การออกแบบโปรแกรมวางผังคลังสินค้า

โดย
นายวัชราวุธ เจนเรือ

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต
สาขาวิชาคอมพิวเตอร์เพื่อการออกแบบทางสถาปัตยกรรม แผน ข แบบ ระดับปริญญามหาบัณฑิต
ภาควิชาเทคนิคสถาปัตยกรรม
บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร
ปีการศึกษา 2563
ลิขสิทธิ์ของบัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

การออกแบบโปรแกรมวางผังคลังสินค้า

โดย
นายวัชราวุธ เจนเรือ

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต
สาขาวิชาคอมพิวเตอร์เพื่อการออกแบบทางสถาปัตยกรรม แผน ข แบบ ระดับปริญญามหาบัณฑิต
ภาควิชาเทคนิคสถาปัตยกรรม
บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร
ปีการศึกษา 2563
ลิขสิทธิ์ของบัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

WAREHOUSE PLANNING PROGRAM

By

MR. Wacharawuth JANRUEA

An Independent Study Submitted in Partial Fulfillment of the Requirements

for Master of Science (Computer-aided Architectural Design)

Department of Architectural Technology

Graduate School, Silpakorn University

Academic Year 2020

Copyright of Graduate School, Silpakorn University

หัวข้อ	การออกแบบโปรแกรมวางผังคลังสินค้า
โดย	วัชราวุธ เจนเรือ
สาขาวิชา	คอมพิวเตอร์เพื่อการออกแบบทางสถาปัตยกรรม แผน ข แบบระดับปริญญามหาบัณฑิต
อาจารย์ที่ปรึกษาหลัก	ศาสตราจารย์ ฐิติพัฒน์ ประทานทรัพย์

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร ได้รับพิจารณาอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต

.....คณบดีบัณฑิตวิทยาลัย
(รองศาสตราจารย์ ดร.จุไรรัตน์ นันทานิช)

พิจารณาเห็นชอบโดย

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร.ปรีชญา มหัทธนทวี)

.....อาจารย์ที่ปรึกษาหลัก
(ศาสตราจารย์ฐิติพัฒน์ ประทานทรัพย์)

.....ผู้ทรงคุณวุฒิภายใน
(ศาสตราจารย์ฐิติพัฒน์ ประทานทรัพย์)

.....ผู้ทรงคุณวุฒิภายนอก
(ผู้ช่วยศาสตราจารย์ ดร.วัชระ เพียรสุภาพ)

61059304 : คอมพิวเตอร์เพื่อการออกแบบทางสถาปัตยกรรม แผน ข แบบ ระดับปริญญามหาบัณฑิต
คำสำคัญ : วางผังคลังสินค้า, โปรแกรมวางผังคลังสินค้า, สถาปนิก, สถาปัตยกรรม, โรงงานอุตสาหกรรม, คลังสินค้า

นาย วัชราวุธ เจนเรือ: การออกแบบโปรแกรมวางผังคลังสินค้า อาจารย์ที่ปรึกษาวิทยานิพนธ์ : ศาสตราจารย์ ฐิติพัฒน์ ประทานทรัพย์

การค้นคว้าอิสระเรื่องการออกแบบโปรแกรมวางผังคลังสินค้านี้มีจุดมุ่งหมายเพื่อพัฒนาการรูปแบบการทำงานในปัจจุบันของบริษัทให้คำปรึกษาและออกแบบงานสถาปัตยกรรมอุตสาหกรรมที่ผู้ทำการค้นคว้าทำงานอยู่ ให้สามารถแก้ไขปัญหาหลักของรูปแบบการทำงานของการวางผังคลังสินค้าแบบเดิมคือ ปัญหาความซับซ้อนของวิธีการทำงานและปัญหาระยะเวลาในการทำงานที่จำกัด โดยการนำเครื่องมือทางคอมพิวเตอร์ที่เรียกว่าโปรแกรมมิ่งด้วยการเขียนโปรแกรมด้วยโปรแกรม Dynamo ซึ่งเป็นส่วนเสริมการทำงานบนโปรแกรม Revit ที่ใช้ในการทำงานแบบปัจจุบันอยู่แล้ว มาช่วยในการออกแบบโปรแกรมสำหรับเป็นเครื่องมือช่วยในงานวางผังคลังสินค้าผ่านการศึกษารูปแบบการทำงานจากกลุ่มสถาปนิกผู้ออกแบบและวางผังคลังสินค้าโดยตรง เพื่อลดความซับซ้อนของการทำงานและระยะเวลาในการทำงานลงแต่ยังคงได้ผลลัพธ์ที่ถูกต้องแม่นยำสอดคล้องกับความต้องการการใช้งานของอาคารคลังสินค้าได้

61059304 : Major (Computer-aided Architectural Design)
Keyword : warehouse, planning, warehouse planning program, dynamo, revit, visual programming

MR. WACHARAWUTH JANRUEA : WAREHOUSE PLANNING PROGRAM THESIS ADVISOR : PROFESSOR THITIPAT PRATHARNSAP

This independent study in warehouse planning program design aims to develop the current working systems of the consulting and industrial architectural design firms where the researcher works. To be able to solve the main problem of the working systems of the conventional warehouse palnning are complexity in working process and time limit. By using a computer tool called programming by Dynamo, which is a visual programming of the Revit program used in the current work create a tool to help in warehouse planning through a study of working patterns from a group of architects, warehouse planning designers. This is to reduce the complexity of work and the time limit, but still get accurate results and meet the user requirments.

มหาวิทยาลัยศิลปากร

กิตติกรรมประกาศ

การค้นคว้าอิสระนี้เป็นผลสำเร็จได้ด้วยความกรุณาจาก ศาสตราจารย์ ฐิติพัฒน์ ประทานทรัพย์ อาจารย์ที่ปรึกษา ที่ให้คำแนะนำตั้งแต่ตอนต้นของแนวความคิดในการค้นคว้า ตลอดจนถึงชี้ให้เห็นถึงจุดบกพร่องที่ต้องแก้ไขในระหว่างการค้นคว้า จนกระทั้งโครงร่างการค้นคว้าอิสระนี้เสร็จสมบูรณ์ ขอขอบพระคุณอาจารย์เป็นอย่างสูง

การค้นคว้าอิสระนี้ได้รับการสนับสนุนจาก บริษัท ชัชวาลย์รอยัลแฮสโคนิ่ง จำกัด ในหลากหลายเรื่อง โดยเฉพาะเรื่องของเวลาและอุปกรณ์ที่ช่วยให้การค้นคว้านี้เสร็จสมบูรณ์ ขอขอบพระคุณเป็นอย่างสูง

การค้นคว้าอิสระนี้ได้คำปรึกษาจากผู้มีประสบการณ์ในงานออกแบบอาคารคลังสินค้าคือ ทีมงานแผนกสถาปัตยกรรม บริษัท ชัชวาลย์รอยัลแฮสโคนิ่ง จำกัด โดยเฉพาะ คุณปวันรัตน์ จิตกวิน หัวหน้า แผนกสถาปัตกรรม จนกระทั้งโครงร่างการค้นคว้าอิสระนี้เสร็จสมบูรณ์ ขอขอบพระคุณเป็นอย่างสูง

สุดท้ายนี้ขอขอบพระคุณ คุณพ่อ คุณแม่ และครอบครัวที่ให้การสนับสนุนนในทุกด้านเป็นอย่างดีมาโดยตลอด

วัชราวุธ เจนเรือ

มหาวิทยาลัยศิลปากร

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง
บทคัดย่อภาษาอังกฤษ ..... จ
กิตติกรรมประกาศ ..... ฉ
สารบัญ ..... ช
สารบัญภาพ ..... ฌ
บทที่ 1 บทนำ 1
1.1 ความเป็นมาของปัญหา 1
1.1.1 ปัญหาของการวิธีการวางผังอาคารคลังสินค้าในปัจจุบัน 1
1.1.1.1 ความซับซ้อนของวัตถุที่จัดเก็บ 1
1.1.1.2 เวลา 2
1.1.2 แนวทางการแก้ปัญหา 2
1.2 วัตถุประสงค์ของการวิจัย 3
1.3 ประโยชน์ของการวิจัย 3
1.4 ขอบเขตของการวิจัย 3
1.5 นิยามศัพท์เฉพาะ 4
1.6 ขั้นตอนและวิธีการศึกษา 4
บทที่ 2 วรรณกรรมและงานวิจัยที่เกี่ยวข้อง 6
2.1 ขั้นตอนการวางผังคลังสินค้าในปัจจุบัน 6
2.1.1 ขั้นตอนที่1 การรับข้อมูลและความต้องการที่เกี่ยวข้องกับการออกแบบและวางผังคลังสินค้า (Get requirements) 8
2.1.1.1 คลังสินค้าประเภทสินค้าสำเร็จ (Finish Good Warehouse) 9

2.1.1.2 คลังสินค้าประเภทวัตถุดิบ (Raw Material Warehouse) 10
2.1.2 ขั้นตอนที่2 การจัดกลุ่มวัตถุที่จัดเก็บ (Classification)..... 11
2.1.2.1 เงื่อนไขเรื่องอุณหภูมิ (Temperature)..... 11
2.1.2.2 เงื่อนไขประเภทสารเคมี (Classification and Labeling of Chemicals)... 12
2.1.3 ขั้นตอนที่3 การหาขนาดของแต่ละกลุ่มพื้นที่ (Create Space)..... 16
2.1.3.1 ประเภทพื้นที่คลังสินค้า..... 17
ประเภทที่1 พื้นที่การจัดเก็บ (Storage Areas)..... 17
ประเภทที่2 พื้นที่ขนส่งสินค้า (Loading Area)..... 20
2.1.3.2 วิธีคำนวณพื้นที่คลังสินค้า..... 22
วิธีคำนวณพื้นที่คลังประเภทที่1 พื้นที่การจัดเก็บ (Storage Areas)..... 22
วิธีคำนวณพื้นที่คลังประเภทที่2 พื้นที่ขนส่งสินค้า (Loading Area)..... 26
2.1.4 ขั้นตอนที่ 4 การจัดเรียงพื้นที่คลังสินค้าโดยมีผลลัพธ์การวางผังคลังสินค้าที่หลากหลายตามเงื่อนไขของวัตถุที่จัดเก็บ (Space Planning and Alternative Design)..... 27
2.1.5 ขั้นตอนที่5 การพิจารณาแบบที่เหมาะสมที่สุดกับการทำงานของโรงงานนั้นๆ (Discussion)..... 28
2.1.5.1 หลักเกณฑ์การพิจารณาเลือกแบบอาคารคลังสินค้า..... 29
2.2 ปัญหาของการวางผังในปัจจุบัน 29
2.2.1 ความซับซ้อนในการวางผังคลังสินค้า 29
2.2.1.1 ความซับซ้อนจากขั้นตอนการทำงาน 30
2.2.1.2 ความซับซ้อนจากเงื่อนไขในการวางผังคลังสินค้า..... 30
2.2.2 เวลา 30
2.2.2.1 ขั้นตอนการทำงานกับเวลา..... 31
สัปดาห์ที่ 1 31
สัปดาห์ที่ 2 31

สัปดาห์ที่ 3 32
2.3 แนวคิดเกี่ยวกับการแก้ปัญหาโดยการวางผังด้วยคอมพิวเตอร์ 32
2.3.1 Dynamo 33
2.3.2 Dynamo กับการวางผังคลังสินค้า 34
2.3.3 Refinery 36
2.3.4 Refinery กับการวางผังคลังสินค้า 38
บทที่ 3 วิธีการวิจัย 41
3.1 แนวความคิดของโปรแกรมวางผังคลังสินค้า (Warehouse Planning Program) 41
3.1.1 ใช้ Excel ในขั้นการรับข้อมูลความต้องการของลูกค้าที่มีผลต่อการออกแบบวางผังคลังสินค้า 42
3.1.1.1 ข้อมูลกลุ่มที่ 1 42
(1) พื้นที่ขอบเขตที่ดิน (Area) 42
(2) จำนวนของรถชนิดต่างๆที่ถูกใช้ในการขนส่ง (No. of Each Car Type) 42
(3) รายชื่อพื้นที่จัดเก็บสินค้าแต่ละประเภท (Name Space) 43
(4) จำนวนความต้องการจัดเก็บสินค้าในแต่ละประเภท (Capacity) .. 43
(5) จำนวนการซ้อนชั้นของการจัดเก็บสินค้าแต่ละประเภท (Level) .. 43
(6) กำหนดสีเพื่อแบ่งที่การจัดเก็บสินค้าแต่ละประเภท (Color) 43
3.1.1.2 ข้อมูลกลุ่มที่2 43
(1) กำหนดพื้นที่ที่ควรอยู่ใกล้กัน (Adjacent Space) 43
(2) กำหนดพื้นที่ต้องอยู่ไกลกัน (Nonadjacent Space) 43
3.1.2 ใช้ Dynamo ในขั้นตอนการสร้างพื้นที่ (Create Space) และขั้นตอนการวางผัง (Space Planning) 43
3.1.2.1 แนวความคิดในขั้นตอนการสร้างพื้นที่ (Create Space) 43

(1) หลักการหาขนาดของพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) 43

(2) หลักการหาขนาดของขอบเขตอาคาร (Building Space) 46

(3) หลักการหาขนาดของพื้นที่ในการบริหารจัดการการลานโหลดสินค้า (Loading Area) 46

3.1.2.2 แนวความคิดในขั้นตอนการวางผัง (Space Planning) 47

(1) ลำดับของการจัดเรียง (Permutation) 47

(2) กระบวนการจัดเรียงพื้นที่ (Strategy) 48

3.1.3 ใช้ Refinery ในการสร้างและกำหนดผลลัพธ์การวางผังคลังสินค้าที่หลากหลาย 50

3.1.3.1 Input ของโปรแกรมวางผังคลังสินค้า 50

(1) Input 1 ค่า Input ที่มีผลต่อพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) 50

(2) Input 2 ค่าInputที่มีผลต่อการจัดเรียงหรือการวางผังคลังสินค้าในแง่ของลำดับของการจัดเรียง (Permutation) 51

(3) Input 3 ค่าInputที่มีผลต่อการจัดเรียงหรือการวางผังคลังสินค้าในแง่ของกระบวนการจัดเรียงพื้นที่ (Strategy) 51

3.1.3.2 Output ของโปรแกรมวางผังคลังสินค้า 51

(1) Output 1 ค่า Output ที่มีผลต่อการเลือกพิจารณาการใช้งานที่เหมาะสมสอดคล้องกับการทำงานของผู้ใช้งาน 52

(2) Output 2 ค่า Output ที่มีผลต่อการเลือกพิจารณาความปลอดภัยในการจัดเก็บวัตถุดิบแต่ละประเภท 52

(3) Output 3 ค่า Output ที่มีผลต่อการเลือกพิจารณาปริมาณการจัดเก็บที่เพียงพอ 52

(4) Output 4 ค่า Output ที่มีผลต่อการเลือกพิจารณาขนาดอาคาร 53

3.2 การออกแบบและขั้นตอนการทำงานของโปรแกรมวางผังคลังสินค้า (Warehouse Planning Program) 54

3.2.1 โปรแกรมวางผังคลังสินค้าส่วน Excel 55
3.2.1.1 ประเภทที่ 1 จำกัดจำนวนข้อมูลในการกรอก 55
(1) พื้นที่ขอบเขตที่ดิน (Area) 55
(2) จำนวนของรถชนิดต่างๆที่ถูกใช้ในการขนส่ง (No. of Each Car Type) 56
3.2.1.2 ประเภทที่ 2 ข้อมูลที่ไม่จำกัดปริมาณการกรอก 56
(1) รายชื่อพื้นที่จัดเก็บสินค้าแต่ละประเภท (Name Space) 56
(2) จำนวนความต้องการจัดเก็บสินค้าในแต่ละประเภท (Capacity) .. 56
(3) จำนวนการซ้อนชั้นของการจัดเก็บสินค้าแต่ละประเภท (Level) .. 56
(4) กำหนดสีเพื่อแบ่งที่การจัดเก็บสินค้าแต่ละประเภท (Colour) 56
3.2.1.3 ประเภทที่ 3 ข้อมูลที่กรอกในแต่ละ Column จะมีมากกว่า 1 ค่า 56
(1) พื้นที่ที่ควรอยู่ใกล้กัน (Adjacent Space) 56
(2) พื้นที่ต้องอยู่ไกลกัน (Nonadjacent Space) 56
3.2.2 โปรแกรมวางผังคลังสินค้าส่วน Dynamo 57
3.2.2.1 นำเข้าข้อมูล (Import Information) 59
(1) ส่วนที่ 1 Import Excel into program 59
(2) ส่วนที่ 2 Build Color From Color Values 63
(3) ส่วนที่ 3 Create Column Values 68
3.2.2.2 สลับสับเปลี่ยนข้อมูล (Shuffle Information) 71
3.2.2.3 สร้างพื้นที่จากข้อมูลขั้นตอนที่ 2 (Create Space) 75
(1) กลุ่มพื้นที่ 1 Create Loading Area 76
(2) กลุ่มพื้นที่ 2 Create Space Area 80
(3) กลุ่มพื้นที่ 3 Create Site Area 84
3.2.2.4 จัดเรียงวางผังพื้นที่จากขั้นตอนที่ 3 (Planning) 87

3.2.2.5 ตรวจสอบความถูกต้องของผังที่ถูกจัดเรียงในขั้นตอนที่ 4 (Recheck) 89
(1) ส่วนที่ 1 Recheck OSR (Open Space Ratio) 90
(2) ส่วนที่ 2 Recheck number of space 92
3.2.2.6 ระบุข้อมูลจำเพาะให้แสดงผลลงบนผังที่ถูกจัดเรียงในขั้นตอนที่ 4 (Identification) 94
(1) การแสดงผลด้วยสีและเส้น 95
(2) การแสดงผลด้วยข้อความ 96
3.2.2.7 สร้างเงื่อนไขเพื่อให้เกิดค่าของข้อมูลในแต่ละครั้งของผลลัพธ์ในการวางผังที่เกิดขึ้น (Set Condition) 99
(1) Adjacent 99
(2) Nonadjacent 104
3.2.2.7 บันทึกค่า Input ของผลลัพธ์ที่ผู้ใช้งานพึงพอใจ (Export Favorite Input) 110
3.2.3 โปรแกรมวางผังคลังสินค้าส่วน Refinery 112
3.2.3.1 ข้อมูลเข้า (IS INPUT) ในโปรแกรมวางผังคลังสินค้า 112
(1) Input 1 ช่วงเปอร์เซ็นความกว้าง Column (100%-50%) 113
(2) Input 2 ช่วง Probability (วิธีที่ 1- วิธีที่ n!) 113
(3) Input 3 ช่วง Strategy (รูปแบบที่1 - รูปแบบที่3) 114
3.2.3.2 ข้อมูลออก (IS OUTPUT) ในโปรแกรมวางผังคลังสินค้า 115
(1) Output 1 ค่าผลลัพธ์การใกล้กันของพื้นที่ (Adjacent) 115
(2) Output2 ค่าผลลัพธ์การไม่ควรใกล้กันของพื้นที่ (Nonadjacent) 116
(3) Output 3 ค่าผลลัพธ์ปริมาณการจัดเก็บ (Capacity) 117
(4) Output 4 ค่าผลลัพธ์ขนาดอาคาร (Building Area) 118
(5) Output 5 ค่าผลลัพธ์ผลรวมจำนวนพื้นที่ (COUNT SPACE) 118
บทที่ 4 ผลการวิจัยและอภิปรายผล 121

4.1 โปรแกรมวางผังคลังสินค้าและวิธีการใช้งาน 121
4.1.1 โปรแกรมวางผังคลังสินค้าส่วน Excel 121
4.1.1.1 วิธีการใช้งานโปรแกรมวางผังคลังสินค้าส่วน Excel 122
(1) Area 122
(2) Car Type 122
(3) Number Of Car Type 123
(4) Space ID 123
(5) Space Name 124
(6) Capacity 124
ในหัวข้อนี้เป็นการกรอกจำนวนการจัดเก็บที่ต้องการในแต่ละพื้นที่โดยมีหน่วย
เป็น Pallet 124
(7) Rack Level 124
(8) Colour 125
(9) Adjacent Spaces 125
(9) Nonadjacent Spaces 125
4.1.2 โปรแกรมวางผังคลังสินค้าส่วน Dynamo 126
4.1.2.1 วิธีการใช้งานโปรแกรมวางผังคลังสินค้าส่วน Dynamo 126
(1) เลือก ไฟล์Excel ข้อมูลความต้องการของลูกค้า 127
(2) ทดลองใส่ค่า Input ที่ต้องการ 127
(3) เริ่มการประมวลผล 131
(4) เลือกบันทึกค่า Input ที่ได้ผลลัพธ์ตรงตามความต้องการ 133
4.1.3 โปรแกรมวางผังคลังสินค้าส่วน Refinery 134
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ 138
5.1 สรุปผลการวิจัยโปรแกรมวางผังคลังสินค้าจากการนำไปใช้งานจริง 138

