E ในแนวนอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การเปรียบเทียบระยะเวลาและค่าชั่วโมงแรงงานในการปฏิบัติงานด้วยวิธีการเดิมและวิธีการใหม่

ตารางที่ 4.1 แสดงระยะเวลาและค่าชั่วโมงแรงงานในการปฏิบัติงานด้วยวิธีการเดิมและวิธีการใหม่ของทุกสถานีในสถานีรถไฟฟ้าใต้ดินสายสีส้ม

ปัญหา	ชั่วโมงแรงงาน (ชั่วโมง)								
	วิธีการเดิม							วิธีการใหม่ (ปีแรก)	วิธีการใหม่ (ปีถัดไป)
	OR13	OR14	OR15	OR16	OR17	OR18	OR19	ทุกสถานี	
การแสดงมุมมองของส่วนที่เฉพาะเจาะจงของแบบจำลอง 3 มิติ โดยใช้เครื่องมือ Section Views	2	2	2	2	2	2	2	16+0.56 = 16.56	0.56
การเพิ่มพารามิเตอร์ Reference Level ของท่อใน Schedule	16	16	16	16	16	16	16	75 + 0.35 = 75.35	0.35
การคำนวณหาขนาดของมอเตอร์ปั๊มน้ำในระบบประปาและสุขาภิบาลที่ซ้ำซ้อน	8	8	8	8	8	8	8	65 + 17.50 = 82.50	17.50
การสร้าง Grid Line ในรูปแบบ 3 มิติ	4	4	4	4	4	4	4	24+0.23 = 24.23	0.23
การระบุตำแหน่งของวัตถุที่ต้องการในแบบจำลอง 3 มิติด้วย Grid Line	4	4	4	4	4	4	4	24+0.35 = 24.35	0.35
รวมชั่วโมงแรงงาน (ชั่วโมง)	34	34	34	34	34	34	34	222.99	18.99
รวมชั่วโมงแรงงานต่อวัน (ชั่วโมง)	29.75							27.87	2.37
รวมชั่วโมงแรงงานต่อเดือน (ชั่วโมง)	0.99							0.93	0.08
รวมค่าชั่วโมงแรงงานต่อปี (บาท)	23,800							22,299	1,899
เปอรเซ็นต์ความแตกต่างในระยะเวลา 2 ปี	{[(23,800*2) - (22,299 + 1,899)]/47600}*100 = 49.16%								