5.1.1 สรุปผลโปรแกรมวางผังคลังสินค้ากับปัญหาความซับซ้อนของการวางผังคลังสินค้า .. 141
5.1.1.1 สรุปผลขั้นตอนการทำงานด้วยโปรแกรมวางผังคลังสินค้า 141
5.1.1.2 สรุปผลความถูกต้องของผลลัพธ์จากโปรแกรมวางผังคลังสินค้า 142
5.1.2 สรุปผลโปรแกรมวางผังคลังสินค้ากับเวลา 143
รายการอ้างอิง 145
ประวัติผู้เขียน 146

สารบัญภาพ

หน้า

ภาพที่ 1 ตัวอย่างคลังสินค้าแบบ Conventional Rack 6
ภาพที่ 2 ตัวอย่างคลังสินค้าแบบ Automated 7
ภาพที่ 3 แผนภาพสรุปขั้นตอนการวางผังอาคารคลังสินค้าโดยสถาปนิก 8
ภาพที่ 4 แผนภาพความเชื่อมโยงของประเภทคลังสินค้า 9
ภาพที่ 5 รูปแบบบรรจุภัณฑ์ของคลังสินค้าประเภทสินค้าสำเร็จ 9
ภาพที่ 6 แผนภาพสรุปการจัดกลุ่มวัตถุดิบ 11
ภาพที่ 7 ภาพแสดงการจัดกลุ่มระหว่างวัตถุที่จัดเก็บกับอุณหภูมิ 12
ภาพที่ 8 ประเภทสารเคมีที่เป็นวัตถุดิบ 13
ภาพที่ 9 ภาพแสดงการจัดกลุ่มประเภทสารเคมี 14
ภาพที่ 10 ภาพแสดงการจัดกลุ่มอุณหภูมิสินค้ากับประเภทสารเคมีเพื่อกำหนดพื้นที่ 15
ภาพที่ 11 ภาพแสดงกลุ่มพื้นที่ที่เกิดจากการจัดกลุ่มอุณหภูมิสินค้ากับประเภทสารเคมี 15
ภาพที่ 12 แผนภาพแสดงขนาดที่แตกต่างกันของแต่ละกลุ่มพื้นที่ 16
ภาพที่ 13 แผนภาพสรุปความสัมพันธ์ของประเภทพื้นที่คลังสินค้า 17
ภาพที่ 14 ตัวอย่างการวางสินค้าแบบ On floor 18
ภาพที่ 15 ตัวอย่างการจัดวางสินค้า On floor โดย Cargo Optimizer Enterprise เป็นโปรแกรมที่ใช้คำนวณการจัดเรียงสินค้า 19
ภาพที่ 16 ตัวอย่างการวางสินค้าแบบ On rack 19
ภาพที่ 17 รูปด้านบนเป็นแผนภาพของวิธีการขนส่ง (Method of Transportation) 21
ภาพที่ 18 รถประเภทต่างๆที่ใช้ในการขนส่ง (kargo, 2017) 22
ภาพที่ 19 ภาพตัวอย่าง Pallet ของบริษัท Roscam ที่นิยมใช้ในประเทศไทย (Loscam, 2020) ... 23
ภาพที่ 20 ตัวอย่างระยะร่นตามกฎหมายของอาคารประเภทคลังสินค้า ภาพจาก TERRABKK 23

ภาพที่ 21 ภาพแสดงองค์ประกอบของชั้นวางสินค้า (macrack, 2017) 24
ภาพที่ 22 ตัวอย่าง Selective Rack จาก www.lpi.co.th 25
ภาพที่ 23 ภาพแสดงทางสัญจรของรถยก 25
ภาพที่ 24 ภาพแสดงรูปแบบการโหลดสินค้า Back Loading (แดง) และ Side Loading (น้ำเงิน) . 26
ภาพที่ 25 ตัวอย่างการจัดเรียงผังอาคารคลังสินค้าของโรงงานเครื่องดื่มชูกำลังแห่งหนึ่ง 28
ภาพที่ 26 ตัวอย่างการผลลัพธ์ของผังอาคารคลังสินค้าของโรงงานเครื่องดื่มชูกำลังแห่งหนึ่ง 28
ภาพที่ 27 ภาพแสดงแผนงานช่วง Conceptual Desing ในเวลา 3 สัปดาห์ 31
ภาพที่ 28 ขั้นตอนการทำงานแบบเดิมกับเวลา 32
ภาพที่ 29 รูปตัวอย่างNode Rectangle.ByWidthLength เป็นNodeในการสร้างสี่เหลี่ยม มีInput คือ ความกว้างและความยาว ส่วนOutput จะได้สี่เหลี่ยมออกมา 33
ภาพที่ 30 แผนภาพแสดงการทำงานส่วนที่ 1, 2, 5 ที่ยังคงทำโดยสถาปนิกอยู่ 34
ภาพที่ 31 แผนภาพแสดงการทำงานส่วนที่ 3, 4 ที่ใช้โปรแกรมมาทดแทนการทำงานแบบเดิม 34
ภาพที่ 32 แผนภาพแสดงการทำงานส่วนที่ 3, 4 ที่ใช้โปรแกรม Dynamo และ Refinery มาทดแทนการทำงานแบบเดิม 35
ภาพที่ 33 ภาพแสดงการวางผังคลังสินค้าโดยหลักการ Container (Das, 2016) 36
ภาพที่ 34 ภาพแสดงการเลือก Node ที่เป็น Is Input 37
ภาพที่ 35 ภาพแสดงการเลือก Node ที่เป็น Is Output 37
ภาพที่ 36 ภาพแสดงผลลัพธ์ของ Input และ Output ในโปรแกรม Refinery โดยผู้ใช้สามารถกำหนดผลลัพธ์ที่อยากให้เป็นได้ 38
ภาพที่ 37 ภาพแสดงลูกศรเส้นประแสดงถึงค่าของผลลัพธ์ที่จะได้จาก Refinery 39
ภาพที่ 38 ตัวอย่างแผนภูมิค่าความสัมพันธ์ของ Input และ Output ที่ออกมาอย่างหลากหลาย ... 40
ภาพที่ 39 ตัวอย่างผลลัพธ์ที่ออกมาอย่างหลากหลายโดย Refinery 40
ภาพที่ 40 แผนภาพแสดงแนวความคิดโปรแกรมวางผังคลังสินค้า 42
ภาพที่ 41 ภาพแสดงตำแหน่ง Row และ Column ของชั้นวางสินค้าในแปลน (Mecalux, 2018) . 44
ภาพที่ 42 ภาพแสดงลำดับการจัดเรียง (Permutation) โดยสี่เหลี่ยมสีต่างๆ 48

ภาพที่ 43 ภาพการวางผังแบบ Short Side Strategy (MATTERLAB, 2020) 48
ภาพที่ 44 ภาพการวางผังแบบ Long Side Strategy (MATTERLAB, 2020) 49
ภาพที่ 45 ภาพการวางผังแบบ Area strategy (MATTERLAB, 2020) 49
ภาพที่ 46 แผนภาพการออกแบบขั้นตอนการทำงานของโปรแกรมวางผังคลังสินค้า 54
ภาพที่ 47 ภาพแสดงโปรแกรมวางผังคลังสินค้าส่วน Excel 55
ภาพที่ 48 ภาพแสดงข้อมูลประเภทที่ 1 จำกัดจำนวนข้อมูลในการกรอก (สีเหลือง) 55
ภาพที่ 49 ภาพแสดงข้อมูลประเภทที่ 2 ข้อมูลที่ไม่จำกัดปริมาณการกรอก (สีเหลือง) 56
ภาพที่ 50 ภาพแสดงข้อมูลประเภทที่ 3 ข้อมูลที่กรอกในแต่ละ Column จะมีมากกว่า 1 ค่า (สีเหลือง) 57
ภาพที่ 51 ภาพแสดงโปรแกรมวางผังคลังสินค้าส่วน Dynamo 58
ภาพที่ 52 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Import Information 59
ภาพที่ 53 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel into program 59
ภาพที่ 54 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel into program หมายเลข 1 60
ภาพที่ 55 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel into program หมายเลข 2 60
ภาพที่ 56 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel into program หมายเลข 3 61
ภาพที่ 57 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 1 Import Excel into program หมายเลข 3 61
ภาพที่ 58 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel Into program หมายเลข 4 62
ภาพที่ 59 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 1 Import Excel Into program หมายเลข 4 62
ภาพที่ 60 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values 63
ภาพที่ 61 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 1 63
ภาพที่ 62 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 1 63
ภาพที่ 63 ตัวอย่างข้อมูลที่ถูกขั้นด้วย Comma (,) 64
ภาพที่ 64 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 2 64

ภาพที่ 65 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 2 65
ภาพที่ 66 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 3 65
ภาพที่ 67 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 3 66
ภาพที่ 68 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 4 66
ภาพที่ 69 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 5 67
ภาพที่ 70 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 6 67
ภาพที่ 71 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values 68
ภาพที่ 72 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 1 68
ภาพที่ 73 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 2 69
ภาพที่ 74 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 3 69
ภาพที่ 75 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 4 70
ภาพที่ 76 ภาพแสดง Is Input ของโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 4 70
ภาพที่ 77 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 5 71
ภาพที่ 78 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Shuffle Information 71
ภาพที่ 79 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 1 72
ภาพที่ 80 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 2 72
ภาพที่ 81 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 3 72
ภาพที่ 82 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 4 73
ภาพที่ 83 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 5 73
ภาพที่ 84 Input เพื่อควบคุมค่าให้เกิดความหลากหลายของผลลัพธ์ได้ 73
ภาพที่ 85 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 6 74
ภาพที่ 86 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 6 - 7 74

ภาพที่ 87 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Create Space 75
ภาพที่ 88 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area 76
ภาพที่ 89 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 1..... 76
ภาพที่ 90 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 2..... 77
ภาพที่ 91 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 3..... 78
ภาพที่ 92 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 4..... 78
ภาพที่ 93 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 5..... 79
ภาพที่ 94 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 6..... 79
ภาพที่ 95 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area..... 80
ภาพที่ 96 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 1..... 80
ภาพที่ 97 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 2..... 81
ภาพที่ 98 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 3..... 82
ภาพที่ 99 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 4..... 82
ภาพที่ 100 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 5..... 82
ภาพที่ 101 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 6..... 83
ภาพที่ 102 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 7..... 83
ภาพที่ 103 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area..... 84
ภาพที่ 104 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area หมายเลข 1..... 84
ภาพที่ 105 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area หมายเลข 2..... 85
ภาพที่ 106 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area หมายเลข 3..... 86
ภาพที่ 107 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area หมายเลข 4..... 86
ภาพที่ 108 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area ในส่วนกรอกค่าระยะร่น 86
ภาพที่ 109 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Planning..... 87
ภาพที่ 110 ภาพแสดงโปรแกรมส่วน Planning หมายเลข 1 88

ภาพที่ 111 Input เพื่อให้ผู้ใช้สามารถกำหนดค่าในส่วน Strategy ได้ 88
ภาพที่ 112 ภาพแสดงโปรแกรมส่วน Planning หมายเลข 2 88
ภาพที่ 113 ภาพแสดงโปรแกรมส่วน Planning หมายเลข 3 89
ภาพที่ 114 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Recheck 89
ภาพที่ 115 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR 90
ภาพที่ 116 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR หมายเลข 1 90
ภาพที่ 117 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR หมายเลข 2 91
ภาพที่ 118 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR หมายเลข 3 91
ภาพที่ 119 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR หมายเลข 4 91
ภาพที่ 120 ภาพแสดงโปรแกรมส่วนที่ 2 Recheck number of space 92
ภาพที่ 121 ภาพแสดงโปรแกรมส่วนที่ 2 Recheck number of space หมายเลข 1 92
ภาพที่ 122 ภาพแสดงโปรแกรมส่วนที่ 2 Recheck number of space หมายเลข 2 93
ภาพที่ 123 ภาพแสดงโปรแกรมส่วนที่ 2 Recheck number of space หมายเลข 2 - 4 93
ภาพที่ 124 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Identification 94
ภาพที่ 125 ภาพแสดงโปรแกรมส่วนที่ 1 การแสดงผลด้วยสีและเส้นหมายเลข 1 95
ภาพที่ 126 ภาพแสดงโปรแกรมส่วนที่ 1 การแสดงผลด้วยสีและเส้นหมายเลข 1-2 95
ภาพที่ 127 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความ 96
ภาพที่ 128 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 1 96
ภาพที่ 129 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 2 97
ภาพที่ 130 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 3 97
ภาพที่ 131 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 4 97
ภาพที่ 132 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 5 98
ภาพที่ 133 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 6 98
ภาพที่ 134 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Set Condition 99

ภาพที่ 135 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent 99
ภาพที่ 136 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent (ขยาย) 100
ภาพที่ 137 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent (ขยาย) 100
ภาพที่ 138 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 1 100
ภาพที่ 139 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 2 101
ภาพที่ 140 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 3 101
ภาพที่ 141 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 4 102
ภาพที่ 142 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 5 102
ภาพที่ 143 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 6 103
ภาพที่ 144 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 7 103
ภาพที่ 145 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 8 103
ภาพที่ 146 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent 104
ภาพที่ 147 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent (ขยาย) 104
ภาพที่ 148 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent (ขยาย) 104
ภาพที่ 149 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 1 105
ภาพที่ 150 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 2 105
ภาพที่ 151 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 3 106
ภาพที่ 152 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 4 106
ภาพที่ 153 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 5 106
ภาพที่ 154 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 6 107
ภาพที่ 155 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 7 107
ภาพที่ 156 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 8 108
ภาพที่ 157 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 9 108
ภาพที่ 158 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 10 108

ภาพที่ 159 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 11 109
ภาพที่ 160 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 12 109
ภาพที่ 161 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Export Favorite Input 110
ภาพที่ 162 ภาพแสดงโปรแกรมส่วน Export Favorite Input หมายเลข 1 111
ภาพที่ 163 ภาพแสดงโปรแกรมส่วน Export Favorite Input หมายเลข 2 111
ภาพที่ 164 ภาพแสดงโปรแกรมส่วน Export Favorite Input หมายเลข 3 112
ภาพที่ 165 ภาพแสดงโปรแกรมส่วน Export Favorite Input หมายเลข 4 112
ภาพที่ 166 ภาพแสดงโปรแกรมส่วน Input 1 ช่วงเปอร์เซ็นความกว้าง Column 113
ภาพที่ 167 ภาพแสดงโปรแกรมส่วน Input2 ช่วงprobability 113
ภาพที่ 168 ภาพแสดงโปรแกรมส่วน Input 3 ช่วง Strategy 114
ภาพที่ 169 ภาพแสดง Node ที่เป็น Input ทั้ง 3 Strategy 114
ภาพที่ 170 ภาพแสดงผลลัพธ์การใกล้กันของพื้นที่กลุ่ม 1 115
ภาพที่ 171 ภาพแสดงผลลัพธ์การใกล้กันของพื้นที่กลุ่ม 2 115
ภาพที่ 172 ภาพแสดงโปรแกรมส่วน Output2 ค่าผลลัพธ์การไม่ควรใกล้กันของพื้นที่ 116
ภาพที่ 173 ภาพแสดงผลลัพธ์การไม่ควรใกล้กันของพื้นที่กลุ่ม 1 116
ภาพที่ 174 ภาพแสดงผลลัพธ์การไม่ควรใกล้กันของพื้นที่กลุ่ม 2 117
ภาพที่ 175 ภาพแสดงโปรแกรมส่วน Output 3 ค่าผลลัพธ์ปริมาณการจัดเก็บ 117
ภาพที่ 176 ภาพแสดงโปรแกรมส่วน Output4 ค่าผลลัพธ์ขนาดอาคาร 118
ภาพที่ 177 ภาพแสดงโปรแกรมส่วน Output5 ค่าผลลัพธ์ผลรวมจำนวนพื้นที่ 119
ภาพที่ 178 ภาพแสดงผลรวมจำนวนพื้นที่กลุ่มที่ 1 119
ภาพที่ 179 ภาพแสดงผลรวมจำนวนพื้นที่กลุ่มที่ 2 120
ภาพที่ 180 ภาพแสดงขั้นตอนการทำงานด้วยโปรแกรมวางผังคลังสินค้า 121
ภาพที่ 181 ภาพแสดงตำแหน่ง Row และ Column ในโปรแกรมวางผังคลังสินค้าส่วน Excel 122
ภาพที่ 182 ภาพแสดงส่วนกรอกข้อมูล Area 122

ภาพที่ 183 ภาพแสดงส่วนกรอกข้อมูล Car Type 123
ภาพที่ 184 ภาพแสดงส่วนกรอกข้อมูล Number Of Car Type 123
ภาพที่ 185 ภาพแสดงส่วนกรอกข้อมูล Space ID 123
ภาพที่ 186 หน้าตาโปรแกรมวางผังคลังสินค้าส่วน Dynamo 126
ภาพที่ 187 ส่วน Input ซ้ายสุดบนโปรแกรมวางผังคลังสินค้าส่วน Dynamo 126
ภาพที่ 188 Node ที่ใช้เลือกไฟล์ 127
ภาพที่ 189 Node ที่ใช้ปรับค่า Input1 รูปทรงของพื้นที่ 127
ภาพที่ 190 ผลลัพธ์การวางผังคลังสินค้าที่หลากหลายจากการปรับค่าช่วงเปอร์เซ็นความกว้าง 128
ภาพที่ 191 Node ที่ใช้ปรับค่า Input2 ช่วง Probability 128
ภาพที่ 192 ผลลัพธ์การวางผังคลังสินค้าที่หลากหลายจากการปรับค่าลำดับของการจัดเรียง 129
ภาพที่ 193 Node ที่ใช้ปรับค่า Input3 ช่วง Strategy 129
ภาพที่ 194 ผลลัพธ์การวางผังคลังสินค้าที่หลากหลายจากการปรับค่า Strategy 130
ภาพที่ 195 Node ที่ใช้ปรับค่า Input4 ค่าSetback Values 130
ภาพที่ 196 ผลลัพธ์การวางผังคลังสินค้าที่หลากหลายจากการปรับค่า Setback 131
ภาพที่ 197 ปุ่ม Run ทางด้านล่างซ้ายของโปรแกรม 131
ภาพที่ 198 สัญลักษณ์มุมมองแสดงผลส่วนโมเดลสามมิติ 132
ภาพที่ 199 มุมมองแสดงผลส่วนโมเดลสามมิติ 132
ภาพที่ 200 สัญลักษณ์มุมมองแสดงผล Node 132
ภาพที่ 201 มุมมองแสดงผล Node 133
ภาพที่ 202 ปุ่มเลือกในการบันทึกค่า 133
ภาพที่ 203 โปรแกรมในส่วนบันทึกค่า Input ที่ได้ผลลัพธ์ตรงตามความต้องการ 134
ภาพที่ 204 การประมวลผล 4 วิธีคือ Optimize, Cross Product, Randomize, Like This 135
ภาพที่ 205 หน้าต่างเมื่อเลือกการประมวลผลวิธี Optimize สามารถกำหนดผลลัพธ์ที่ได้ 136
ภาพที่ 206 ส่วนบนของโปรแกรมแสดงค่า Input และ output ของผลลัพธ์ที่เกิดขึ้น 137

ภาพที่ 207 ส่วนล่างของโปรแกรมจะแสดงผลลัพธ์สามมิติของผังคลังสินค้าในรูปแบบที่หลากหลาย 137
ภาพที่ 208 ผลการใช้โปรแกรมวางผังคลังสินค้าส่วน Excel รับข้อมูลจากลูกค้า 139
ภาพที่ 209 ผลลัพธ์การใช้โปรแกรมวางผังคลังสินค้าส่วน Dynamo 139
ภาพที่ 210 ผลลัพธ์การใช้โปรแกรมวางผังคลังสินค้าส่วน Refinery 140
ภาพที่ 211 ผลลัพธ์ของแบบที่ถูกพัฒนามาจากโปรแกรมวางผังคลังสินค้า 140
ภาพที่ 212 เปรียบเทียบขั้นตอนการวางผังแบบเดิมกับการวางผังคลังสินค้าด้วยโปรแกรม 142
ภาพที่ 213 ผลลัพธ์ของผังคลังสินค้าที่หลากหลายและถูกต้องตามเงื่อนไขที่ผู้ใช้งานกำหนด 142
ภาพที่ 214 แผนภาพแสดงเวลากับขั้นตอนการทำงานวางผังคลังสินค้าแบบเดิม 143
ภาพที่ 215 แผนภาพแสดงเวลากับขั้นตอนการทำงานวางผังคลังสินค้าด้วยโปรแกรมวางผังคลังสินค้า 144

มหาวิทยาลัยศิลปากร

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

เนื่องจากงานผู้วิจัยเป็นสถาปนิกที่ทำงานเกี่ยวกับการออกแบบงานสถาปัตกรรมในกลุ่มอาคารประเภทโรงงาน ซึ่งประกอบไปด้วยหลากหลายอาคาร อาทิเช่น อาคารผลิต(Production building) อาคารคลังสินค้า(Warehouse) อาคารงานระบบ(Utility building) อาคารเศษซาก (Waste storage building) อาคารบำบัดน้ำ(WWTP building)เป็นต้น ซึ่งกระบวนการออกแบบแต่ละอาคารมีความซับซ้อนตามปัจจัยที่มีผลต่อการออกแบบ โดยเฉพาะอาคารคลังสินค้า(Warehouse) เนื่องจากความต้องการใช้งานภายในอาคารที่หลากหลาย โดยเป็นผลมากจากประเภทของวัตถุที่ต้องการจัดเก็บในคลังสินค้านั้นหลากหลายซึ่งวัตถุแต่ละชนิดมีเงื่อนไขในการจัดเก็บที่แตกต่างกัน กล่าวคือ ยิ่งประเภทของวัตถุที่จะจัดเก็บในคลังสินค้ามีความหลากหลายมากเท่าไหร่ ความซับซ้อนในการออกแบบอาคารจะมากขึ้นเท่านั้น

ขั้นตอนการออกแบบวางผัง เป็นขั้นตอนเบื้องต้นในช่วงที่เรียกว่า Conceptual design คือ ช่วงของการออกแบบงานสถาปัตยกรรมเบื้องต้น ถือว่าเป็นขั้นตอนที่สำคัญเนื่องจากจะเป็นตัวกำหนด การใช้งาน(Function) ตำแหน่ง(Location) ,ขอบเขต(boundary) ,พื้นที่(Area) และรูปร่าง หน้าตา(Shape Form) ของอาคารนั้นๆ กล่าวคือหากการวางผังไม่ดีตั้งแต่ต้น จะส่งผลต่อประสิทธภาพทุกด้านของอาคาร เช่น การใช้งานที่ไม่เหมาะสม ตำแหน่งห้องที่อาจเกิดปัญหา ขนาดห้องที่ไม่เพียงพอต่อการใช้งาน เป็นต้น

1.1.1 ปัญหาของการวิธีการวางผังอาคารคลังสินค้าในปัจจุบัน

จากขั้นตอนการวางผังอาคารคลังสินค้าที่อธิบายมาข้างต้นจะสังเกตุได้ว่ามีความซับซ้อนสูงโดยแปรผันตามความหลากหลายของประเภทของที่จัดเก็บ หากความซับซ้อนมากขั้นตอนการวางผังจะใช้เวลานานและต้องการความละเอียดของการวิเคราะการวางผังสูง ปัจจัย2สิ่งที่กระทบต่อวิธีการวางผังอาคารคลังสินค้าในปัจจุบันคือ

1.1.1.1 ความซับซ้อนของวัตถุที่จัดเก็บ

อาคารคลังสินค้าที่มีความหมากหลายของประเภทสินค้าสูงย่อมต้องการการวิเคราะการวางผังที่มีประสิทธิภาพสูงตาม จากวิธีการวางผังแบบเดิมเป็นน้ำกลุ่มของพื้นที่มาจัดเรียงเพื่อวางผังโดยสถาปนิกเอง ซึ่งมีโอกาสที่จะได้รูปแบบที่ไม่ตรงหรือผิดไปจากเงื่อนไขการ

จัดเก็บ ยิ่งประเภทวัตถุหลากหลาย ยิ่งยากต่อการวางผังตามเงื่อนไข และยิ่งเกิดความคลาดเคลื่อนและผิดพลาดได้ง่าย

1.1.1.2 เวลา

ขั้นตอนการออกแบบวางผัง เป็นขั้นตอนในช่วงที่เรียกว่า Conceptual design คือ ช่วงของการออกแบบงานสถาปัตยกรรมเบื้องต้น เป็นตัวกำหนด การใช้งาน(Function) ตำแหน่ง(Location) ,ขอบเขต(boundary) ,พื้นที่(Area) และรูปร่าง หน้าตา(Shape Form) ของอาคารนั้นๆ ซึ่งระยะเวลาการทำงานโดยทั่วไปในช่วงนี้มีประมาณ 3 อาทิตย์ ถึง 1เดือน โดยขอบเขตของการทำงาน(Scope of work)ในช่วงนี้นั่นไม่เพียงแต่ขั้นตอนของการวางผังที่ซับซ้อนเท่านั่นแต่ยังรวมถึงการจัดทำการนำเสนอต่อลูกค้า การทำภาพเสมือนจริงของอาคารในเบื้องต้น การแก้ไขความเปลี่ยนแปลงของการตัดสินใจของลูกค้าในระหว่างออกแบบ อีกด้วย ดังนั่นเวลาเพียง3อาทิตย์ไม่เพียงพอต่อการออกแบบวางผังอาคารคลังสินค้าที่ซับซ้อนให้ได้ผลลัพธ์ที่ดีต่อการใช้งานและถูกต้องตามเงื่อนไขการวางผังได้

1.1.2 แนวทางการแก้ปัญหา

ปัจจุบันมีวิธีการวางผังด้วยโปรแกรมคอมพิวเตอร์โดยใช้วิธีการการสร้างข้อมูล (รูปทรง, ตำแหน่ง, ช่องว่าง,ฯลฯ)จากนั้นนำมาสู่การประเมินผลภายใต้เงื่อนไขที่กำหนด เพื่อสร้างผลลัพธ์หรือทางออกทั้งหมดที่ตอบสนองความต้องการภายใต้เงื่อนไขที่กำหนด จากนั้นนำมาสู่การวิเคราะห์และทางเลือกของทางออกที่ดีที่สุด โดยวิธีการนี้เรียกว่า Generative Design โดยสามารถนำการวิธีนี้มาใช้ในการออกแบบโปรแกรมวางผังคลังสินค้าโดยสร้างเงื่อนไข ความสัมพันธ์ของข้อมูลต่างๆ เช่นประเภทของสินค้าวัสดุ, วิธีการจัดเก็บและข้อจำกัดด้านการจัดเก็บ จากนั้นซอฟต์แวร์จะสร้างรูปแบบผังด้วยวิธีการเรียงสับเปลี่ยนผลลัพธ์ที่เป็นไปได้ทั้งหมดภายใต้เงื่อนไขที่ระบุอย่างรวดเร็ว โดยที่ผลลัพธ์นั่นจะมีจำนวนมากหรือน้อยจะขึ้นอยู่ความแตกต่างของเงื่อนไขหรือปัจจัยที่กำหนดลงไปในแต่ละครั้งของการออกแบบ ซึ่งวิธีการดังกล่าวจะได้ผลลัพธ์ที่แม่นยำ ถูกต้องและควบคุมได้ โดยกระบวนทั้งหมดใช้เวลาน้อยมากเมื่อเทียบกับกระบวนการทำงานแบบปัจจุบัน

ทั้งหมดนี้จึงเป็นที่มาของการนำเสนอหัวข้อโปรแกรมการออกแบบวางผังอาคารคลังสินค้าเพื่อแก้ปัญหาการทำงานออกแบบในกระบวนการวางผังคลังสินค้าช่วง Conceptual design คือ ช่วงของการออกแบบงานสถาปัตยกรรมเบื้องต้น

1.2 วัตถุประสงค์ของการวิจัย

โปรแกรมช่วยในการวางผังคลังสินค้าสามารถคำนวนและสร้างขนาดพื้นที่ห้อง ของแต่ละการใช้งานในอาคารคลังสินค้ารวมทั้งพื้นที่ตั้งอาคารหรือขอบเขตที่ดินได้โดยคำนวนจากเพียงข้อมูลดิบที่ได้รับมาจากลูกค้าหรือข้อมูลดิบที่เกิดจากการวิเคราะห์ของผู้ออกแบบเอง และสามารถการจัดเรียงพื้นที่คลังสินค้าโดยเกิดผลลัพธ์การวางผังคลังสินค้าที่หลากหลายตามเงื่อนไขของวัตถุที่จัดเก็บในคลังสินค้านั่นๆ โดยผลลัพธ์ที่หลากหลายจะถูกประเมินและให้คะแนนเพื่อช่วยในการตัดสินใจเลือกรูปแบบการวางผังที่สอดคล้องกับความต้องการของลูกค้าและผู้ออกแบบได้

1.3 ประโยชน์ของการวิจัย

1) สถาปนิกสามารถนำโปรแกรมการออกแบบวางผังคลังสินค้า (Warehouse planning program) ไปใช้ช่วยวางผังในช่วงของการออกแบบงานสถาปัตยกรรมเบื้องต้น โดยได้ผลลัพธ์ที่มีโอกาสเกิดขึ้นภายใต้เงื่อนไขที่กำหนดอย่างถูกต้องและควบคุมได้ในเวลาที่รวดเร็วและลดข้อผิดพลาดที่เกิดจากมนุษย์ ที่อาจคำนึงถึงเงื่อนไข ไม่ครบถ้วน หรือตกหล่น

2) ลดระยะเวลาการทำงานในการวางผังคลังสินค้า ซึ่งเป็นผลดีช่วยเพิ่มเวลาให้กับการทำงานอย่างอื่นที่เกี่ยวข้องในช่วง Conceptual design อย่างเช่น การจัดทำการนำเสนอต่อลูกค้า การทำภาพเสมือนจริงของอาคารในเบื้องต้น การแก้ไขความเปลี่ยนแปลงของการตัดสินใจของลูกค้าในระหว่างออกแบบ เป็นต้น

3) เจ้าของโครงการได้ใช้งานอาคารคลังสินค้าที่มีประสิทธิภาพและตรงกับการจัดเก็บสินค้าที่ถูกต้อง

4) ลดปัญหาความไม่ปลอดภัยที่จะเกิดขึ้นต่อคน ทรัพย์สิน และอาคาร จากการวางผังการจัดเก็บสินค้าที่อันตรายอย่างไม่ถูกต้อง

1.4 ขอบเขตของการวิจัย

ศึกษาวิธีการออกแบบอาคารคลังสินค้าระบบการขนส่งแบบใช้แรงงานคนร่วมกับเครื่องมือ (Conventional Rack) ในบริษัทออกแบบและให้คำปรึกษาโรงงานอุตสาหกรรมของตัวผู้วิจัยซึ่งทำหน้าที่เป็นสถาปนิกทำงานอยู่ เพื่อนำมาสรุปผลเป็นรายการและการออกแบบโปรแกรมที่ช่วยในการวางผังที่สอดคล้องกับปัจจัยและเงื่อนไขในการวางผังอาคารสินค้าในระยะเวลาที่จำกัดช่วง

Conceptual design คือ ช่วงของการออกแบบงานสถาปัตยกรรมเบื้องต้น โดยได้ผลลัพธ์ที่มีโอกาสเกิดขึ้นภายใต้เงื่อนไขที่กำหนด

1.5 นิยามศัพท์เฉพาะ

Generative design = นักออกแบบหรือวิศวกรป้อนข้อมูลเป้าหมายการออกแบบลงในซอฟต์แวร์การออกแบบพร้อมกับพารามิเตอร์ เงื่อนไข หรือ ปัจจัย เช่นประเภทของสินค้าวัสดุ วิธีการจัดเก็บและข้อจำกัดด้านการจัดเก็ซอฟต์แวร์จะสร้างวิธีการเรียงสับเปลี่ยนผลลัพธ์ที่เป็นไปได้ทั้งหมด สร้างทางเลือกในการออกแบบอย่างรวดเร็ว

WWTP = Waste Water Treatment Plant อาคารบำบัดน้ำที่ใช้ในการผลิต

Conceptual design = ช่วงของการออกแบบงานสถาปัตยกรรมเบื้องต้น เป็นขั้นตอนที่สำคัญเนื่องจากจะเป็นตัวกำหนด การใช้งาน(Function) ตำแหน่ง(Location) ,ขอบเขต (boundary) ,พื้นที่(Area) และรูปร่าง หน้าตา(Shape Form) ของอาคารนั้นๆ

Conventional Rack = รูปแบบการจัดเก็บสินค้าบนชั้นวางโดยใช้แรงงานคนในการจัดเก็บสินค้าร่วมกับอุปกรณ์เสริม เช่น รถยกชนิดต่างๆ

1.6 ขั้นตอนและวิธีการศึกษา

ชั้นตอนที่ 1 ศึกษามาตรฐานการจัดเก็บรวมทั้งเงื่อนไขและปัจจัยในการจัดเก็บของสินค้าละประเภทจากเอกสารมาตรฐานการจัดเก็บและวิธีการออกแบบวางผังจากการทำงานจริงของสถาปนิก

ชั้นตอนที่ 2 ศึกษาDynamo script ที่เกี่ยวกับการวิเคราะห์พื้นที่ และการวางผังอาคารด้วยแนวทางการออกแบบโปรแกรมแบบที่เรียกว่า Generative Design คือการป้อนข้อมูลของเป้าหมายการออกแบบลงในซอฟต์แวร์การออกแบบพร้อมกับพารามิเตอร์ เงื่อนไข หรือ ปัจจัย เช่น ประเภทของสินค้าวัสดุ วิธีการจัดเก็บและข้อจำกัดด้านการจัดเก็บ จากนั้นซอฟต์แวร์จะสร้างวิธีการเรียงสับเปลี่ยนผลลัพธ์ที่เป็นไปได้ทั้งหมดภายใต้เงื่อนไขที่ระบุพร้อมสร้างทางเลือกในการออกแบบอย่างรวดเร็ว

ชั้นตอนที่ 3 กำหนดเงื่อนไขและปัจจัยในการออกแบบวางผังคลังสินค้า

ชั้นตอนที่ 4 สร้างโปรแกรมในDynamo ให้สอดคล้องกับเงื่อนไขและวิธีการในการออกแบบวางผังคลังสินค้าที่ทำการศึกษามา

ขั้นตอนที่ 5 ทดลองโปรแกรมกับเงื่อนไขที่หลากหลายจากการใช้งานจริงของสถาปนิกที่ทำการออกแบบวางผังคลังสินค้า

ขั้นตอนที่ 6 รวบผลและปรับปรุงโปรแกรม

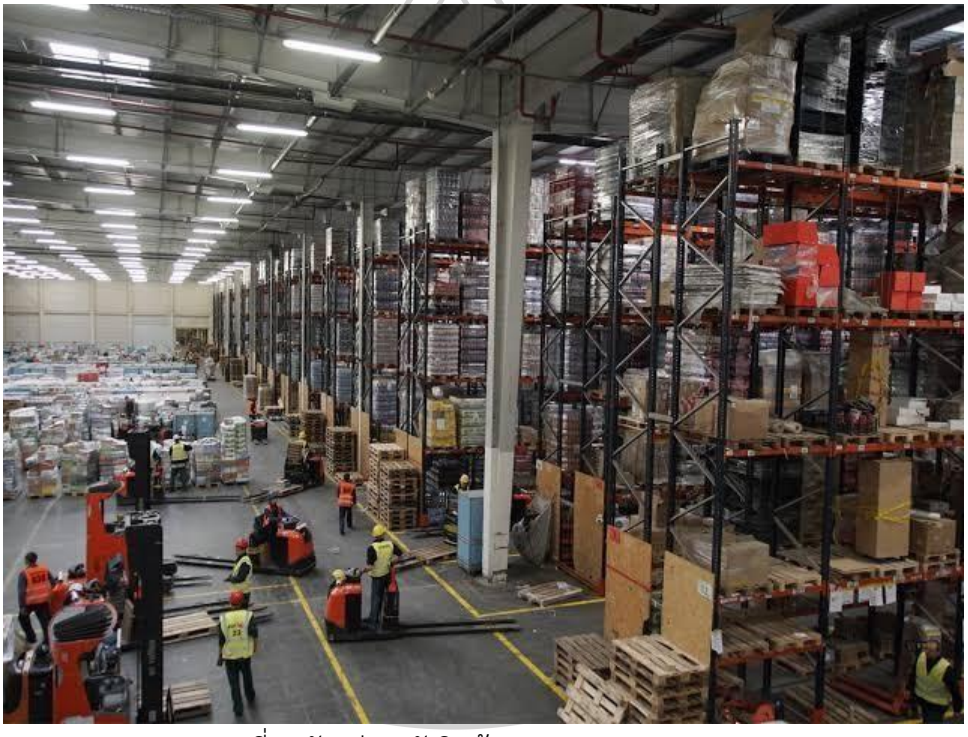
ขั้นตอนที่ 7 สรุปผลและพัฒนาโปรแกรมให้สมบูรณ์

บทที่ 2
วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

2.1 ขั้นตอนการวางผังคลังสินค้าในปัจจุบัน

ในปัจจุบันระบบการจัดเก็บสินค้าในคลัง(Warehouse Systems) กล่าวได้ว่ามี 2 รูปแบบใหญ่ๆคือ

รูปแบบที่1 ระบบการขนส่งแบบใช้แรงงานคนร่วมกับเครื่องมือ (Conventional Rack)



ภาพที่ 1 ตัวอย่างคลังสินค้าแบบ Conventional Rack

เป็นระบบคลังสินค้าที่ใช้กันโดยทั่วไปคือใช้แรงงานคนในการจัดเก็บหรือขนถ่ายสินค้าบน ชั้นวางสินค้า(Rack) โดยมีเครื่องมือในการขนถ่ายคือ รถโฟล์คลิฟไฟฟ้าประเภทต่างๆ ตามเงื่อนไขในการจัดเก็บ เพื่อจัดเก็บเข้าคลังสินค้า และ ออกจากคลังเพื่อขนส่งขึ้นรถขนส่งเพื่อกระจายสินค้าต่อไป

รูปแบบที่2 ระบบคลังสินค้าแบบอัตโนมัติ (Automated Storage/Retrieval System)

ภาพที่ 2 ตัวอย่างคลังสินค้าแบบ Automated

เป็นวิธีการขนส่งที่ใช้หุ่นยนต์และระบบคอมพิวเตอร์เข้ามามีหน้าที่ในการนำสินค้าจัดเก็บเข้าคลังสินค้า และ นำสินค้าออกจากคลังเพื่อโหลดขึ้นรถขนส่งสินค้าแบบอัตโนมัติโดยควบคุมจากระบบคอมพิวเตอร์ซึ่งโดยระบบคลังสินค้าประเภทนี้จะใช้คนน้อยลง หรือ ไม่มีเลย รวมถึงพื้นที่ในการจัดการจะน้อยลงไปอย่างมาก ซึ่งการจัดวางผังคลังสินค้าชนิดนี้จะไม่ซับซ้อนและใช้เวลาน้อยลงจากเดิมเป็นอย่างมากเนื่องจาก สถาปนิกจะทำงานร่วมกับผู้ผลิตเครื่องจักรASRS(Automated Storage/Retrieval System) โดยขั้นตอนในการวางผังส่วนการจัดเก็บภายในคลังสินค้าทั้งหมดจะเป็นหน้าที่ของผู้ผลิต ส่วนงานอาคารและส่วนที่ยังใช้แรงงานคน อย่างเช่นส่วนออฟฟิศ(ถ้ามี) จะเป็นของสถาปนิก

ดังนั้นจะเห็นว่าระบบการจัดเก็บสินค้าที่มีผลต่อการวางผังคลังโดยสถาปนิก ในแง่ของขั้นตอนและความซับซ้อนนั่น จะเป็นระบบแรกคือระบบการขนส่งแบบใช้แรงงานคนร่วมกับเครื่องมือ (Conventional Rack) โดยรายละเอียดยรูปแบบการขั้นตอนการวางผังอาคารคลังสินค้าในปัจจุบันที่จะกล่างถึงในหัวข้อถัดไป จะเป็นรายละเอียดขั้นตอนการจัดวางผังคลังสินค้าในปัจจุบันโดยสถาปนิกที่อยู่ภายใต้รูปแบบของระบบคลังสินค้าแบบ Conventional Rack