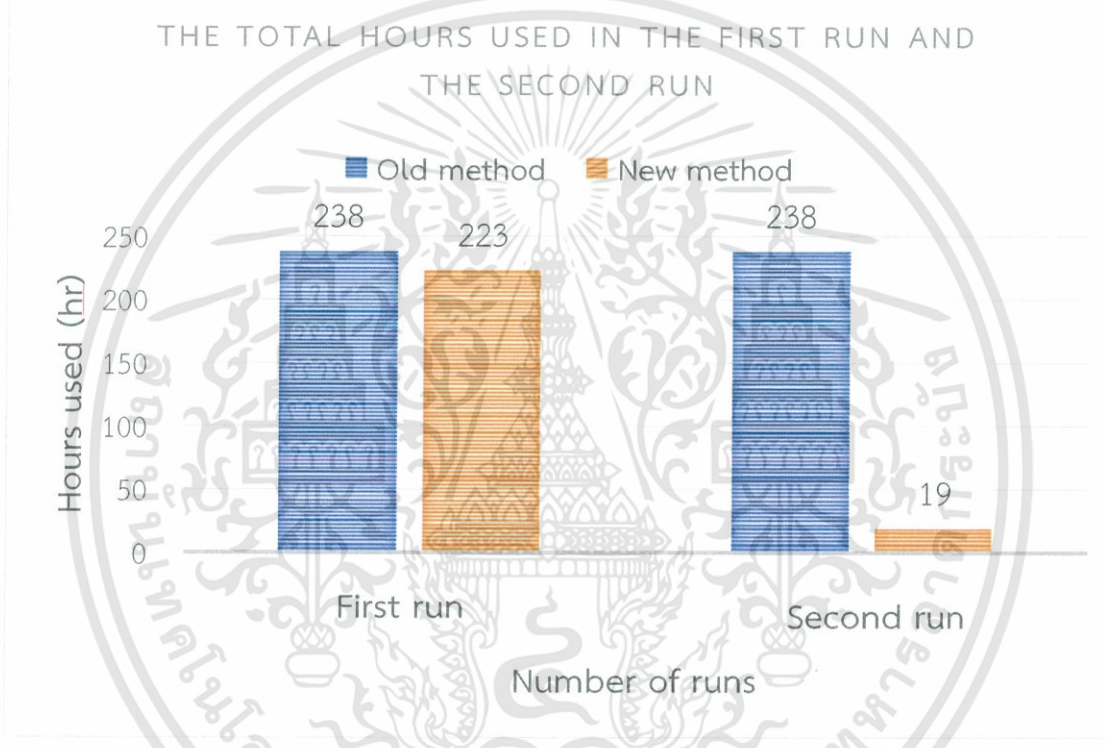
<u>หมายเหตุ</u>

1. ค่าชั่วโมงแรงงานของพนักงาน 24000 บาทต่อเดือน คิดเป็น 100 บาทต่อชั่วโมงและไม่คำนึงถึงค่าล่วงเวลาในการปฏิบัติงาน
2. การคำนวณในครั้งนี้ไม่คำนึงถึงค่าใช้จ่ายของซอฟท์แวร์และอุปกรณ์ในการปฏิบัติงาน

จากผลการวิเคราะห์จากตารางที่ 4.1 ทำให้สามารถนำมาสร้างกราฟดังรูปที่ 4.15 และ 4.16 ได้ ซึ่งในรูปที่ 4.15 แสดงให้เห็นว่าชั่วโมงแรงงานในการแก้ไขปัญหาด้วยวิธีการเดิมซึ่งใช้ระยะเวลาในการแก้ไขใกล้เคียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

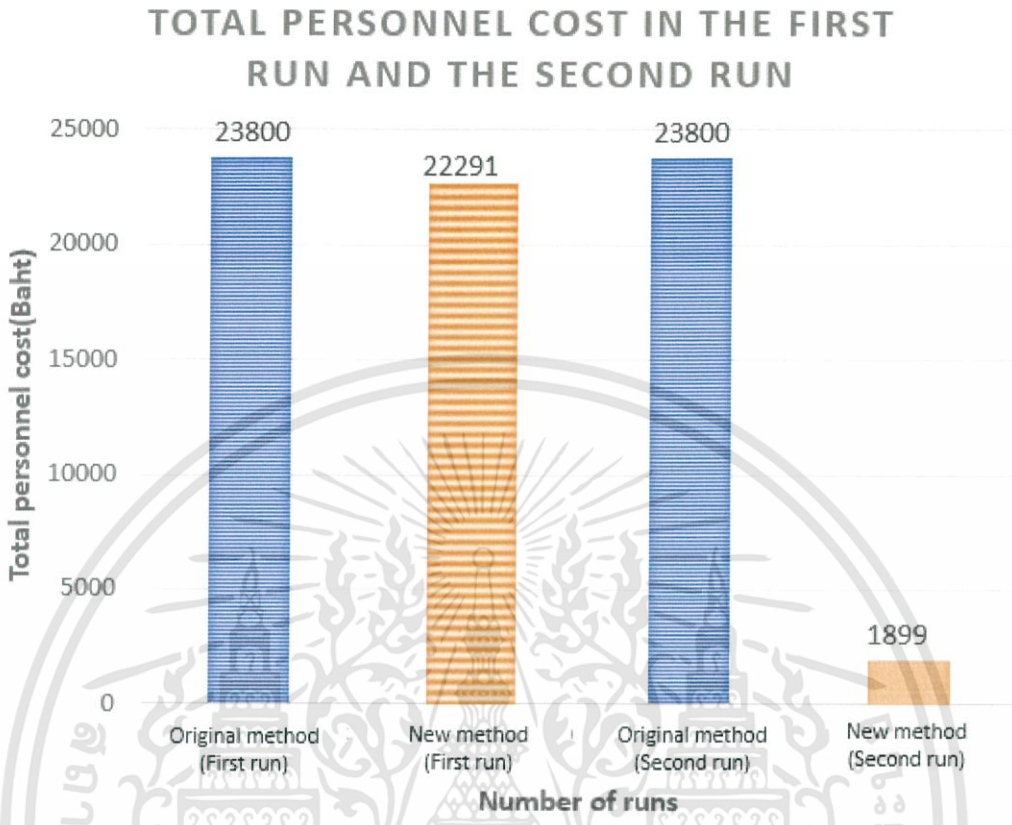
กันในทุกสถานีรวมเป็น 238 ชั่วโมง และเมื่อคำนวณเป็นค่าชั่วโมงแรงงานต่อปีจะได้ 23,800 ชั่วโมงต่อปี แต่ในช่วงปีแรกของการแก้ไขปัญหาด้วยวิธีการใหม่จะใช้ระยะเวลาถึง 222.99 ชั่วโมง คิดเป็น 22,299 บาทต่อปี เนื่องจากใช้เวลาในทำการศึกษาและทดลองปฏิบัติงานจริง หลังจากนั้นในปีถัดมา ระยะเวลาที่ใช้ในการแก้ไขปัญหาจะลดลงอย่างเห็นได้ชัดนั่นคือ 18.99 ชั่วโมงต่อปี คิดเป็น 1,899 บาทต่อปี เมื่อรวมเวลาในการปฏิบัติงานทั้งสองครั้ง การปฏิบัติงานด้วยวิธีการเดิมจะใช้ระยะเวลารวมทั้งหมด 576 ชั่วโมง คิดเป็น 47,600 บาท และระยะเวลาในการแก้ไขปัญหาด้วยวิธีการใหม่รวม 241.98 ชั่วโมง คิดเป็น 24,198 บาท แสดงให้เห็นว่าหากใช้วิธีการใหม่ในการแก้ไขปัญหาจะสามารถลดค่าใช้จ่ายไปได้ถึงร้อยละ 49.16