โดยขั้นตอนการวางผังอาคารคลังสินค้าโดยสถาปนิกมีดังนี้

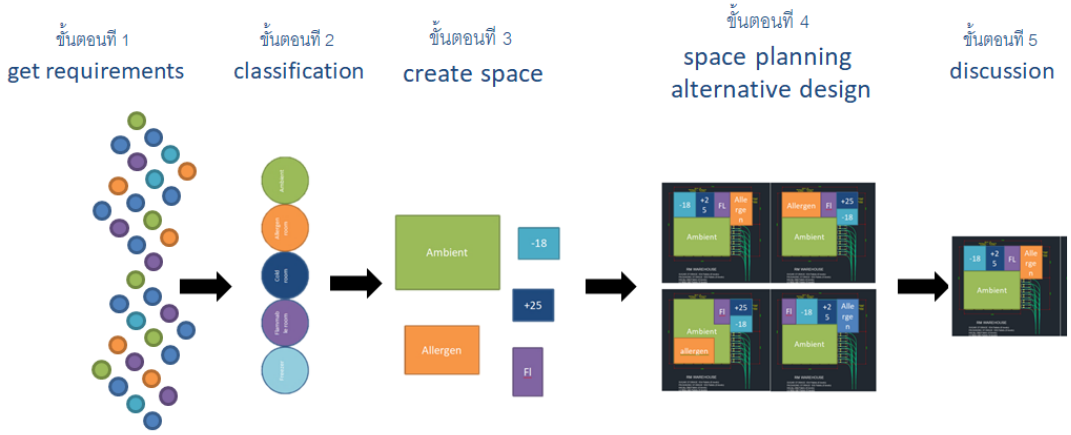
ขั้นตอนที 1 การรับข้อมูลความต้องการที่เกี่ยวข้องกับการออกแบบและวางผังคลังสินค้า (get requirements)

ขั้นตอนที 2 การจัดกลุ่มวัตถุที่จัดเก็บ (classification)

ขั้นตอนที 3 การหาขนาดของแต่ละกลุ่มพื้นที่ (create space)

ขั้นตอนที 4 การจัดเรียงพื้นที่คลังสินค้าโดยมีผลลัพธ์การวางผังคลังสินค้าที่หลากหลายตามเงื่อนไขของวัตถุที่จัดเก็บ (space planning and alternative design)

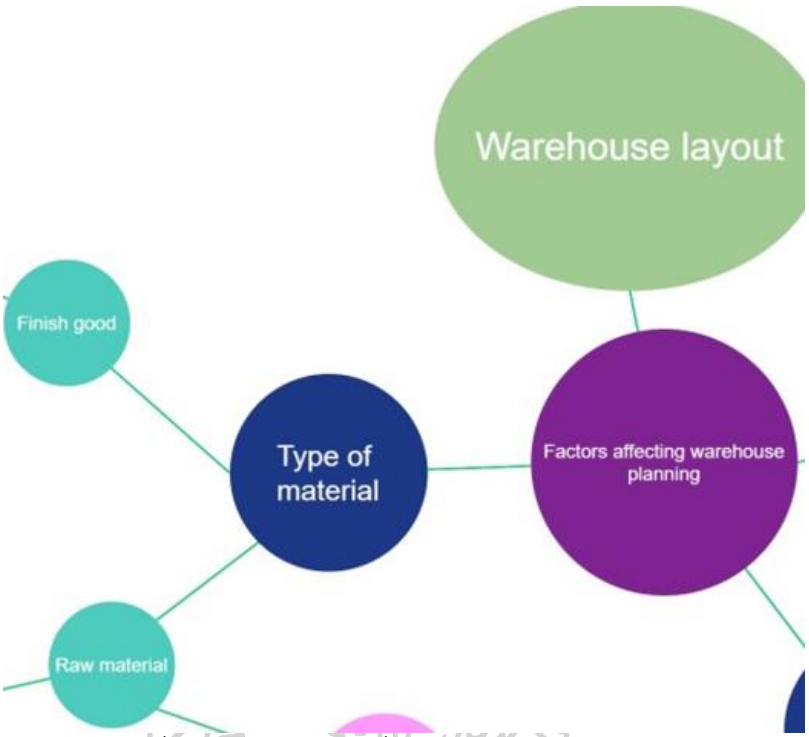
ขั้นตอนที 5 การพิจารณาแบบที่เหมาะสมที่สุดกับการทำงานของโรงงานนั่นๆ (discussion)



ภาพที 3 แผนภาพสรุปขั้นตอนการวางผังอาคารคลังสินค้าโดยสถาปนิก

2.1.1 ขั้นตอนที่1 การรับข้อมูลและความต้องการที่เกี่ยวข้องกับการออกแบบและวางผังคลังสินค้า (Get requirements)

ข้อมูลและความต้องการที่เกี่ยวข้องกับการออกแบบและวางผังคลังสินค้าจะได้จากเจ้าของคลังสินค้าหรือผู้ใช้งาน ซึ่งข้อมูลขั้นต้นที่มีผลต่อการออกแบบที่ต้องได้รับคือ ประเภทของคลังสินค้าและสินค้าหรือวัตถุที่จะจัดเก็บภายในคลังสินค้า ซึ่งประเภทของคลังสินค้าจำแนกตามวิธีการจัดเก็บนั่นจะถูกแบ่งเป็น 2 ประเภทหลักๆ คือ คลังสินค้าประเภทสินค้าสำเร็จ (Finish Good Warehouse) และ คลังสินค้าประเภทวัตถุดิบ (Raw Material Warehouse)



ภาพที่ 4 แผนภาพความเชื่อมโยงของประเภทคลังสินค้า

2.1.1.1 คลังสินค้าประเภทสินค้าสำเร็จ (Finish Good Warehouse)

ภาพที่ 5 รูปแบบบรรจุภัณฑ์ของคลังสินค้าประเภทสินค้าสำเร็จ

ประเภทนี้คือสินค้าที่ถูกผลิตออกมาจากอาคารผลิต(Production building) ส่วนใหญ่แล้วจะถูกส่งมาเก็บที่คลังสินค้าในรูปแบบของสินค้าที่อยู่ในบรรจุภัณฑ์เรียบร้อยแล้ว โดยกระบวนการบรรจุภัณฑ์สินค้าคือกระบวนการที่สินค้าที่ผลิตถูกบรรจุเรียงในลังสินค้า

(Packaging and Unitization) โดยเรียกกระบวนการนี้ว่าการรวมหน่วยร่วมย่อยจากนั้น ลังสินค้าแต่ละลังจะถูกนำมาเรียงบนพาเลท (Pallet) โดยเรียกกระบวนการนี้ว่าการรวมหน่วยร่วมย่อยหลายๆ หน่วยให้เป็นหน่วยร่วมใหญ่ (Master carton) จากนั้นจะทำการนำมาจัดเก็บที่คลังสินค้าก่อนที่จะจัดส่งโดยวิธีการขนส่งต่างๆตามแต่ประเภทสินค้า

ดังนั้นประเภทของสินค้าที่จะจัดเก็บจึงไม่หลากหลายเพราะมีการบรรจุภัณฑ์เรียบร้อยแล้วทำให้การออกแบบวางผังนั่นทำได้ง่ายกว่าคลังสินค้าประเภทอื่น

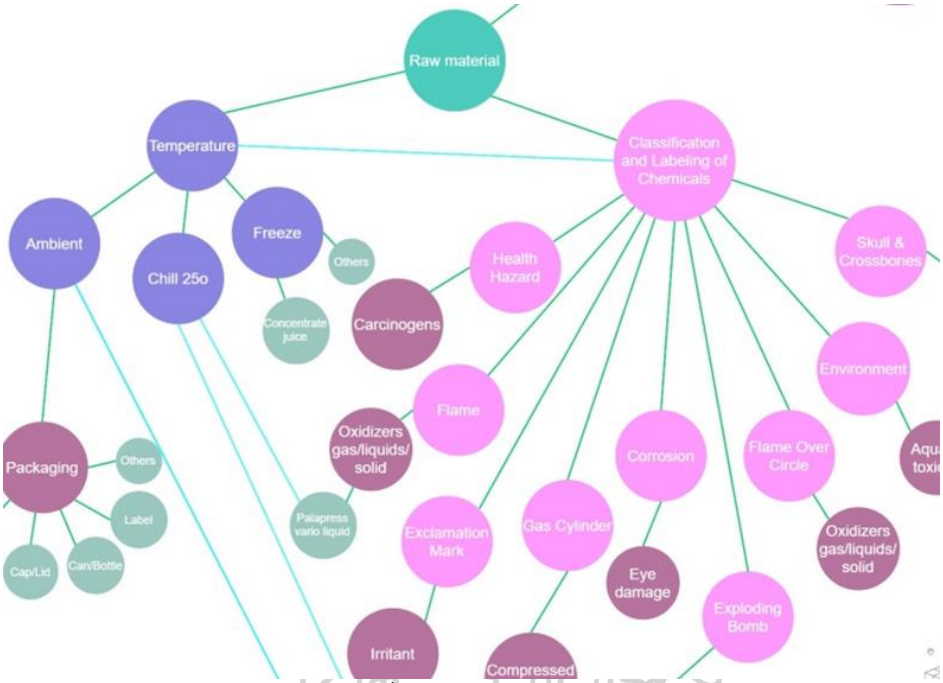
2.1.1.2 คลังสินค้าประเภทวัตถุดิบ (Raw Material Warehouse)

ประเภทนี้คือคลังสินค้าที่ใช้งานจัดเก็บวัตถุดิบก่อนที่จะนำไปทำการผลิต โดยมีทั้งวัตถุดิบที่ใช้ในการผลิตสินค้าและวัตถุดิบที่ใช้ในการบรรจุภัณฑ์อีกด้วย ซึ่งทำให้ประเภทของวัตถุที่จะจัดเก็บนั่นหลากหลายมาก

ยกตัวอย่างเช่นโรงงานผลิตเครื่องดื่มชูกำลังแห่งหนึ่ง มีรายการวัตถุที่จัดเก็บในคลังวัตถุดิบดังนี้ สาร Premix A, สาร Premix B, สาร Premix C, สาร Premix D, สาร Premix E, สาร Premix F, สารแต่งกลิ่น, สีผสม, น้ำผลไม้เข้มข้น, น้ำเชื่อมชนิดพิเศษ, น้ำตาล, กลุ่มวัตุดิบสำหรับบรรจุภัณฑ์, กล่อง, ขวด/กระป๋อง, ฝา, ฉลาก, แรพพลาสติก, เทปกาว

ตัวอย่างข้างต้นจะเห็นว่าประเภทของวัตถุดิบนั่นมีความหลากหลายมาก ซึ่งแต่ละประเภทนั่นมีเงื่อนไขการจัดเก็บที่เฉพาะตัว ดังนั่นทำให้การออกแบบวางผังอาคารคลังสินค้าประเภทวัตถุดิบ (Raw Material Warehouse)นั่นทำได้ยากกว่าคลังสินค้าประเภทสินค้าสำเร็จ (Finish Good Warehouse อย่างมาก

2.1.2 ขั้นตอนที่2 การจัดกลุ่มวัตถุที่จัดเก็บ (Classification)



ภาพที่ 6 แผนภาพสรุปการจัดกลุ่มวัตถุดิบ

จากขั้นตอนที่1 จะเห็นได้ว่าอาคารคลังสินค้าประเภทวัตถุดิบ (Raw Material Warehouse) นั่นมีวัตถุที่ต้องการจัดเก็บหลากหลายประเภทมาก จึงต้องทำการจำแนกประเภทหรือจัดกลุ่มวัตถุดิบตามเงื่อนไขที่จะมีผลต่อการวางผัง โดยเงื่อนที่มีผลต่อการวางผังมี 2เงื่อนไขหลักคือ อุณหภูมิและสารเคมี

2.1.2.1 เงื่อนไขเรื่องอุณหภูมิ (Temperature)

วัตถุดิบแต่ละชนิดจะมีความต้องการพื้นที่จัดเก็บในคลังสินค้าที่อุณหภูมิแตกต่างกัน จากตัวอย่างโรงงานผลิตเครื่องดื่มชูกำลังแห่งหนึ่งสามารถจัดกลุ่มสินค้ากับอุณหภูมิได้ดังภาพด้านล่าง

วัตถุดิบ	อุณหภูมิจัดเก็บ(องศาเซลเซียส)
สารPremix A	-18
สารPremix B	25
สารPremix C	35
สารPremix D	35
สารPremix E	25
สารPremix F	-18
สารแต่งกลิ่น	25
สีผสม	25
น้ำผลไม้เข้มข้น	-18
น้ำเชื่อมชนิดพิเศษ	25
น้ำตาล	32-35
กล่องกระดาษ	32-35
ขวด/กระป๋อง	32-35
ฝา	32-35
ฉลาก	32-35
แรพพลาสติก	32-35
เทปกาว	32-35

ภาพที่ 7 ภาพแสดงการจัดกลุ่มระหว่างวัตถุที่จัดเก็บกับอุณหภูมิ

2.1.2.2 เงื่อนไขประเภทสารเคมี (Classification and Labeling of Chemicals)

วัตถุดิบที่ใช้ในกระบวนการผลิตสินค้าบางชนิดเป็นสารเคมีซึ่งจะต้องถูกจำแนกโดยมาตรฐานที่เรียกว่า Globally Harmonized System of Classification and Labeling of Chemicals (GHS) ((GHS), 2020) คือระบบการจัดกลุ่มสารเคมี การติดฉลาก และการแสดงรายละเอียดบนเอกสารข้อมูลความปลอดภัย (Safety Data Sheet : SDS) เพื่อให้แต่ละประเทศสามารถสื่อสารและเข้าใจข้อมูลที่เกี่ยวข้องกับอันตรายที่เกิดจากสารเคมีในทิศทางเดียวกัน ซึ่งมีผลต่อความต้องการพื้นที่จัดเก็บในคลังสินค้าที่แตกต่างกัน โดยแบ่งเป็น 9 ประเภท

ประเภทสารเคมีที่เป็นวัตถุดิบ

ประเภทที่1 (Health Hazard) สารก่อมะเร็ง, สารกระตุ้นอาการแพ้ต่อระบบทางเดินหายใจ ,สารที่มีผลต่อการทำงานของอวัยวะภายในของมนุษย์

ประเภทที่2 (Flame) สารไวไฟ

ประเภทที่3 (Exclamation Mask) สารก่อการระคายเคืองต่อผิวหนัง, ระบบทางเดินหายใจ

ประเภทที่4 (Gas Cylinder) ก๊าซภายใต้ความดัน

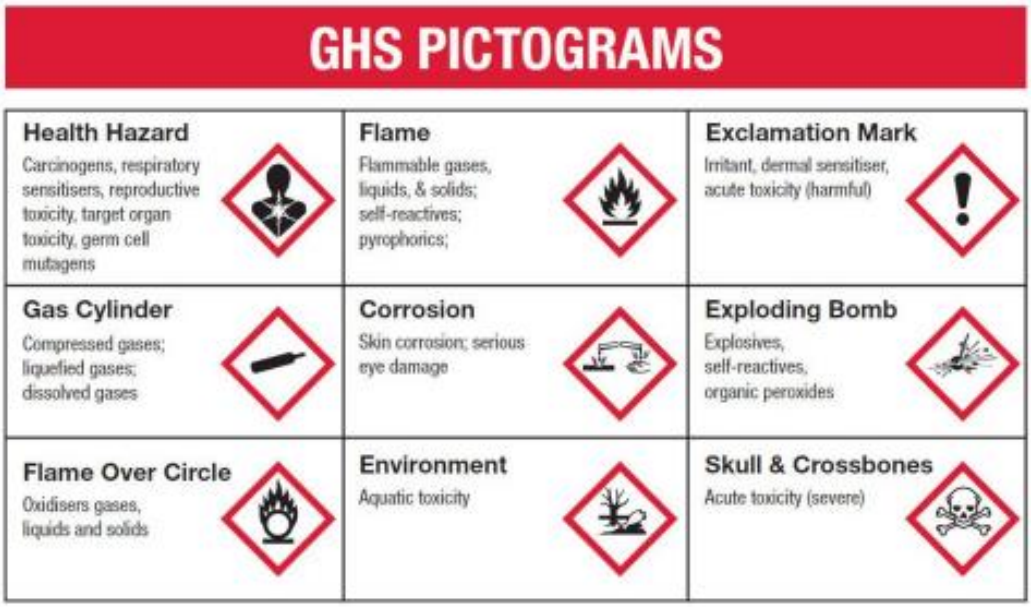
ประเภทที่5 (Corrosion) สารกัดกร่อนโลหะ

ประเภทที่6 (Exploding Bomb) วัตถุระเบิด

ประเภทที่7 (Flame over Circle) สารออกซิไดส์ติดไฟง่าย

ประเภทที่8 (Environment) ความเป็นอันตรายเฉียบพลันต่อสิ่งแวดล้อมในน้ำ

ประเภทที่9 (Skull & Crossbones) ความเป็นพิษเฉียบพลันเป็นอันตรายถึงตายได้



ภาพที่ 8 ประเภทสารเคมีที่เป็นวัตถุดิบ

จากตัวอย่างโรงงานผลิตเครื่องดื่มชูกำลังแห่งหนึ่งสามารถจัดกลุ่มอุณหภูมิสินค้ากับประเภทสารเคมี ได้ดังภาพด้านล่าง

วัตถุดิบ	อุณหภูมิจัดเก็บ (องศาเซลเซียส)	ประเภทสารเคมี
สารPremix A	-18	-
สารPremix B	25	Flame
สารPremix C	35	Exclamation Mask
สารPremix D	35	Exclamation Mask
สารPremix E	25	Flame
สารPremix F	-18	-
สารแต่งกลิ่น	25	-
สีผสม	25	-
น้ำผลไม้เข้มข้น	-18	-
น้ำเชื่อมชนิดพิเศษ	25	-
น้ำตาล	32-35	-
กล่องกระดาษ	32-35	-
ขวด/กระป๋อง	32-35	-
ฝา	32-35	-
ฉลาก	32-35	-
แรพพลาสติก	32-35	-
เทปกาว	32-35	-

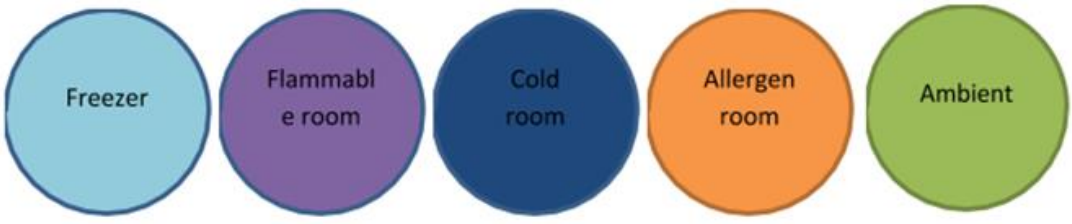
ภาพที่ 9 ภาพแสดงการจัดกลุ่มประเภทสารเคมี

จากนั้นทำการจัดกลุ่มพื้นที่จัดเก็บจาก 2 เงื่อนไขหลักโดยมีหลักการคือกลุ่มที่มีความต้องการในการจัดเก็บเหมือนกันทั้งเรื่องของอุณหภูมิ และ ประเภทสารเคมีจะถูกจัดให้อยู่กลุ่มเดียวกัน ดังภาพด้านล่าง

วัตถุดิบ	อุณหภูมิจัดเก็บ (องศาเซลเซียส)	ประเภทสารเคมี	พื้นที่ (Zone)
สารPremix A	-18	-	Freezer
สารPremix B	25	Flame	Flammable room
สารPremix C	35	Exclamation Mask	Allergen room
สารPremix D	35	Exclamation Mask	Allergen room
สารPremix E	25	Flame	Flammable room
สารPremix F	-18	-	Freezer
สารแต่งกลิ่น	25	-	Cold-room
สีผสม	25	-	Cold-room
น้ำผลไม้เข้มข้น	-18	-	Freezer
น้ำเชื่อมชนิดพิเศษ	25	-	Cold-room
น้ำตาล	32-35	-	Ambient zone
กล่องกระดาษ	32-35	-	Ambient zone
ขวด/กระป๋อง	32-35	-	Ambient zone
ฝา	32-35	-	Ambient zone
ฉลาก	32-35	-	Ambient zone
แรพพลาสติก	32-35	-	Ambient zone
เทปกาว	32-35	-	Ambient zone

ภาพที่ 10 ภาพแสดงการจัดกลุ่มอุณหภูมิสินค้ากับประเภทสารเคมีเพื่อกำหนดพื้นที่

สรุปจากการจำแนกจะได้กลุ่มพื้นที่มาทั้งหมด 5 กลุ่มพื้นที่ ดังนี้



ภาพที่ 11 ภาพแสดงกลุ่มพื้นที่ที่เกิดจากการจัดกลุ่มอุณหภูมิสินค้ากับประเภทสารเคมี

พื้นที่ที่1 Freezer พื้นที่จัดเก็บวัตถุดิบที่อุณหภูมิติดลบ 18องศา

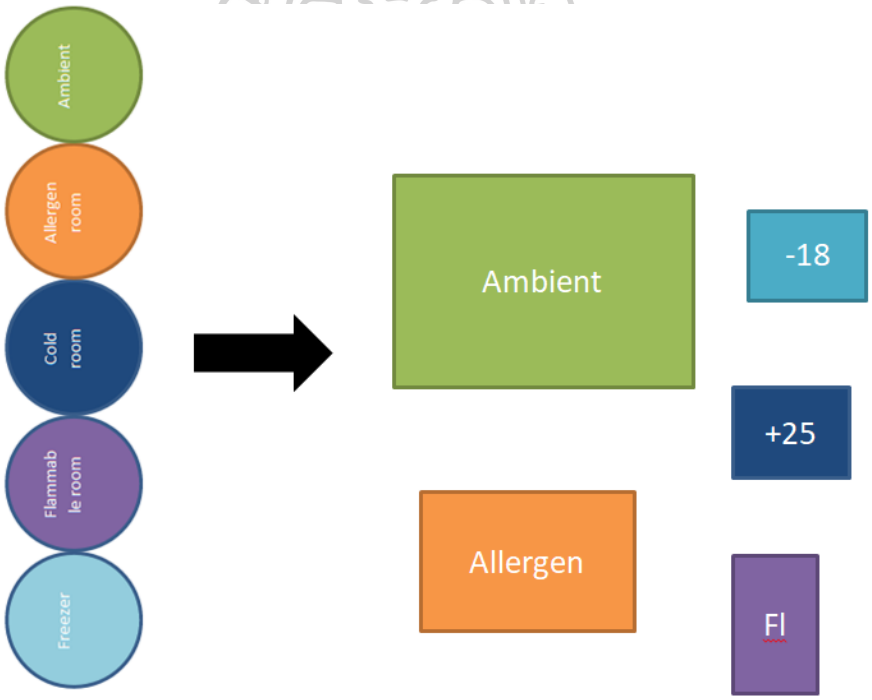
พื้นที่ที่2 Flammable พื้นที่จัดเก็บวัตถุดิบประเภทที่7 (Flame over Circle) สารออกซิไดส์ติดไฟง่าย โดยจัดเก็บภายใต้อุณหภูมิ 25 องศา

พื้นที่ที่3 Cold room พื้นที่จัดเก็บวัตถุดิบที่อุณหภูมิ 25องศา

พื้นที่ที่4 Allergen พื้นที่จัดเก็บวัตถุดิบประเภทที่3 (Exclamation Mask) สารก่อการระคายเคืองต่อผิวหนัง, ระบบทางเดินหายใจ

พื้นที่ที่5 Ambient พื้นที่จัดเก็บวัตถุดิบที่อุณหภูมิห้อง 32- 35องศา

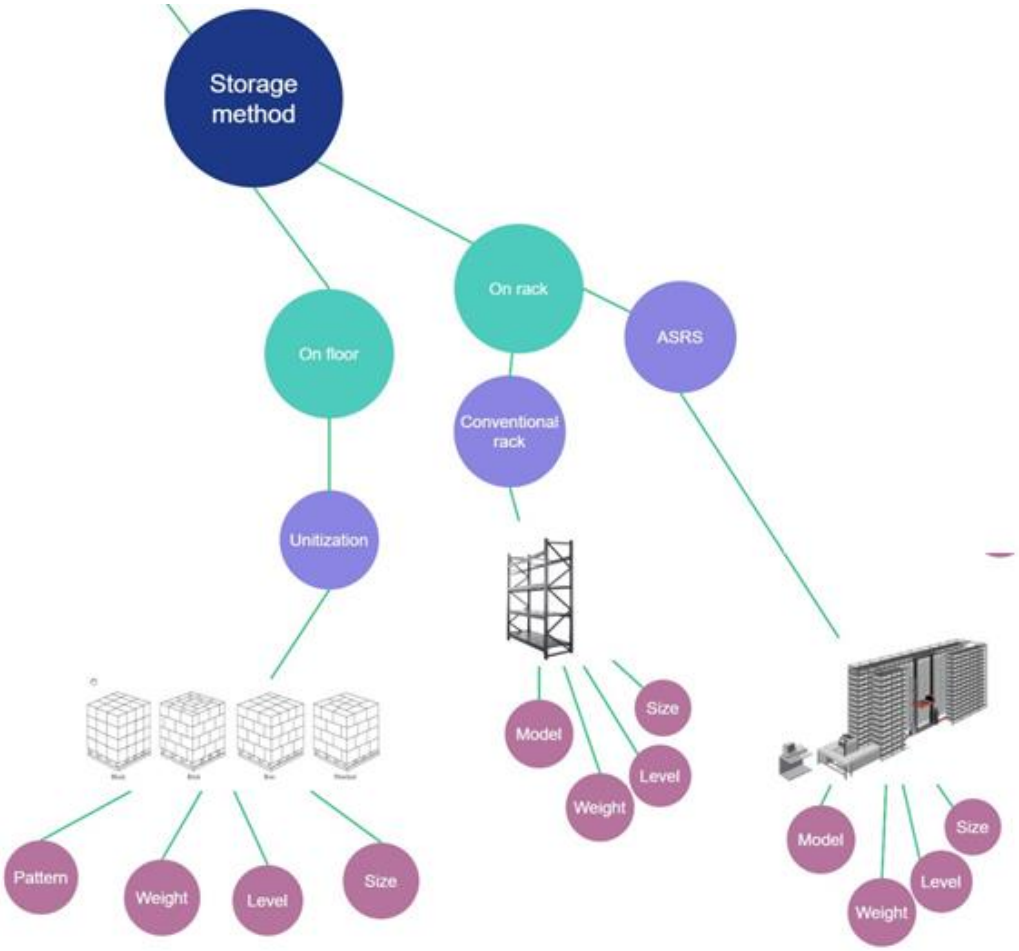
2.1.3 ขั้นตอนที่3 การหาขนาดของแต่ละกลุ่มพื้นที่ (Create Space)



ภาพที่ 12 แผนภาพแสดงขนาดที่แตกต่างกันของแต่ละกลุ่มพื้นที่

หลังจากจำแนกวัตถุที่จัดเก็บออกมาเป็นกลุ่มพื้นที่เรียร้อยขั้นตอนต่อมาคือการหาขนาดของแต่ละกลุ่มพื้นที่ ซึ่งปัจจัยที่มีผลและถูกนำมาใช้หาขนาดพื้นที่จัดเก็บในคลังคือวิธีการจัดเก็บ(Storage method)และ วิธีการขนส่งสินค้า(Method of transportation) ทำให้สามารถจำแนกพื้นที่ของคลังสินค้าได้2ส่วนดังนี้

2.1.3.1 ประเภทพื้นที่คลังสินค้า



ภาพที่ 13 แผนภาพสรุปความสัมพันธ์ของประเภทพื้นที่คลังสินค้า

ประเภทที่1 พื้นที่การจัดเก็บ (Storage Areas)

แบ่งเป็น 2 แบบหลักตามวิธีการจัดเก็บสินค้า คือ

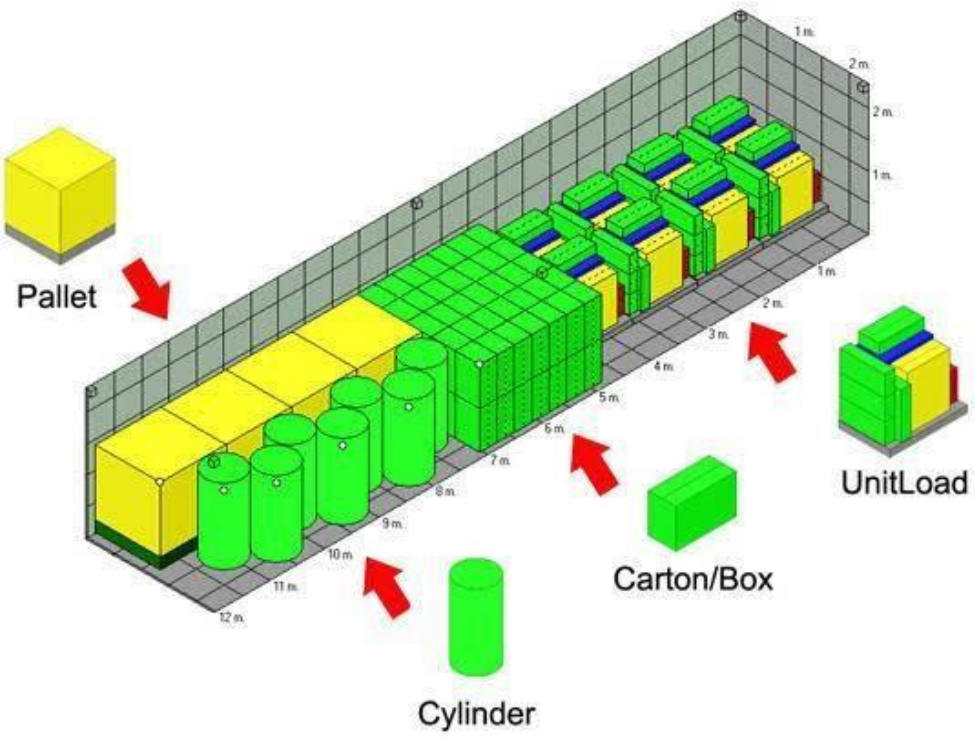
แบบที่1 พื้นที่จัดเก็บสินค้าแบบกองบนพื้น (On floor)

ภาพที่ 14 ตัวอย่างการวางสินค้าแบบ On floor

การจัดเก็บสินค้าวิธีนี้จะใช้กับสินค้าที่บรรจุภัณฑ์แล้ว ซึ่งวิธีการจัดเก็บสินค้าดังกล่าวจะบอกถึง

- จำนวนที่จะจัดเก็บ (Quantity)
- ขนาดสินค้าที่จัดเก็บ (Sizing)
- จำนวน/วิธีการซ้อนชั้นสินค้า (Level)
- วิธีการจัดวางสินค้า (Pattern)
- น้ำหนักสินค้า (Weight)

โดยปัจจัยข้างต้นนี้สามารถนำมาคำนวนหาขนาดของแต่ละกลุ่มพื้นที่ได้ โดยรายละเอียดวิธีการคำนวนจะอธิบายอย่างละเอียดในบทที่ 2



ภาพที่ 15 ตัวอย่างการจัดวางสินค้า On floor โดย Cargo Optimizer Enterprise เป็นโปรแกรมที่ใช้คำนวณการจัดเรียงสินค้า

แบบที่2 พื้นที่จัดเก็บสินค้าบนชั้นวาง (On rack)

ภาพที่ 16 ตัวอย่างการวางสินค้าแบบ On rack

การจัดเก็บสินค้าวิธีนี้มักจะใช้กับคลังสินค้าทุกรูปแบบโดยนำสินค้าขึ้นชั้นวางเป็นการจัดเก็บในแนวสูงหากเปรียบเทียบกับการจัดเก็บสินค้าแบบกองบนพื้น (On floor) จะทำให้ได้จำนวนในการจัดเก็บมากขึ้นในขณะที่พื้นที่อาคารเท่าเดิมหรือเล็กลง โดยวิธีการจัดเก็บดังกล่าวจะบอกถึง

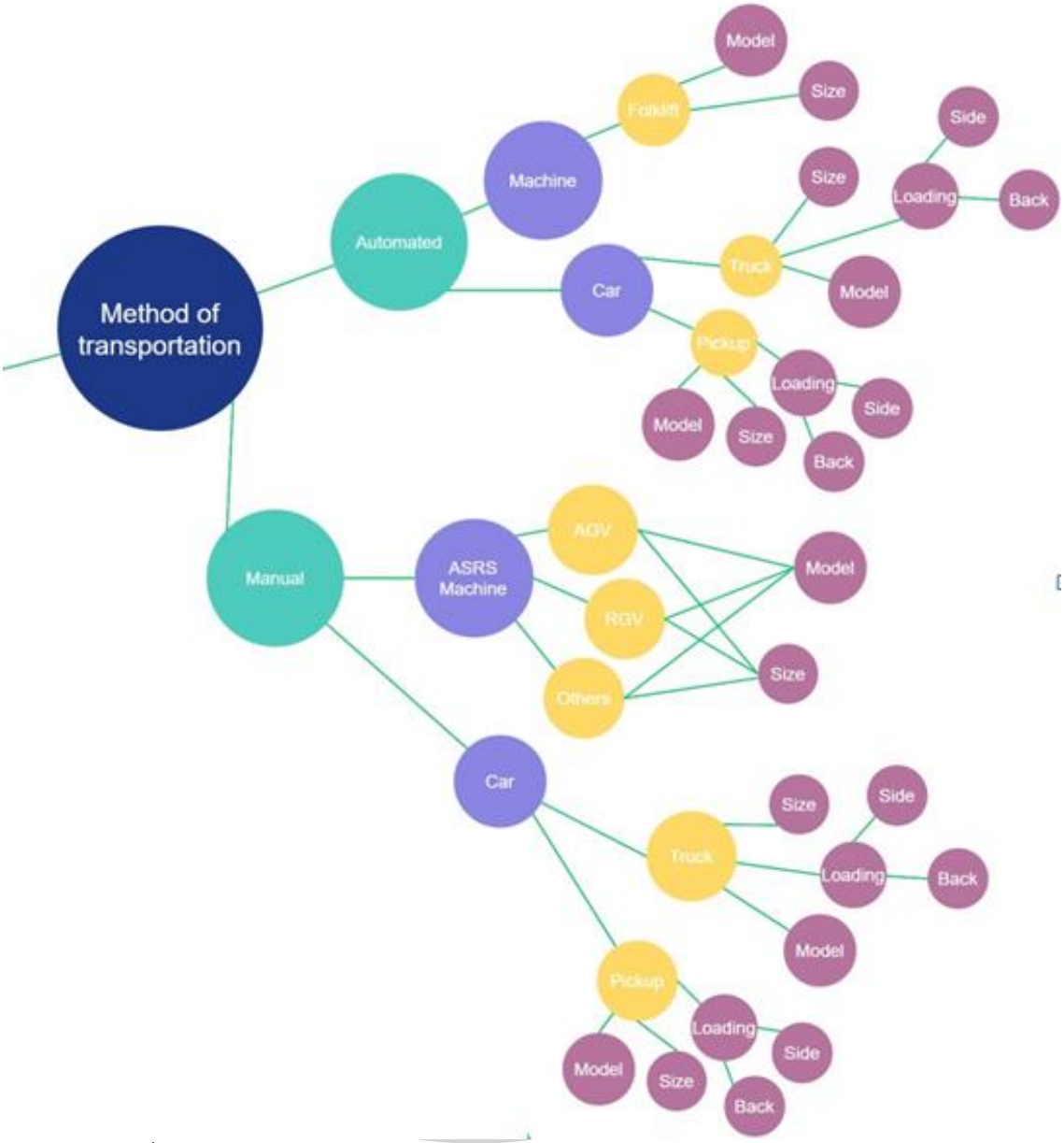
- จำนวนที่จะจัดเก็บ (Quantity)
- ขนาดสินค้าที่จัดเก็บ (Sizing)
-รุ่นและข้อมูลจำเพาะของชั้นวาง (Rack model)
- ความสูงชั้นวางสินค้า (Rack Level)
- น้ำหนักสินค้า (Weight)

โดยปัจจัยข้างต้นนี้สามารถนำมาคำนวนหาขนาดของแต่ละกลุ่มพื้นที่ได้ โดยรายละเอียดวิธีการคำนวนจะเพิ่มเติมในบทที่ 2

ประเภทที่2 พื้นที่ขนส่งสินค้า (Loading Area)

พื้นที่ขนส่งสินค้าหรือที่เรียกกันโดยทั่วไปว่า “ลานโหลด” มาจากความหมายเต็มว่า พื้นที่ลานขนถ่ายสินค้า มีขนาดที่หลากโดยขึ้นอยู่กับประเภทของรถ (Car Types) และวิธีการขนส่ง (Method of Transportation) ที่ใช้ในการขนถ่ายสินค้า

มหาวิทยาลัยศิลปากร



ภาพที่ 17 รูปด้านบนเป็นแผนภาพของวิธีการขนส่ง (Method of Transportation)

ประเภทรถที่ใช้ในการขนส่งสินค้า(Car Types)

รูปแบบของรถขนส่งสินค้านั่นมีด้วยกันหลากหลายรุ่น แต่ที่นิยมใช้ในประเทศไทยจะมีด้วยกัน 3 ขนาด คือ รถPick-Up 4ล้อ , รถTruck 6ล้อ และ รถ Trailer 40 feet ซึ่งรถทั้งสามชนิดต้องการพื้นที่ใช้งานที่แตกต่างกัน เช่น วงเลี้ยว ,ระยะถอย ,ระยะจอด ตามขนาดของตัวรถแต่ละชนิดดังนี้ (kargo, 2017)

-รถPick up 4ล้อ

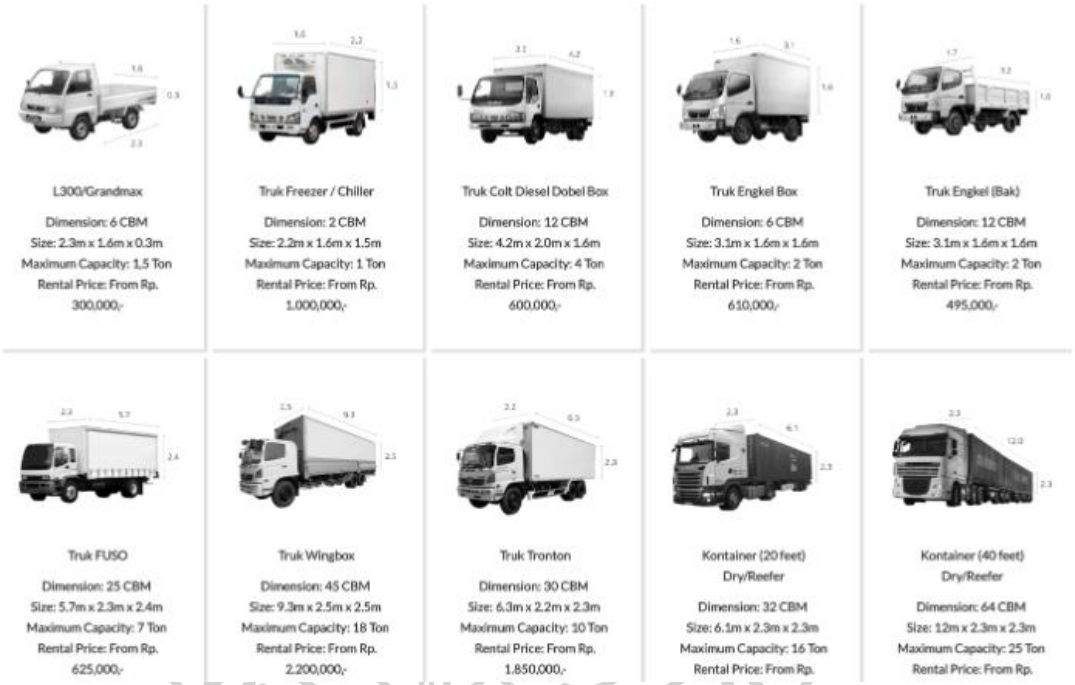
โดยต้องการพื้นที่ในการจอดรวมระยะวงเลี้ยวเท่ากับ 12 x 4 ตร.ม.

-รถTruck 6ล้อ

โดยต้องการพื้นที่ในการจอดรวมระยะวงเลี้ยวเท่ากับ 15 x 4 ตร.ม.

-รถ Trailer 40 feet

โดยต้องการพื้นที่ในการจอดรวมระยะวงเลี้ยวเท่ากับ 30 x 4 ตร.ม.