รูปที่ 4.15 แสดงกราฟเปรียบเทียบชั่วโมงแรงงานที่ใช้ในระหว่างการแก้ไขด้วยวิธีการเดิมและวิธีการใหม่

นอกจากนี้จะสังเกตเห็นได้ว่าเวลาที่ใช้ในวิธีการแก้ปัญหาด้วยวิธีการใหม่ในช่วงการปฏิบัติงานในครั้งแรกจะต้องใช้เวลาในการศึกษางานภายในสถานีรถไฟฟ้าสายสีส้ม การศึกษาซอฟท์แวร์และการแก้ไขปัญหา ทำให้ระยะเวลาที่ใช้ในการแก้ปัญหาด้วยวิธีการใหม่ในการปฏิบัติงานครั้งแรกและครั้งต่อไปมีความแตกต่างกันเป็นอย่างมาก ในขณะเดียวกัน การแก้ไขปัญหาด้วยวิธีการเดิมจะไม่มีการเปลี่ยนแปลงของระยะเวลา สาเหตุที่เป็นเช่นนี้อันเนื่องมาจากการใช้กระบวนการทำงานเหมือนเดิมในทุกครั้ง ในส่วนของค่าใช้จ่ายในส่วนของแรงงานนั้นสามารถเห็นความแตกต่างได้อย่างชัดเจนในรูปที่ 16 เมื่อสังเกตในกรณีการแก้ปัญหาด้วยวิธีการใหม่ในการปฏิบัติงานครั้งแรกและครั้งถัดไป จะมีความแตกต่างของค่าใช้จ่ายถึง 14 เท่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 แสดงกราฟเปรียบเทียบจำนวนเงินที่ใช้ในระหว่างการแก้ไขด้วยวิธีการเดิมและวิธีการใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

การทำโครงงานสหกิจศึกษาครั้งนี้มีวัตถุประสงค์เพื่อลดระยะเวลาการทำงานในกระบวนการแบบจำลองสารสนเทศอาคารโดยการใช้โปรแกรม Autodesk Revit ซอฟท์แวร์ Dynamo โปรแกรม Microsoft Excel และภาษา Python เพื่อทำแบบจำลอง 3 มิติของโครงสร้างและระบบภายในสถานีรถไฟฟ้าสายสีส้มในสัญญาที่ 1 และ 2 อีกทั้งยังลดค่าใช้จ่ายในส่วนของแรงงาน นอกจากนี้ยังช่วยหลีกเลี่ยงการทำงานที่ซ้ำซ้อนซึ่งเป็นสาเหตุให้เกิดข้อมูลที่ผิดพลาดและโครงงานสหกิจศึกษานี้จะทำให้การทำงานมีความละเอียดและครอบคลุมมากยิ่งขึ้น สามารถสรุปผลวิธีการแก้ไขปัญหาต่างๆ ได้ดังนี้

1. การแสดงมุมมองของส่วนที่เฉพาะเจาะจงของแบบจำลอง 3 มิติโดยใช้เครื่องมือ Section Views ผลลัพธ์ที่แสดงออกมาเป็นการแสดงถึงชั้นระดับทั้งหมดที่อยู่ในโครงการ สามารถแก้ไขได้โดยการใช้เครื่องมือ Section ในโปรแกรม Autodesk Revit ได้โดยตรง
2. ใช้ภาษา Python และรันผ่าน Interactive Python Shell ซึ่งเป็นเครื่องมือที่อยู่ในโปรแกรม Autodesk Revit ในการเขียนโค้ดเพื่อเพิ่มพารามิเตอร์ Reference Level ของท่อ อีกทั้งยังสามารถระบุชั้นระดับที่ถูกต้องของท่อชนิดกลม ท่อชนิดเหลี่ยมและอุปกรณ์ข้อต่อได้
3. การคำนวณหาขนาดของมอเตอร์ปั๊มน้ำในระบบประปาและสุขาภิบาลที่ซ้ำซ้อนสามารถทำได้ 2 วิธี นั่นคือการใช้ความสัมพันธ์ของโปรแกรม Autodesk Revit และโปรแกรม Microsoft Excel ในการคำนวณและการคำนวณผ่านซอฟท์แวร์ Dynamo ซึ่งผลลัพธ์ที่ได้จะคล้ายคลึงกันนั่นคือได้ตารางแสดงคุณสมบัติของท่อ อุปกรณ์ข้อต่อ (Pipe Fitting) และ Pipe Accessory ทั้งหมดในระบบในโปรแกรม Microsoft Excel
4. การสร้าง Grid Line ในรูปแบบ 3 มิติและการระบุตำแหน่งหรือ Grid Line ของวัตถุที่ต้องการสามารถทำได้โดยการเขียนโค้ดในซอฟท์แวร์ Dynamo

จากวิธีการแก้ไขปัญหาที่กล่าวมาข้างต้น ในปีแรกของการแก้ไขปัญหาเหล่านี้ภายในสถานีรถไฟฟ้าใต้ดินสายสีส้มทั้งหมดจะใช้เวลาทั้งหมด 222.99 ชั่วโมง คิดเป็นเงิน 22,299 บาทต่อปี เนื่องจากต้องใช้ระยะเวลาในการศึกษาหาข้อมูลเกี่ยวกับปัญหาเหล่านี้, โปรแกรมและซอฟท์แวร์ต่างๆ ส่วนในปีถัดมาจะใช้เวลาเพียงแค่รันโปรแกรมเท่านั้น ซึ่งเป็นระยะเวลาทั้งหมด 18.99 ชั่วโมง คิดเป็น 1,899 บาทต่อปี แต่จะใช้เวลาถึง 238 ชั่วโมงในการแก้ไขปัญหาเหล่านี้ด้วยวิธีการเดิมซึ่งคิดเป็นเงินได้ 23,800 บาทต่อปี เมื่อนำค่าใช้จ่ายของค่าชั่วโมงแรงงานของทั้งสองวิธีการนี้ในระยะเวลา 2 ปี มาเปรียบเทียบจะแสดงให้เห็นว่าการปฏิบัติงานด้วยวิธีการใหม่สามารถลดค่าใช้จ่ายในส่วนของแรงงานได้ถึงร้อยละ 49.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ข้อเสนอแนะ