ภาพที่ 18 รถประเภทต่างๆที่ใช้ในการขนส่ง (kargo, 2017)

2.1.3.2 วิธีคำนวณพื้นที่คลังสินค้า

วิธีคำนวณพื้นที่คลังประเภทที่1 พื้นที่การจัดเก็บ (Storage Areas)

ขั้นตอนที่1 เมื่อได้ความต้องการของปริมาณการจัดเก็บสินค้าจากลูกค้าแล้ว สิ่งแรกที่ต้องพิจารณาคือขนาดพาเลทหรือขนาดผลิตภัณฑ์ นี่เป็นส่วนที่สำคัญที่สุดเนื่องจากจะมีผลต่อตัวแปรที่เหลือในการกำหนดพื้นที่คลังสินค้า โดยพาเลต(Pallet) มีด้วยกันหลากหลายชนิดตามวัสดุที่นำมาทำ อย่างเช่น พาเลทพลาสติก พาเลทเหล็ก แต่ที่ใช้กันส่วนใหญ่เนื่องต้นทุนที่ไม่สูงนั่นคือพาเลทไม้ ซึ่งมีขนาดมาตรฐานที่ใช้กันมากที่สุดตาม ISO (International Standards Organization) มีอยู่ 3 ขนาด ดังนี้ (Loscam, 2020)

(1) ขนาด 80 x 120 x 1.5 cm. หรือ (EURO Pallet)

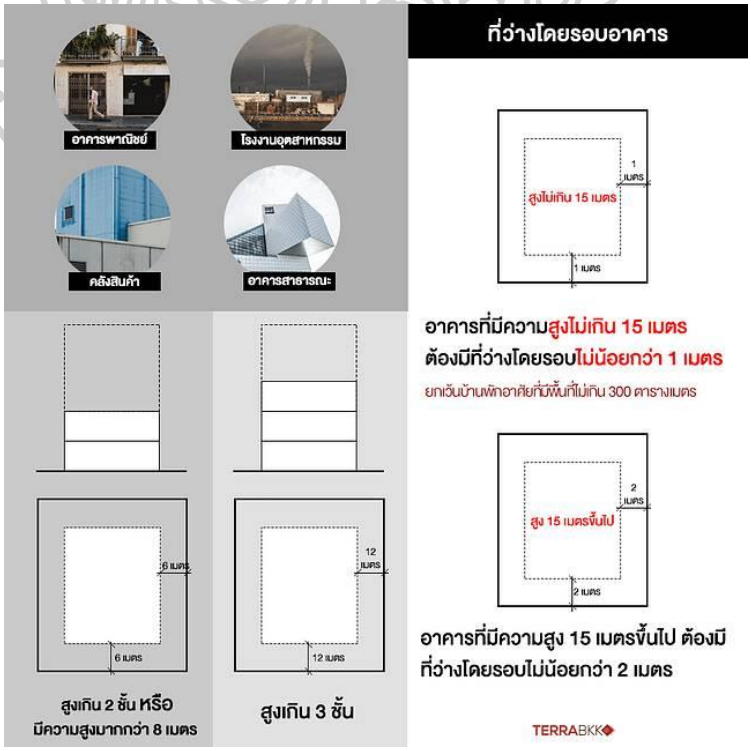
(2) ขนาด 110 x 110 x 1.5 cm. หรือ Japan Pallet

(3) ขนาด 100 x 120 x 1.5 cm. หรือ Thai Pallet

ภาพที่ 19 ภาพตัวอย่าง Pallet ของบริษัท Roscam ที่นิยมใช้ในประเทศไทย (Loscam, 2020)

ขั้นตอนที่2 วัดขนาดอาคารโดยรวม(Site Area)

ขนาดอาคารโดยรวมที่สามารถสร้างคลังสินค้าได้โดยการนำพื้นที่ดินหักลบระยะร่นตามกฎหมายออก กำหนดพื้นที่ใช้งานอื่นๆที่นอกเหนือจากพื้นที่จัดเก็บสินค้า เช่น สำนักงาน,บล็อกห้องน้ำ ,บันได ,ฯลฯ การร่างพื้นที่ต่างๆลงบนแผ่นกระดาษจะทำให้มีผังของคลังสินค้าขั้นต้นได้

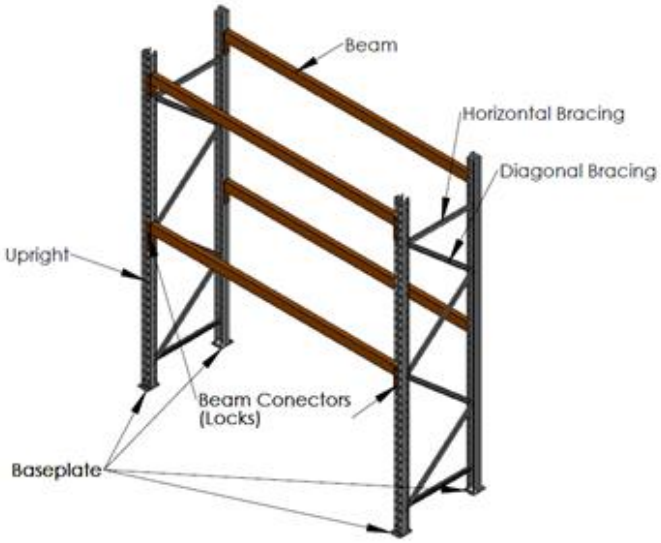


ภาพที่ 20 ตัวอย่างระยะร่นตามกฎหมายของอาคารประเภทคลังสินค้า ภาพจาก TERRABKK

ขั้นตอนที่3 วัดความสูงของอาคาร

ความสูงของอาคารจะสัมพันธ์กับจำนนการซ้อนชั้นของความสูงชั้นวาง (Rack)โดยทั่วไปจำนวนการซ้อนชั้นของชั้นวาง(Rack) จะถูกกำหนดจากความต้องการของลูกค้า

ขั้นตอนที่4 การจัดวางชั้นวางสินค้า (Rack)



ภาพที่ 21 ภาพแสดงองค์ประกอบของชั้นวางสินค้า (macrack, 2017)

ชั้นวางสินค้ามีส่วนประกอบหลักสองส่วน

(1) ส่วนแรกคือ เฟรม (Frame)

ประกอบด้วยโครงสร้างเสาหลักและเสาค้ำยันระหว่าง โดยทั่วไปเฟรมจะมีความลึกประมาณ 840 มม. และมีความสูงได้ทุกระดับ

(2) ส่วนที่สองคือ คาน (Beams)

คือโครงสร้างล็อคเข้ากับเฟรม สามารถปรับความสูงได้ตามความต้องการ โดยความยาวคานมีขนาดแปรผันตามจำนวนพาเลทที่สามารถรองรับได้ โดยมีขนาดมาตรฐานดังนี้

- ชั้นวาง (Rack) รองรับ 1 พาเลท

คาน (Beams) ยาว 1372 มม.ต่อ 1 ชุด

- ชั้นวาง (Rack) รองรับ 2 พาเลท

คาน (Beams) ยาว 2590 มม.ต่อ 1 ชุด

- ชั้นวาง (Rack) รองรับ 3 พาเลท

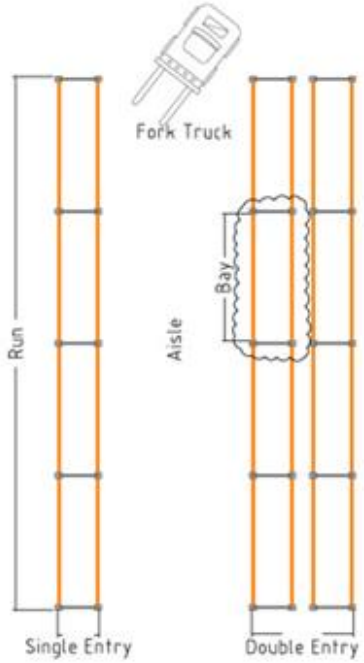
คาน (Beams) ยาว 3658 มม.ต่อ 1 ชุด

ภาพที่ 22 ตัวอย่าง Selective Rack จาก www.lpi.co.th

จากข้อมูลขนาดของชั้นวางสินค้าขั้นตอนนี้สามารถเริ่มวางผังโดยการจัดเรียงขนาดของชั้นวางสินค้าลงบนพื้นได้ โดยมีข้อกำหนดเล็กน้อยที่จะลืมไม่ได้เช่น

- เฟรมมีความหนาประมาณ 100 มิลลิเมตร
- ชั้นวาง (Rack) ต้องอยู่ห่างจากผนังหรือเสา 200 มิลลิเมตร
- ชั้นวาง (Rack) แต่ละแถว (Column) มีระยะห่างกันประมาณ 430 มิลลิเมตร อาจแตกต่างกันตามมาตรฐานของผู้ผลิตแต่ละบริษัท

ขั้นตอนที่5 กำหนดขนาดช่องทางสัญจร(Circulation)



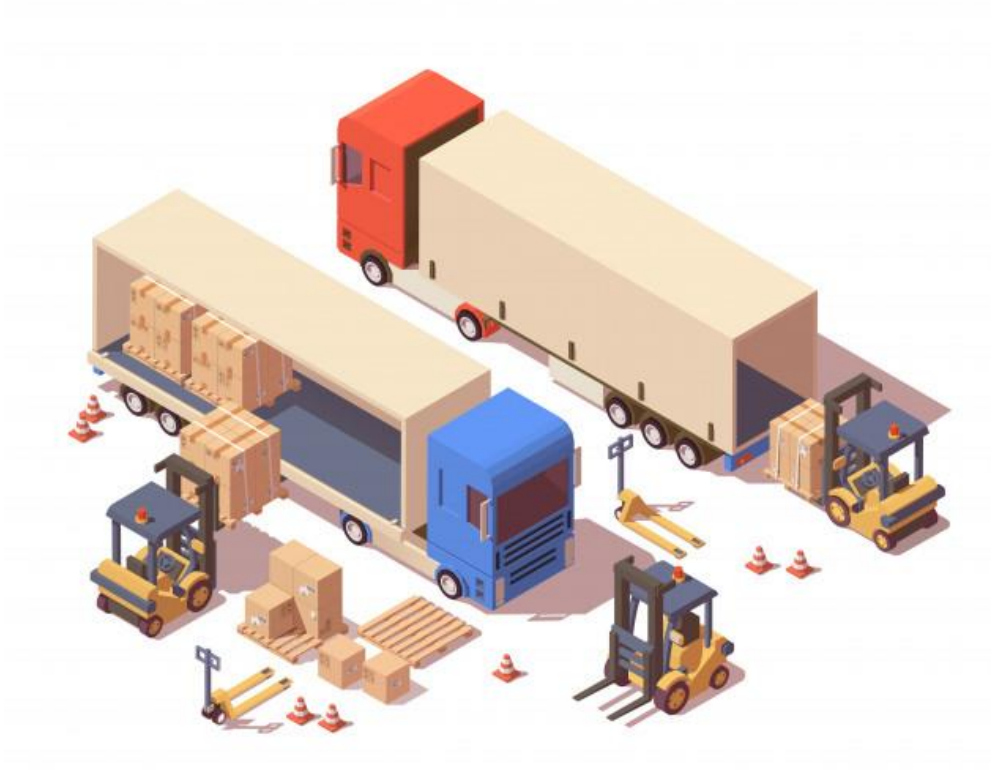
ภาพที่ 23 ภาพแสดงทางสัญจรของรถยก

การจัดวางชั้นวางสินค้านั่นสิ่งหนึ่งที่สถาปนิกต้องคำนึงถึงคือการวางรูปแบบการสัญจรในการขนถ่ายสินค้าซึ่งขนาดรถโฟลคลิฟท์จะส่งผลโดยตรงกับขนาดทางสัญจรที่จะถูกจัดวางลงบนผัง ขนาดทางสัญจรนี้อาจแตกต่างกันไปมากน้อยขึ้นอยู่กับประเภทของรถโฟลคลิฟท์โดยขึ้นอยู่กับขนาดของสินค้า ความสูงของชั้นวาง โดยส่วนใหญ่จะเป็นข้อมูลที่ถูกกำหนดโดยลูกค้า

ปกติสำหรับทางสัญจรรถโฟลค์ลิฟท์ทั่วไปจะอยู่ที่ประมาณ 4.5 เมตร รถโฟลค์ลิฟท์ไฟฟ้าขนาดเล็กอาจลงไปได้ถึง 3 เมตร

นอกเหนือจากทางสัญจรของโฟลค์ลิฟท์นั่นยังมีพื้นที่ทางสัญจรที่อาจไม่ยังไม่ปรากฏในขั้นตอนการออกแบบขั้นต้น ยกตัวอย่างเช่น ช่องทางเดินคน, พื้นที่ติดตั้งงานระบบต่างๆ ดังนั่นผู้ออกแบบมักคูณพื้นที่จะได้เพิ่มอีก 10% - 30% ตามความซับซ้อนที่พิจารณาในแต่ละประเภทของงาน (macrack, 2017)

วิธีคำนวณพื้นที่คลังประเภทที่2 พื้นที่ขนส่งสินค้า (Loading Area)



ภาพที่ 24 ภาพแสดงรูปแบบการโหลดสินค้า Back Loading (แดง) และ Side Loading (น้ำเงิน)

ความยาวของลานโหลด ถูกกำหนดจาก ความยาวรถชนิดที่ใหญ่ที่สุดที่ใช้รวม กับพื้นที่วงเลี้ยวจะเท่ากับ ระยะความยาวรวมของพื้นที่ที่ใช้ในการโหลดสินค้า

ความกว้างลานโหลด ถูกกำหนดจาก รูปแบบการขนส่งสินค้า ซึ่งแบ่งเป็น 2แบบหลัก คือ

1.การโหลดสินค้าขึ้นท้ายรถ (Back Loading)

การโหลดสินค้าแบบนี้ไม่ต้องการพื้นที่ระหว่างรถแต่ละคันมากนัก โดยควรห่างกันประมาณ 1เมตร

2.การโหลดสินค้าขึ้นข้างรถ (Side Loading)

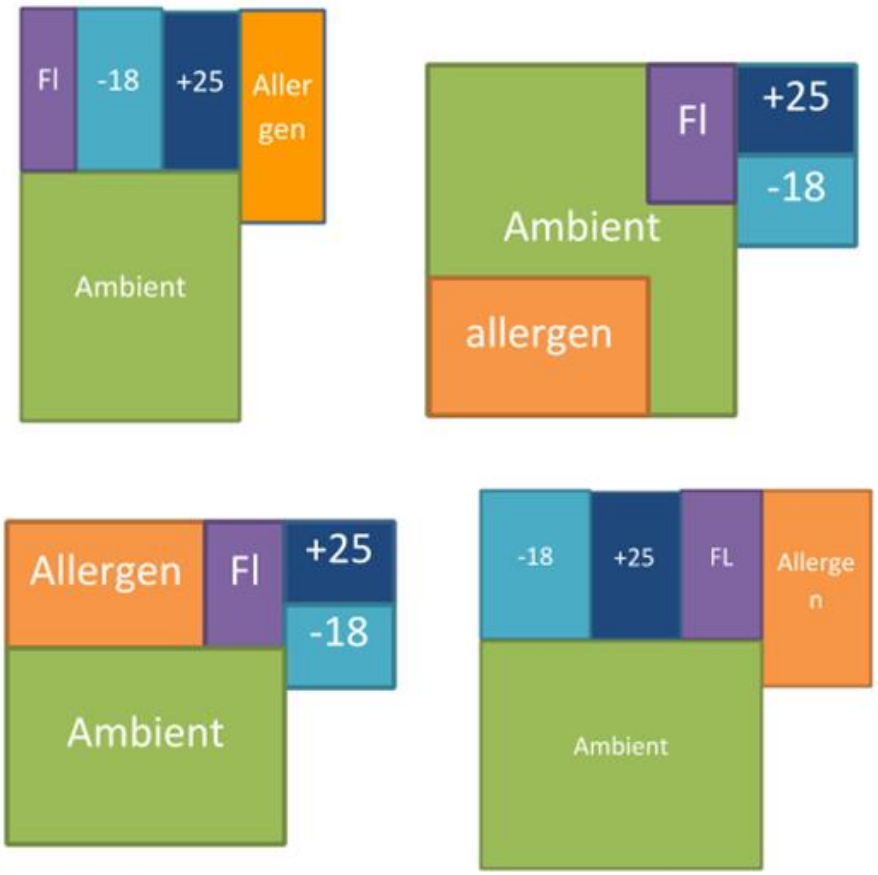
การโหลดสินค้าแบบนี้ต้องการพื้นที่ระหว่างรถแต่ละคันมากเนื่องจากต้องใช้รถโฟลคลิฟ ขนส่งสินค้าเข้าด้านข้างรถ ดังนั่นทั้งสองฝั่งของรถต้องการพื้นที่ในการขนถ่ายสินค้า ทำให้ต้องการพื้นที่ระหว่างรถแต่ละคัน ประมาณ 3.5 เมตรเป็นอย่างต่ำ

ดังนั่นจึงสรุปได้ว่าความกว้างของลานโหลด เท่ากับ ความของรถชนิดที่ใหญ่ที่สุด บวกกับพื้นที่ระหว่างรถแต่ละคันตามรูปแบบการโหลดสินค้าแต่ละแบบ โดยทั้งหมดคูณจำนวนรถแต่ต้องการใช้

2.1.4 ขั้นตอนที่ 4 การจัดเรียงพื้นที่คลังสินค้าโดยมีผลลัพธ์การวางผังคลังสินค้าที่หลากหลายตามเงื่อนไขของวัตถุที่จัดเก็บ (Space Planning and Alternative Design)

นำพื้นที่แต่ละประเภทที่ถูกกำหนดขนาดแล้วมาจัดวางผัง โดยสถาปนิกภายใต้เงื่อนไขของวัตถุที่จัดเก็บ 2เงื่อนไขหลัก คือ อุณหภูมิ (Temperature) และ ประเภทสารเคมี (Classification and Labeling of Chemicals) โดยพิจารณาจากหลักการการใกล้กันได้ของพื้นที่แต่ละประเภท (Adjacent) หรือไม่ควรใกล้กันของพื้นที่แต่ละประเภท (Nonadjacent) ยกตัวอย่างเช่น ห้องที่เก็บสารไวไฟไม่ควรติดกับห้องที่อาจเกิดประกายไฟได้ ,ห้องที่อุณหภูมิติดลบควรอยู่ติดกับห้องที่อุณหภูมิใกล้เคียงกันเพื่อป้องกันการควบแน่นของความชื้นเกิดเป็นน้ำหยดในห้อง เป็นต้น

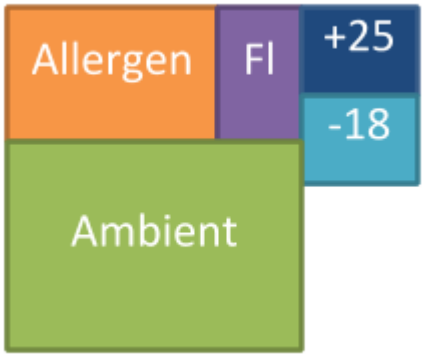
ผลลัพธ์ของการจัดเรียงพื้นที่จะออกมาในผลลัพธ์ของผังอาคารคลังสินค้าที่มากกว่าหนึ่งรูปแบบเพื่อให้เกิดตัวเลือกที่หลากหลายเพียงพอที่จะนำไปสู่การพิจารณาถึงความเหมาะสมในปัจจัยต่างๆร่วมกันระหว่างผู้ออกแบบและเจ้าของคลังสินค้าหรือผู้ใช้งานคลังสินค้า เพื่อเลือกผลลัพธ์สุดท้ายของผังคลังสินค้าที่เหมาะสมที่สุด



ภาพที่ 25 ตัวอย่างการจัดเรียงผังอาคารคลังสินค้าของโรงงานเครื่องดื่มชูกำลังแห่งหนึ่ง

2.1.5 ขั้นตอนที่5 การพิจารณาแบบที่เหมาะสมที่สุดกับการทำงานของโรงงานนั้นๆ (Discussion)

รูปแบบผังอาคารคลังสินค้าจะถูกพิจารณารวมกับเจ้าของโรงงานซึ่งเป็นผู้ใช้งานจริงเพื่อให้สอดคล้องและราบรื่นกับการทำงานที่จะเกิดขึ้นในอนาคต



ภาพที่ 26 ตัวอย่างการผลลัพธ์ของผังอาคารคลังสินค้าของโรงงานเครื่องดื่มชูกำลังแห่งหนึ่ง

2.1.5.1 หลักเกณฑ์การพิจารณาเลือกแบบอาคารคลังสินค้า

หลักเกณฑ์การตัดสินใจเลือกแบบผังอาคารคลังสินค้านั่นเกิดจากการตัดสินใจร่วมกันระหว่างผู้ออกแบบคือสถาปนิกและลูกค้าซึ่งเป็นผู้กำหนดความต้องโดยเกณฑ์การพิจารณาของผู้ออกแบบนั่นจะพิจารณาจากการใช้งานที่แตกต่างกันของแต่ละห้องเป็นหลักว่าการจัดวางนั่นมีความเหมาะสมและเป็นไปตามประเภทของกลุ่มวัตถุที่จัดเก็บหรือไม่ โดยคำนึงถึงลักษณะและวิธีการทำงานของผู้ใช้งาน และ ความปลอดภัยในการจัดเก็บเป็นสำคัญ

ในส่วนเกณฑ์การพิจารณาของลูกค้านั่นมักจะพิจารณาว่าผังอาคารที่จัดออกมานั่นสามารถตอบสนองต่อความต้องการของลูกค้าที่ให้ไว้ในตอนต้นครบถ้วนหรือไม่ ซึ่งกรณีของคลังสินค้านั่นสิ่งที่จะถูกพิจารณาเป็นอันแรกคือ ปริมาณการจัดเก็บในแต่ห้องหรือแต่ละพื้นที่นั่นเพียงพอต่อความต้องการในการจัดเก็บสินค้า

หลักเกณฑ์การตัดสินใจในการเลือกรูปแบบผังคลังสินค้านั่นสามารถสรุปได้เป็น 3ส่วนดังนี้