1. การเขียนโค้ดในซอฟท์แวร์ Dynamo ในโครงงานเล่มนี้อาจก่อให้เกิดปัญหาต่อบุคคลที่ไม่มีความเชี่ยวชาญในการใช้ซอฟท์แวร์ เนื่องจากโค้ดมีความซับซ้อนและมีโหนดที่ต้องเชื่อมต่อกันมากอาจทำให้เมื่อรันโค้ดแล้วเกิดข้อผิดพลาดและใช้งานโค้ดเหล่านี้ไม่ได้
2. หากใช้ซอฟท์แวร์ Dynamo และแพ็คเกจที่ไม่ถูกเวอร์ชั่นจะทำให้ไม่สามารถรันโค้ดและเปิดชิ้นงานขึ้นมาได้
3. ในการระบุตำแหน่งหรือ Grid Line ของวัตถุยังคงระบุได้เพียงแค่วัตถุที่มีความยาวและความกว้างในช่วง Grid Line ที่จำกัดเท่านั้น
4. การระบุชั้นระดับของท่อจากการเพิ่มพารามิเตอร์ Reference Level สามารถระบุได้เพียงท่อชนิดกลม ท่อชนิดเหลี่ยมที่วางตัวตามแนวนอนและอุปกรณ์ข้อต่อเท่านั้น ซึ่งไม่สามารถทำได้ในกรณีของท่อชนิดกลมและท่อชนิดเหลี่ยมที่วางตัวตามแนวตั้งและไม่ได้คำนึงถึงท่อที่มีความชัน
5. การแก้ไขปัญหาการ Sectionเพื่อแสดงเพียงมุมมองของวัตถุที่ต้องการอาจต้องแก้ไขเครื่องมือใน Database ของโปรแกรมโดยตรงเพื่อให้ได้วิธีการที่สะดวกและรวดเร็วมากยิ่งขึ้น
6. การคำนวณหาขนาดของมอเตอร์ปั๊มน้ำในระบบประปาและสุขาภิบาลมีข้อเสนอแนะดังต่อไปนี้

   6.1 อาจพัฒนาให้สร้างพารามิเตอร์ ‘System Type’ และพารามิเตอร์ ‘No.’ ในขณะการทำการวาดท่อได้ในทันทีเพื่อทำให้กระบวนการทำงานเป็นไปอย่างรวดเร็วมากยิ่งขึ้น

   6.2 ไม่สามารถกรองแยกเส้นทางเดินของท่อแต่ละเส้นทางได้ในซอฟท์แวร์ Dynamo ทำให้มีผลต่อการคำนวณหาขนาดของมอเตอร์ปั๊มน้ำในทุกสถานี เนื่องจากค่าที่ได้เป็นค่าที่ไม่ถูกต้อง ตัวอย่างเช่น พารามิเตอร์ ‘Static Loss’ นำค่าของท่อที่ระดับสูงที่สุดลบด้วยค่าของท่อที่ระดับต่ำที่สุด ซึ่งค่าของท่อที่ระดับสูงที่สุดและระดับต่ำที่สุดของแต่ละเส้นทางเดินของท่ออาจมีค่าที่ไม่เท่ากัน หากโปรแกรมสามารถเลือกท่อที่ระดับสูงที่สุดและต่ำที่สุดของแต่ละเส้นทางเดินของท่อออกมาได้ในทันที ก็จะมีประโยชน์เป็นอย่างมากในการทำงาน

   6.3 จากข้อที่ 6.2 เมื่อโปรแกรมนำค่าของท่อที่ระดับสูงที่สุดและต่ำที่สุดของท่อที่ไม่ถูกต้องแล้ว อาจส่งผลกระทบถึงค่าของพารามิเตอร์ ‘Flow Rate’ และ ‘Velocity’ ซึ่งมีค่าที่ไม่เท่ากันในแต่ละเส้นทางเดินของท่อ (ขึ้นอยู่กับพื้นที่หน้าตัดของท่อนั้นๆ) ทำให้ค่าของพารามิเตอร์ ‘Dynamic Loss’ ไม่ถูกต้องเช่นเดียวกันด้วย ซึ่งปัญหานี้อาจใช้ภาษา Python ในการเขียน Script เพื่อกำหนดค่าของอัตราการไหลของท่อในแต่ละท่อให้คงที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] สถาบันสถาปนิกสยาม. (2558). คู่มือปฏิบัติวิชาชีพ แนวทางการใช้งานแบบจำลองสารสนเทศอาคารสำหรับประเทศไทย (Thailand BIM Guildline), 1-8

[2] Ryan Duell. Tobias Hathorn. Tessa Reist Hathorn. (2013). Autodesk Revit Architecture 2014 essentials: Autodesk official press, 297-304

[3] Autodesk. (May 17, 2018). Schedule. September 18, 2018, from https://knowledge.autodesk.com/support/revit-products/learn-explore/caas/CloudHelp/cloudhelp/2018/ENU/Revit-DocumentPresent/files/GUID-F50D6FF4-859E-43A2-A2F6-81C84A1BA0EB-htm.html