(1) การใช้งานที่เหมาะสมสอดคล้องกับการทำงานของผู้ใช้งาน

(2) ความปลอดภัยในการจัดเก็บวัตถุดิบแต่ละประเภท

(3) ปริมาณการจัดเก็บที่เพียงพอ

2.2 ปัญหาของการวางผังในปัจจุบัน

จากขั้นตอนการวางผังอาคารคลังสินค้าที่อธิบายมาข้างต้นจะสังเกตุได้ว่ามีความซับซ้อนสูงโดยแปรผันตามความหลากหลายของประเภทของที่จัดเก็บ หากความซับซ้อนมากขั้นตอนการวางผังจะใช้เวลานานและต้องการความละเอียดของการวิเคราะการวางผังสูง ปัจจัย2สิ่งที่กระทบต่อวิธีการวางผังอาคารคลังสินค้าในปัจจุบันคือ

2.2.1 ความซับซ้อนในการวางผังคลังสินค้า

อาคารคลังสินค้าที่มีความหมากหลายของประเภทสินค้าสูงย่อมต้องการการวิเคราะการวางผังที่มีประสิทธิภาพสูงตาม จากวิธีการวางผังแบบเดิมเป็นการนำกลุ่มของพื้นที่มาจัดเรียงเพื่อวางผังโดยสถาปนิกเอง ซึ่งมีโอกาสที่จะได้รูปแบบที่ไม่ตรงหรือผิดไปจากเงื่อนไขการจัดเก็บ โดยเฉพาะยิ่งประเภทวัตถุหลากหลาย ยิ่งยากต่อการวางผังตามเงื่อนไข และยิ่งเกิดความคลาดเคลื่อนและผิดพลาดได้ง่าย โดยสามารถสรุปปัจจัยที่มีผลต่อความซับซ้อนในการวางผังคลังสินค้าได้ดังนี้

2.2.1.1 ความซับซ้อนจากขั้นตอนการทำงาน

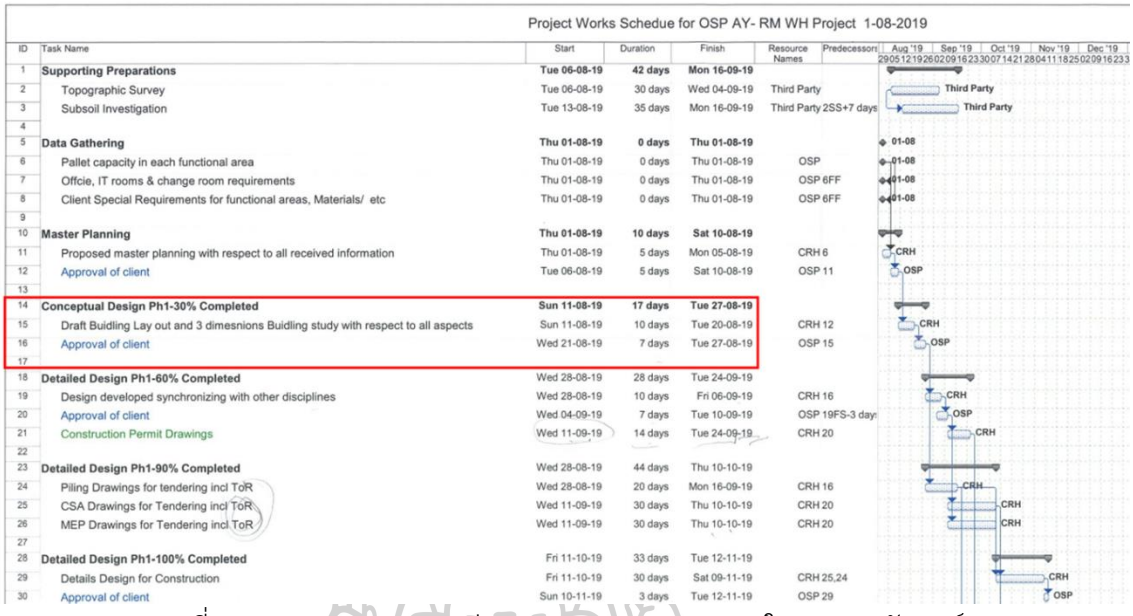
ขั้นตอนการทำงานแบบปัจจุบันที่ทำโดยสถาปนิกสามารถสรุปออกมาได้ถึง 5 ขั้นตอนการทำงานโดยได้อธิบายให้เห็นถึงรายละเอียดกับทำงานที่ซับซ้อนไปแล้วในบทที่ 2 ข้อที่ 2.1 ขั้นตอนการวางผังคลังสินค้าในปัจจุบัน ซึ่งทุกขั้นตอนล้วนแล้วแต่ทำการประมวลผลโดยสถาปนิกผู้ออกแบบเพียงคนเดียว ทำให้มีผลกระทบต่อทั้งคุณภาพในแง่ของความถูกต้องแม่นยำของข้อมูลในการออกแบบวางผังคลังสินค้าที่จะเกิด และปริมาณของแบบที่มีความหลากหลายเพื่อจะถูกพิจารณารวมกับเจ้าของโรงงานซึ่งเป็นผู้ใช้งานจริงเพื่อให้สอดคล้องและราบรื่นกับการทำงานที่จะเกิดขึ้นกับการใช้งานคลังสินค้าจริงในอนาคต

2.2.1.2 ความซับซ้อนจากเงื่อนไขในการวางผังคลังสินค้า

อ้างอิงจากรายละเอียดในบทที่ 2 ข้อที่ 2.1.4 ขั้นตอนที่ 4 การจัดเรียงพื้นที่คลังสินค้าโดยมีผลลัพธ์การวางผังคลังสินค้าที่หลากหลายตามเงื่อนไขของวัตถุที่จัดเก็บ (space planning and alternative design) แสดงให้เห็นแล้วว่าการจัดวางผังคลังสินค้านั่นมีความจำเป็นต้องคำนึงถึงเงื่อนไขต่างๆมากมาย อีกทั้งยังต้องการรูปแบบของผังคลังสินค้าที่หลากหลาย ซึ่งในบางครั้งการประมวลผลโดยสถาปนิก ซึ่งเป็นมนุษย์นั่นมีโอกาสที่จะเกิดข้อผิดพลาดที่ไม่เป็นไปตามเงื่อนไขได้

2.2.2 เวลา

ขั้นตอนการออกแบบวางผัง เป็นขั้นตอนในช่วงที่เรียกว่า Conceptual design คือ ช่วงของการออกแบบงานสถาปัตยกรรมเบื้องต้น เป็นตัวกำหนด การใช้งาน(Function) ตำแหน่ง (Location) ,ขอบเขต(boundary) ,พื้นที่(Area) และรูปร่าง หน้าตา(Shape Form) ของอาคารนั้นๆ ซึ่งระยะเวลาการทำงานโดยทั่วไปในช่วงนี้มีประมาณ 3 อาทิตย์ ถึง 1เดือน โดยขอบเขตของการทำงาน(Scope of work)ในช่วงนี้นั่นไม่เพียงแต่ขั้นตอนของการวางผังที่ซับซ้อนเท่านั่นแต่ยังรวมถึงการจัดทำการนำเสนอต่อลูกค้า การทำภาพเสมือนจริงของอาคารในเบื้องต้น การแก้ไขความเปลี่ยนแปลงของการตัดสินใจของลูกค้าในระหว่างออกแบบ อีกด้วย ดังนั่นเวลาเพียง3อาทิตย์ไม่เพียงพอต่อการออกแบบวางผังอาคารคลังสินค้าที่ซับซ้อนให้ได้ผลลัพธ์ที่ดีต่อการใช้งานและถูกต้องตามเงื่อนไขการวางผังได้



ภาพที่ 27 ภาพแสดงแผนงานช่วง Conceptual Desing ในเวลา 3 สัปดาห์

2.2.2.1 ขั้นตอนการทำงานกับเวลา

หากนำขั้นตอนการทำงานแบบปัจจุบันมาผนวกกับเวลาจะแสดงให้เห็นว่าเวลาเพียงสามอาทิตย์นั่นไม่สอดคล้องกับขั้นตอนการทำงานแบบปัจจุบันของสถาปนิก ดังนี้

สัปดาห์ที่ 1

ขั้นตอนที่ 1 การรับข้อมูลความต้องการที่เกี่ยวข้องกับการออกแบบและวางผังคลังสินค้า (Get Requirements)

ขั้นตอนที่ 2 การจัดกลุ่มวัตถุที่จัดเก็บ (Classification)

ขั้นตอนที่ 3 การหาขนาดของแต่ละกลุ่มพื้นที่ (Create space)

ขั้นตอนที่ 4 การจัดเรียงพื้นที่คลังสินค้าโดยมีผลลัพธ์การวางผังคลังสินค้าที่หลากหลายตามเงื่อนไขของวัตถุที่จัดเก็บ (Space planning and Alternative design)

สัปดาห์ที่ 2

ในการประชุมเพื่อนำเสนอแบบผังคลังสินค้าในสัปดาห์ที่ 2 นี้โดยปกติในสัปดาห์จะไม่สามารถสรุปแบบได้ เนื่องจากยังมีความต้องการเพิ่มเติมหรือเปลี่ยนแปลงจากเจ้าของโครงการอยู่ ทำให้ทางผู้ออกแบบต้องใช้เวลาในสัปดาห์ที่ 2 แก้แบบเพื่อใช้ประชุมในสัปดาห์ถัดไปด้วยการทำงานในขั้นตอนที่ 3 และ 4 ซ้ำอีกครั้ง

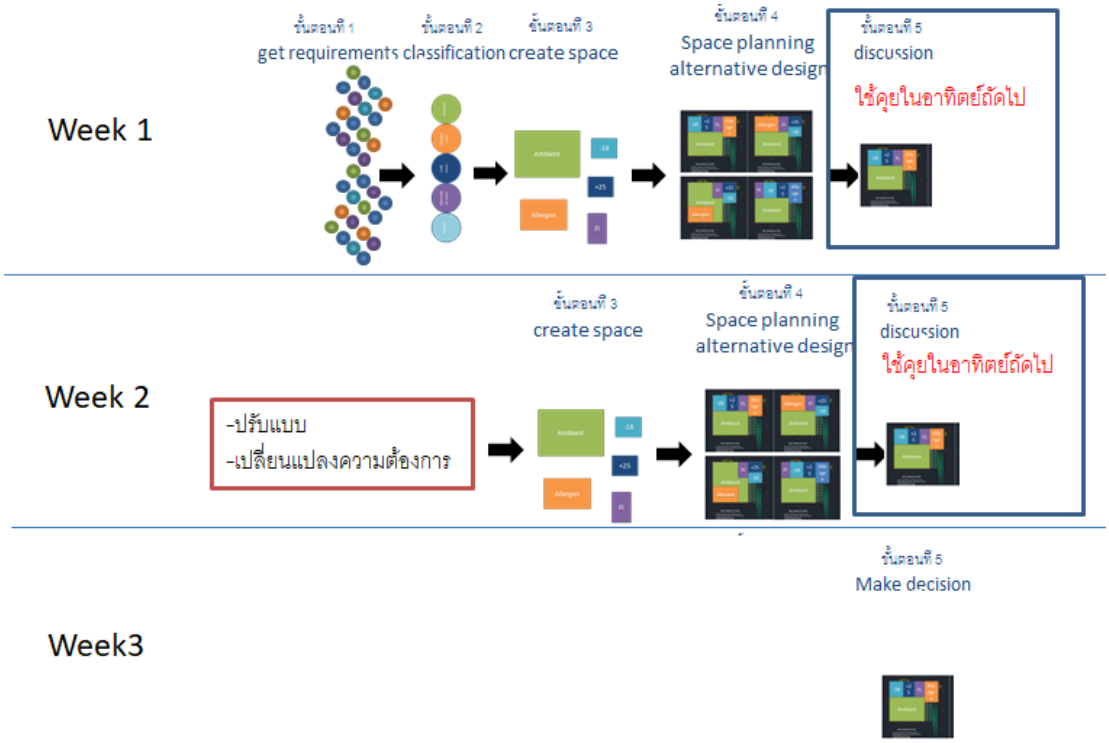
ขั้นตอนที่ 3 การหาขนาดของแต่ละกลุ่มพื้นที่ (Create space)

ขั้นตอนที่ 4 การจัดเรียงพื้นที่คลังสินค้าโดยมีผลลัพธ์การวางผังคลังสินค้าที่หลากหลายตามเงื่อนไขของวัตถุที่จัดเก็บ (Space planning and Alternative design)

สัปดาห์ที่ 3

ขั้นตอนที 5 การพิจารณาแบบที่เหมาะสมที่สุดกับการทำงานของโรงงานนั่นๆ (Discussion)

ดังนั่นจะเห็นได้ว่าสัปดาห์ที่ 1 และ 2 นั่นปริมาณการทำงานที่เยอะและซับซ้อนมากทำให้กระทบต่อ ความหลากหลายของแบบที่จะนำไปคุยกับลูกค้ารวมถึงความถูกต้องแม่นยำของเงื่อนไขในการวางผังคลังสินค้าด้วย



ภาพที่ 28 ขั้นตอนการทำงานแบบเดิมกับเวลา

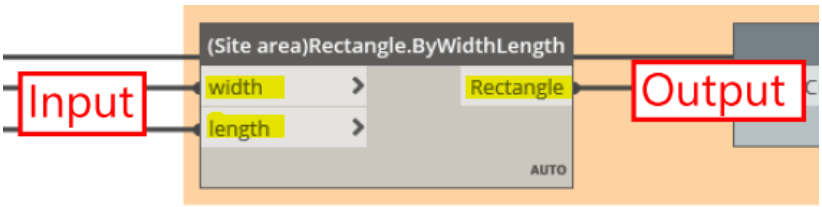
2.3 แนวคิดเกี่ยวกับการแก้ปัญหาโดยการวางผังด้วยคอมพิวเตอร์

ในปัจจุบันคอมพิวเตอร์เข้ามามีบทบาทในการเปลี่ยนแปลงวิธีการทำงานออกแบบทางด้านสถาปัตยกรรมเพื่อแก้ปัญหาและตอบสนองความต้องการของผู้ออกแบบที่มีความซับซ้อนขึ้นเรื่อยๆ เห็นได้จากลักษณะการทำงานสถาปัตยกรรมบนรูปแบบ2มิติ ก็ถูกพัฒนาสู่ระบบการทำงานรูปแบบใหม่ในระบบ 3มิติหรือที่เรียกกันว่า BIM เป็นการทำงานที่มีความซับซ้อนมากยิ่งขึ้นบวกกับเทคโนโลยีในปัจจุบันที่รองรับการประมวลผลเพื่อแก้ปัญหาที่ซับซ้อนและใช้เวลาได้มากยิ่งขึ้นกว่าในอดีตแบบก้าวกระโดด

การแก้ปัญหาการวางผังอาคารคลังสินค้า มีแนวคิดของการนำโปรแกรมคอมพิวเตอร์มาใช้ในการวางผัง โดยทำงานบนโปรแกรมที่ถูกสร้างเพื่อสอดคล้องกับระบบการทำงานปัจจุบัน ในกรณีของการศึกษานี้ผู้วิจัยทำการศึกษาโดยอ้างอิงจากการทำงานบนโปรแกรมที่ชื่อว่า Revit ซึ่งมีส่วนเสริมของโปรแกรมที่ชื่อว่า Dynamo ซึ่งเป็นส่วนที่เปิดให้ผู้ใช้สามารถเขียนโปรแกรมเพื่อเสริมการทำงานบนโปรแกรม Revit ได้ ซึ่งเป็นลักษณะของโปรแกรมมิ่งที่เรียกว่า Vision Programing นอกเหนือจากนั้นยังมีแนวคิดในการออกแบบที่เรียกว่า Generative Design ซึ่งเป็นการทำงานแบบ Machine-Learning-Assisted เพื่อนำมาเพิ่มประสิทธิภาพการทำงานของโปรแกรมที่เราออกแบบไว้ใน Dynamo โดยโปรแกรมในส่วนนี้จะชื่อว่า Refinery

2.3.1 Dynamo

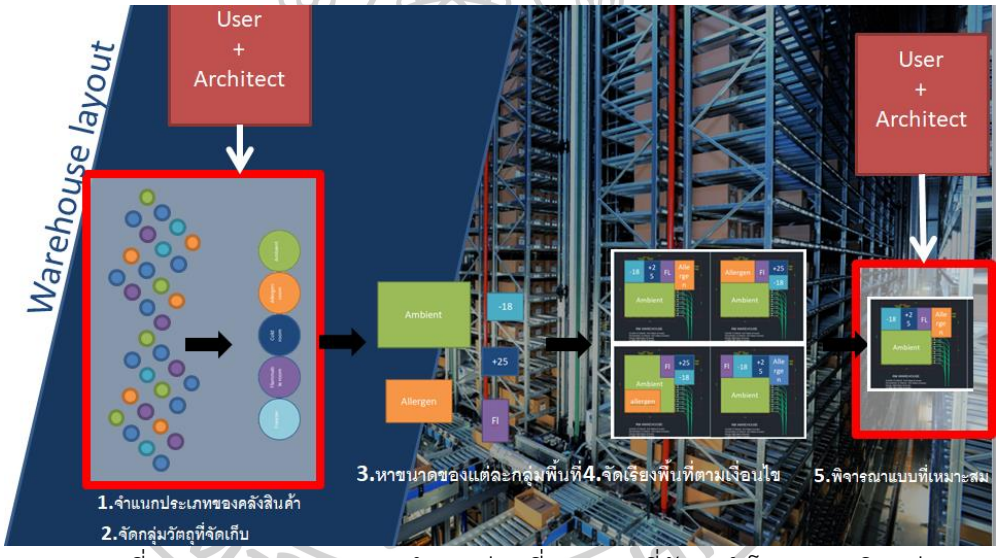
Dynamo เป็นเครื่องมือที่ช่วยในการพัฒนาโปรแกรมโดยมีลักษณะของโปรแกรมมิ่งที่เรียกว่า Vision Programing คือ การพัฒนาโปรแกรมที่ผู้เขียนโปรแกรมสามารถมองเห็นผลลัพธ์ของงาน ได้ตั้งแต่ขณะพัฒนาโปรแกรม โดยไม่จำเป็นต้องรอให้การพัฒนาเสร็จสมบูรณ์ โดยมีเตรียมสิ่งที่เรียกว่า สิ่งแวดล้อมในการทำงาน (Development Environment) และเครื่องมือ หรือชิ้นส่วนที่ผู้พัฒนาต้องใช้ในการสร้างโปรแกรมไว้ให้สามารถเรียกใช้งานได้ โดยที่ไม่ต้องลงมือสร้างเอง เครื่องมือหรือชิ้นส่วนที่ระบบเตรียมไว้ หรือผู้พัฒนาคนอื่นๆสร้างไว้ให้นี้เรียกว่า Node โดยผู้เขียนโปรแกรมสามารถค้นหาและเลือกใช้ Node ต่างเพื่อให้ตรงกับความต้องการของโปรแกรมที่กำลังออกแบบ โดยไม่ต้องเขียนโค้ดคำสั่งทางโปรแกรมมิ่งเอง เพียงแค่กำหนดตัวแปรเพิ่มเติมที่ต้องนำมาใช้และเชื่อมต่อ Node แต่ละ Node เข้าด้วยกัน โดยลักษณะการเขียนโปรแกรมของ Dynamo จะเป็นลักษณะการเชื่อมต่อ Nodeต่างๆเข้าด้วยกัน ซึ่งตัว Node นั่นๆจะมีการรับข้อมูลตัวแปร ที่เรียกว่า Input อยู่ที่ฝั่งซ้าย และการส่งออกข้อมูลตัวแปร ที่เรียกว่า Output อยู่ที่ฝั่งขวา โดยที่จำนวน Input และ Output จะมีมากว่าหนึ่งได้หรือบาง Node มีแค่ฝั่งใดฝั่งหนึ่งก็ได้ตามแต่ลักษณะการใช้งานของ Nodeนั่นๆ (Synergysoft, 2017)



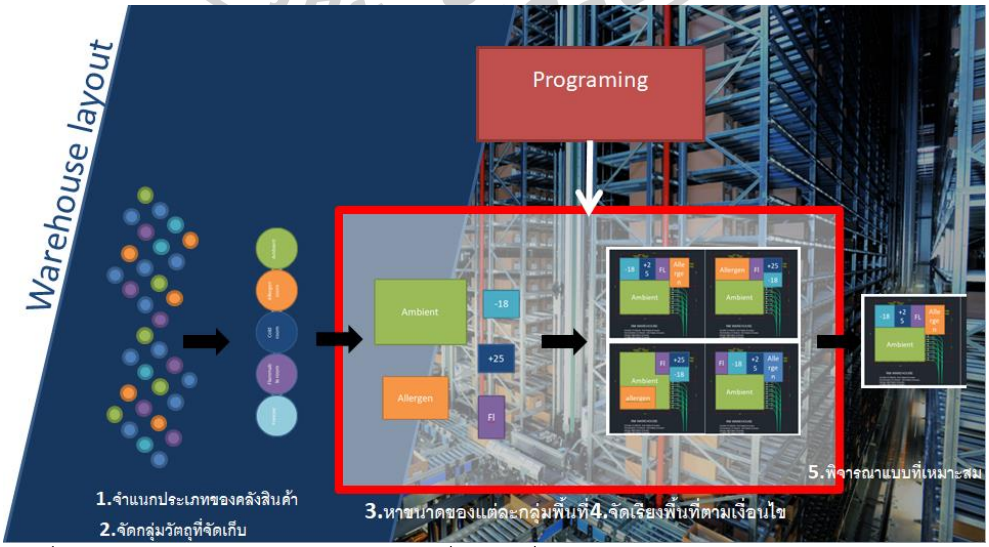
ภาพที่ 29 รูปตัวอย่างNode Rectangle.ByWidthLength เป็นNodeในการสร้างสี่เหลี่ยม มีInput คือ ความกว้างและความยาว ส่วนOutput จะได้สี่เหลี่ยมออกมา