[4] Britannica's editorial staff. *Plumbing*. September 18, 2018, from https://www.britannica.com/technology/plumbing

[5] Autodesk. (July 26, 2018). *Creating Piping Systems*. September 18, 2018, from https://knowledge.autodesk.com/support/revit-products/learn-explore/caas/CloudHelp/cloudhelp/2018/ENU/Revit-Model/files/GUID-647C6E66-C353-409D-B9E8-B22333D5252D-htm.html

[6] Autodesk. (May 17, 2018). Pipe Placement Tools. September 18, 2018, from https://knowledge.autodesk.com/support/revit-products/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Revit-Model/files/GUID-C8E11700-528D-45AE-8480-264858143188-htm.html

[7] Autodesk. (February 08, 2017). *About View Types.* September 18, 2018, from https://knowledge.autodesk.com/support/revit-products/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Revit-DocumentPresent/files/GUID-23418A27-8135-40C7-8078-233FA0B8526B-htm.html

[8] Autodesk. (April 10, 2018). *Grids.* September 18, 2018, from https://knowledge.autodesk.com/support/revit-products/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Revit-Model/files/GUID-DBD917DB-B9F6-417B-B799-5F412A624EB4-htm.html

[9] Ngoc Nguyen. (December 8, 2016). *Create Revit 3D Grids with Dynamo*. September 18, 2018, from https://www.linkedin.com/pulse/create-revit-3d-grids-dynamo-ngoc-nguyen

[10] *Lookup Table*. (February 25, 2011). September 10, 2018, from https://compewter.wordpress.com/2011/02/25/lookup-table-%E0%B8%84%E0%B8%B7%E0%B8%AD-%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/

[11] Autodesk.Help. (February 8, 2017). *About Lookup Tables*. October 2, 2018, from https://knowledge.autodesk.com/search-result/caas/CloudHelp/cloudhelp/2016/ENU/RevitLT-Customize/files/GUID-91270AEF-225A-49D7-BF84-1F44D1E3E216-htm.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[12] Dynamo. *What is Dynamo?*. September 18, 2018, from http://primer.dynamobim.org/en/01_Introduction/1_introduction.html

[13] Dynamo. *The anatomy of a Visual Program*. October 10, 2018, from http://primer.dynamobim.org/en/03_Anatomy-of-a-Dynamo-Definition/3-1_dynamo_nodes.html

[14] Python Software Foundation. *What is Python?*. October 8, 2018, from https://www.python.org/doc/essays/blurb/

[15] ยุทธนา คงคารัตน์. (2561). การศึกษาภาษา Python เพื่อประยุกต์ใช้กับ Add-ins pyRevit. (รายงาน, มหาวิทยาลัยศิลปากร)

[16] Microsoft. *.NET Framework Class Library*. October 8, 2018, from https://msdn.microsoft.com/en-us/library/gg145045(v=vs.110).aspx

[17] *Interfaces*. (February 12, 2016). October 10, 2018, from http://marcuscode.com/lang/csharp/interfaces

[18] Haibo Luo. (September 25, 2007). *IronPython: clr.AddReference*. October 8, 2018, from https://blogs.msdn.microsoft.com/haibo_luo/2007/09/25/ironpython-clr-addreference/

[19] IronPython. *Loading .NET assemblies*. October8, 2018, from http://ironpython.net/documentation/dotnet/

[20] Autodesk.Help. (August 15, 2018). *Restore RevitAPI.dll and RevitAPIUI.dll*. October 8, 2018, from https://knowledge.autodesk.com/support/revit-products/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/Revit-Customize/files/GUID-17C8D2AA-584B-4E24-9521-CDCD7CFD731D-htm.html

[21] Autodesk.Help. (April 21 2017). *About Macro Manager and the Revit Macro IDE*. October 9, 2018, from https://knowledge.autodesk.com/support/revit-products/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/Revit-Customize/files/GUID-071913D8-214A-45AB-A798-A81653E77F88-htm.html

[22] Marilyn Law. (April 24, 2014). *What is a Macro?*. October 10, 2018, from https://www.augi.com/articles/detail/getting-started-with-revit-macros

[23] Robert Vetrano. (July 15, 2006). *What is REVIT API?*. October 8, 2018, from https://forums.autodesk.com/t5/revit-structure-forum/what-is-revit-api/td-p/1705641

[24] Einar Raknes. (June 16, 2018). *How to select and understand the Library?*. October 9, 2018, from https://forum.dynamobim.com/t/how-to-select-and-understand-the-library/4641

[25] Revit API Docs. (2017). *Autodesk.Revit.UI Namespace*. October 8, 2018, from http://www.revitapidocs.com/2017/e86fd90a-8957-02a6-da7f-ced248966e3e.htm

[26] Revit API Docs. (2017). *Autodesk.Revit.UI Namespace*. October 8, 2018, from http://www.revitapidocs.com/2017/e86fd90a-8957-02a6-da7f-ced248966e3e.htm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[27] Revit API Docs. (2015). *FilteredElementCollector Class*. October 9, 2018, from http://www.revitapidocs.com/2015/263cf06b-98be-6f91-c4da-fb47d01688f3.htm

[28] Revit API Docs. (2017). *BuiltInCategory Enumeration*. October 9, 2018, from http://www.revitapidocs.com/2017/ba1c5b30-242f-5fdc-8ea9-ec3b61e6e722.htm

[29] Autodesk.Help. (February 8, 2017). *Transaction Classes.* October 9, 2018, from https://knowledge.autodesk.com/search-result/caas/CloudHelp/cloudhelp/2016/ENU/Revit-API/files/GUID-BECA30DB-23B4-4E71-BE24-DC4DD176E52D-htm.html

[30] Revit API Docs. (2015). *WhereElementIsNotElementType Method.* October 9, 2018, from http://www.revitapidocs.com/2015/061cbbb9-26f1-a8f8-a4b2-3d7ff0105199.htm

[31] Revit API Docs. (2015). *ToElements Method.* October 9, 2018, from http://www.revitapidocs.com/2015/732b4a0d-62d8-b86d-120b-8ea3d9713b34.htm

[32] Revit API Docs. (2015). *LookupParameter Method*. October 9, 2018, from http://www.revitapidocs.com/2015/4400b9f8-3787-0947-5113-2522ff5e5de2.htm

[33] Microsoft. (May 30, 2018). *asdouble function*. October 9, 2018, from https://docs.microsoft.com/en-us/windows/desktop/direct3dhlsl/asdouble

[34] Mozilla. *Element.id*. October 9, 2018, from https://developer.mozilla.org/en-US/docs/Web/API/Element/id

[35] TechTerms. *Integer*. October 9, 2018, from https://techterms.com/definition/integer

[36] Autodesk.Help. (February 8, 2017). *AsValueString() and SetValueString()*. October 9, 2018, from https://knowledge.autodesk.com/search-result/caas/CloudHelp/cloudhelp/2016/ENU/Revit-API/files/GUID-4D70F8D1-3CD4-4B4A-8FD7-75802A441036-htm.html

[37] ยุทธนา คงคารัตน์. (2561). การศึกษาภาษา Python เพื่อประยุกต์ใช้กับ Add-ins pyRevit. (รายงาน ,มหาวิทยาลัยศิลปากร)

[38] Mindphp. (March 14, 2560). *IDE คืออะไร*. October 9, 2018, from http://mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/2245-ide-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html

[39] Pycharm. *Pycharm Features*. October 10, 2018, from https://www.jetbrains.com/pycharm/features/

[40] เทิดศักดิ์ ยุทธเสรี. (2013). การสูญเสียความฝืดในระบบท่อ (Pipe friction losses). September 18 2018, from http://industrialpumps-tsy.blogspot.com/2013/09/pipe-friction-losses.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[41] Massey, Bernard Stanford. (1989). *Mechanics of fluids*, eighth edition, 2554

[42] Pipeflow. (November 12,
2018). *The Moody Chart.* September 19, 2018, from https://www.pipeflow.com/pipe-pressure-drop-calculations/pipe-friction-loss

[43] Harvey Wilson. (2014). *Equivalent Lengths of pipe Fitting and Valves*. September 18, 2018, from https://www.katmarsoftware.com/articles/pipe-fitting-equivalent-length.html

[44] Sean Moran. (2016).
Pump Sizing: Bridging the Gap Between Theory and Practice. October 25, 2018, from https://www.aiche.org/sites/default/files/cep/20161238_r.pdf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้