2.3.2 Dynamo กับการวางผังคลังสินค้า

จากรูปแบบการวางผังคลังสินค้าแบบเดิม 5 ขั้นตอนที่กล่าวมาในบทที่ 2 หัวข้อ 2.1 ขั้นตอนและการวางผังในปัจจุบันขั้นตอนการทำงานในส่วนที่ 1 , 2 และ ขั้นตอนที่ 5 จะยังคงเป็นหน้าที่ของตัวสถาปนิกทำงานร่วมกับลูกค้า เนื่องจากขั้นตอนที่1และ2 เป็นส่วนของการรับเอาความต้องการของลูกค้าและผู้ใช้งานนำมาจำแนกและจัดกลุ่มความต้องการกำหนดการใช้งานในแต่ละพื้นที่ รวมถึงขั้นตอนที่ 5 ขั้นตอนในการพิจารณาตัดสินใจเลือกรูปแบบการผังที่เหมาะสมก็ยังคงเป็นการตัดสินของผู้ออกแบบร่วมกับลูกค้าและผู้ใช้งานแต่จะมีค่าคะแนนของผลลัพธ์ที่ช่วยในการประกอบการตัดสินใจ ส่วนในขั้นตอนที่ 3 การหาขนาดของแต่ละกลุ่มพื้นที่ และขั้นตอนที่ 4 การจัดเรียงพื้นที่หรือการวางผังพื้นที่ นั่นสามารถใช้โปรแกรมเพื่อช่วยในการออกแบบได้



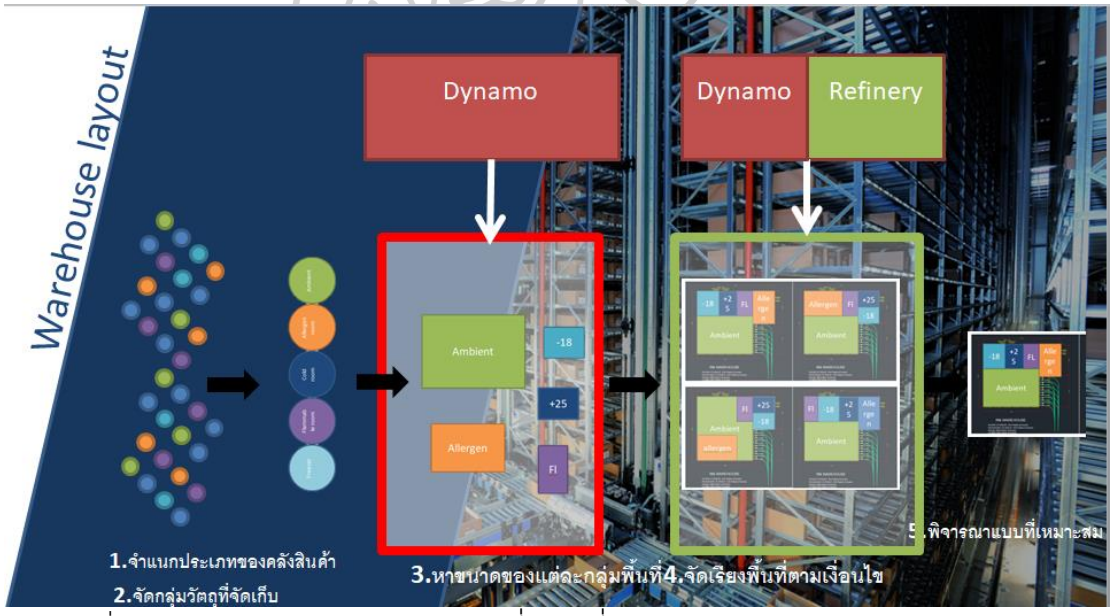
ภาพที่ 30 แผนภาพแสดงการทำงานส่วนที่ 1, 2, 5 ที่ยังคงทำโดยสถาปนิกอยู่



ภาพที่ 31 แผนภาพแสดงการทำงานส่วนที่ 3, 4 ที่ใช้โปรแกรมมาทดแทนการทำงานแบบเดิม

Dynamo จะถูกนำมาใช้เขียนโปรแกรมเป็นส่วนเสริมเพื่อคำนวนหาขนาดและสร้างพื้นที่ต่างๆตามความต้องการของลูกค้าโดยอัตโนมัติ ตรงกับขั้นตอนที่ 3 การหาขนาดพื้นที่ของคลังสินค้าแบบเดิม โดยอ้างอิงจากเนื้อหาในบทที่ 2 หัวข้อ 2.1.3.2 วิธีการคำนวณพื้นที่คลังสินค้า ซึ่งมีรายละเอียดและขั้นตอนที่ยุ่งยากและซับซ้อน ทำให้ผู้ออกแบบสามารถตัดขั้นตอนดังกล่าวออกไป โดยให้เป็นหน้าที่ของโปรแกรมวางผังคลังสินค้าเนื่องจากเป็นการคำนวณทางคณิตศาสตร์ คอมพิวเตอร์สามารถทำการคำนวนได้อย่างแม่นยำและรวดเร็วกว่ามนุษย์ รวมถึงยังสร้างพื้นที่ในแต่ละส่วนออกมาเพื่อเตรียมพร้อมสำหรับการนำไปจัดเรียงหรือวางผังคลังสินค้าในขั้นตอนที่4 การจัดเรียงพื้นหรือวางผังคลังสินค้า ต่อไป

ในส่วนของขั้นตอนที่ 4 การจัดเรียงพื้นหรือวางผังคลังสินค้า นั่นจะมีการทำงานรวมกันของ 2 โปรแกรม คือทั้ง Dynamo และ Refinery โดย Dynamo นั่นจะทำหน้าที่ในการเขียนโปรแกรมเพื่อสั่งงานในการนำพื้นที่ภายในคลังสินค้าพื้นที่ต่างๆที่ถูกสร้างขึ้นในขั้นตอนที่ 3 นำมาจัดเรียงภายใต้ขอบเขตของพื้นที่อาคารที่สร้างได้

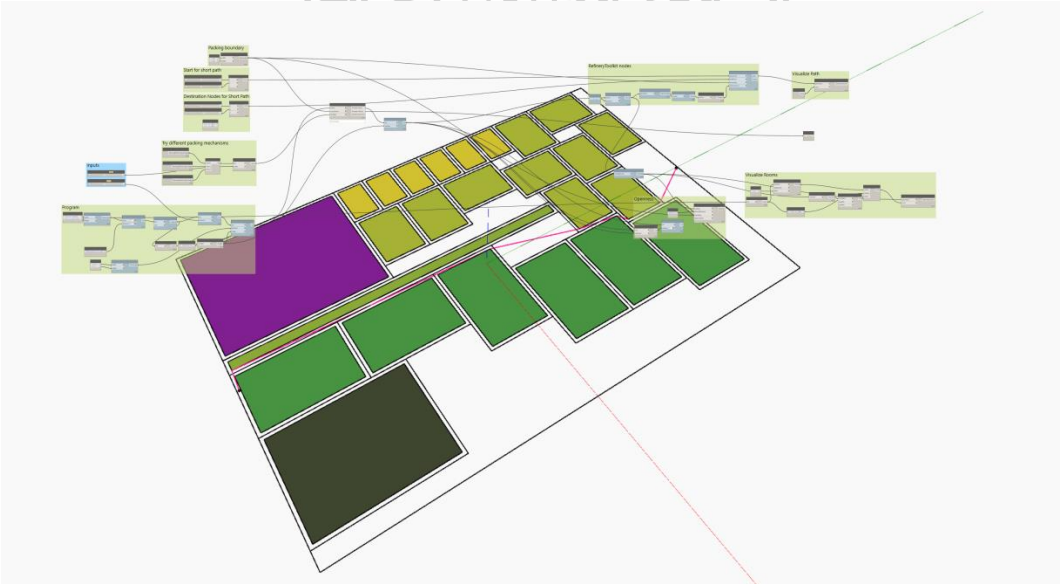


ภาพที่ 32 แผนภาพแสดงการทำงานส่วนที่ 3, 4 ที่ใช้โปรแกรม Dynamo และ Refinery มาทดแทนการทำงานแบบเดิม

การจัดเรียงพื้นที่หรือการวางผังคลังสินค้าโดยมีหลักการทำงานที่เรียกว่า Container บวกกับพื้นฐานทางคณิตศาสตร์เรื่องความน่าจะเป็น (Probability) โดยเลียนแบบการบรรจุสิ่งของลงกล้อง ในที่นี้สิ่งของคือพื้นที่เก็บสินค้าปรเภทพต่างๆที่มีความแตกต่างกันซึ่งถูกจำแนก

และสร้างไว้แล้ว นำมาบรรจุคือจัดเรียงลงใน ขอบเขตของอาคารที่กำหนดขึ้นซึ่งเปรียบเสมือนกล่อง โดยการจัดเรียงหรือการวางผังนี้สามารถจัดได้หลากหลายรูปแบบตามแต่ละรูปแบบในการจัด ยกตัวอย่างเช่น จัดของ หรือ พื้นที่ชิ้นไหนใส่ก่อน ใส่หลัง หรือ นำด้านยาวของพื้นที่เรียงติดกันก่อน หรือ ด้านกว้างของพื้นเรียงติดกันก่อน รูปแบบการจัดเหล่านี้ก็จะทำให้ความหลากหลายของรูปแบบจำนวนมากตามมา

รูปแบบการจัดเรียงที่อาจเกิดขึ้นได้เป็นจำนวนมากนั่นไม่สามารถคัดกรองและพิจารณาเลือกนำมาใช้ได้ด้วยความสามารถของมนุษย์ในเวลาที่จำกัดทำให้ การออกแบบวางผังคลังสินค้าต้องมีการกำหนดเป้าหมายของผลลัพธ์ที่ต้องการให้เกิดขึ้น เพื่อเป็นการคัดกรองเพื่อให้ตรงกับความต้องการของผู้ออกแบบและลูกค้า ซึ่งจะตรงกับกระบวนการทำงานที่เรียกว่า Generative Design ซึ่งจะใช้โปรแกรม Refinery เป็นขั้นตอนถัดไปของการนำโปรแกรมคอมพิวเตอร์มาช่วยในการวางผังคลังสินค้า (Das, 2016)



ภาพที่ 33 ภาพแสดงการวางผังคลังสินค้าโดยหลักการ Container (Das, 2016)

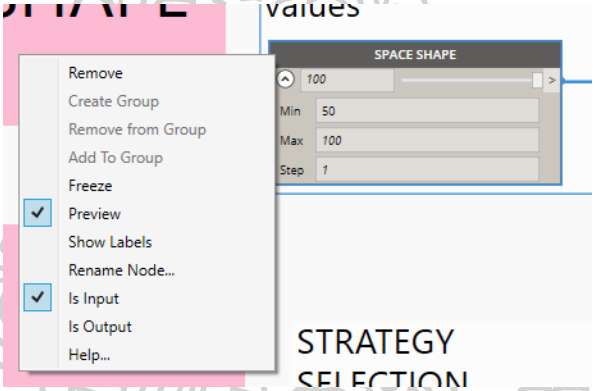
2.3.3 Refinery

Refinery เป็นโปรแกรมที่มีช่วยทำงานเพื่อรองรับแนวคิดในการออกแบบที่เรียกว่า Generative Design ซึ่งเป็นการทำงานแบบ Machine-Learning-Assisted เพื่อนำมาเพิ่มประสิทธิภาพการทำงานของโปรแกรมที่เราออกแบบไว้ใน Dynamo

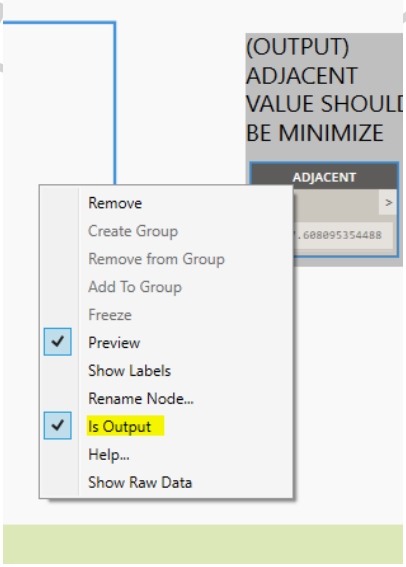
Generative Design คือการออกแบบ Machine-Learning-Assisted เป็นการนำมาใช้เพื่อเพิ่มประสิทธิภาพการออกแบบตามชุดพารามิเตอร์หรือค่าตัวแปรที่แตกต่างกันซึ่งกำหนด

โดยผู้ใช้ โดยวิธีการศึกษาโซลูชั่นการออกแบบตามปัจจัยภายนอกที่แตกต่างกันโดยคอมพิวเตอร์เพื่อสร้างผลลัพธ์ของงานออกแบบที่หลากหลาย ซึ่งจะมุ่งเน้นไปที่การปรับปรุงการออกแบบที่มีอยู่ โดยสามารถกำหนดเป้าหมายของผลลัพธ์ที่ออกมาได้ เพื่อให้มั่นใจได้ว่าจะไม่เกิดการออกแบบที่ดูดีแต่ใช้งานไม่ได้ (Synergysoft, 2017)

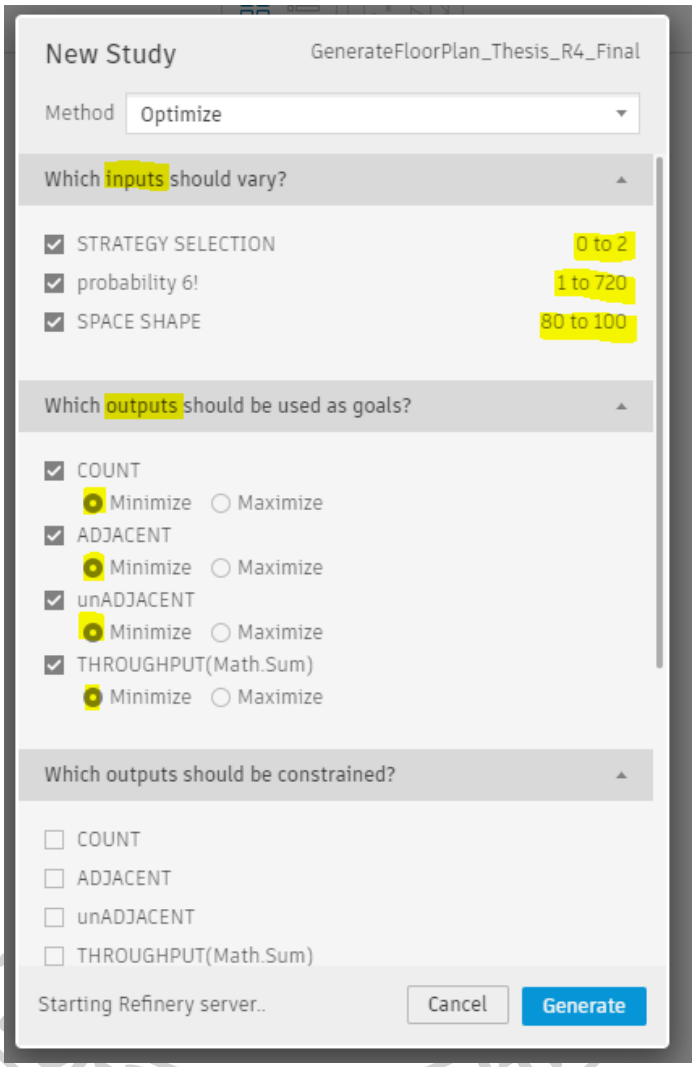
Refinery จะทำงานร่วมกับ Dynamo เพื่อเพิ่มประสิทธิภาพให้กับโปรแกรมที่ออกแบบไว้บนDynamo โดยการศึกษาค่าตัวแปรต่างๆที่เป็นปัจจัยที่ส่งผลต่อผลลัพธ์ที่จะเกิดขึ้นเพื่อนำไปสร้างเป็นผลลัพธ์ที่มีความหลากหลายซึ่งค่าตัวแปรดังกล่าวจะถูกกำหนดที่ Nodeนั่นๆ บน Dynamo ให้เป็น Node ที่เรียกว่า Input โดยผู้ออกแบบสามารถกำหนดเป้าหมายของผลลัพธ์ที่จะเกิดขึ้นได้ โดยเป็นการกำหนดค่าต่างๆของผลลัพธ์ที่เกิดขึ้น ซึ่งค่าของผลลัพธ์ดังกล่าวจะถูกกำหนดที่Nodeนั่นๆ บน Dynamo ให้เป็น Node ที่เรียกว่า Output



ภาพที่ 34 ภาพแสดงการเลือก Node ที่เป็น Is Input



ภาพที่ 35 ภาพแสดงการเลือก Node ที่เป็น Is Output



ภาพที่ 36 ภาพแสดงผลลัพธ์ของ Input และ Output ในโปรแกรม Refinery โดยผู้ใช้สามารถกำหนดผลลัพธ์ที่อยากให้เป็นได้

2.3.4 Refinery กับการวางผังคลังสินค้า

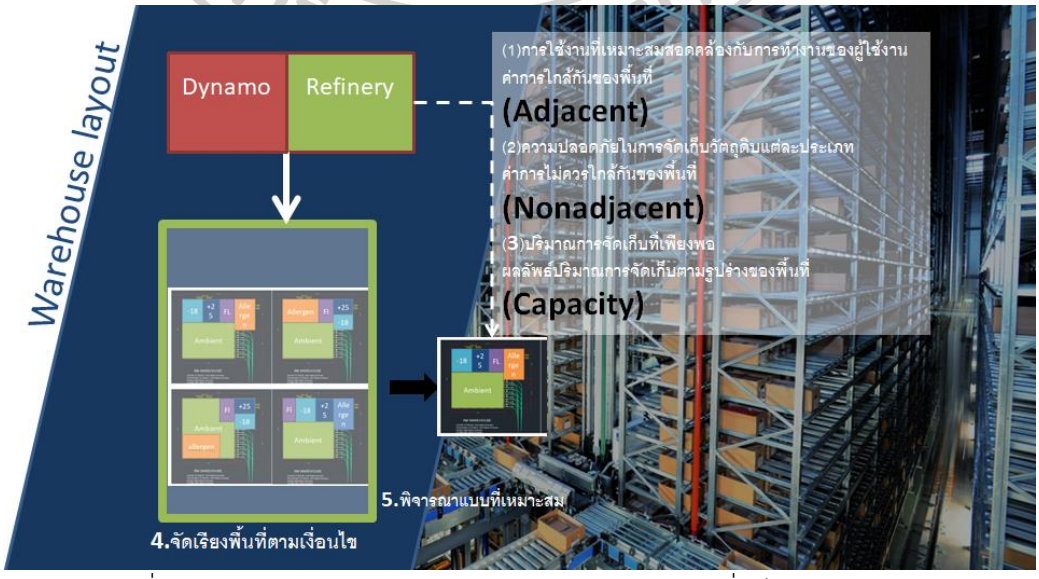
การสร้างผลลัพธ์ของการวางผังจำนวนมากภายใต้เป้าหมายที่ผู้ออกแบบสามารถกำหนดได้ จะทำให้ได้ผลลัพธ์ที่ถูกต้องและตรงตามความต้องการของผู้ออกแบบและลูกค้าโดยผู้ออกแบบสามารถกำหนดค่าของผลลัพธ์ใน Dynamo เพื่อใช้ในการกำหนดเป้าหมายของผลลัพธ์การวางผังได้ โดยค่าของผลลัพธ์จะอ้างอิงจากสิ่งที่จะช่วยผู้ออกแบบและลูกค้าในการพิจารณาเลือกรูปแบบการวางผังที่เหมาะสม

หลักการในการเลือกพิจารณาแบบที่เหมาะอ้างอิงจากบทที่ 2 ข้อที่ 2.1.5.1 หลักเกณฑ์การพิจารณาเลือกแบบอาคารคลังสินค้าสามารถสรุปได้เป็น 3 ส่วนดังนี้

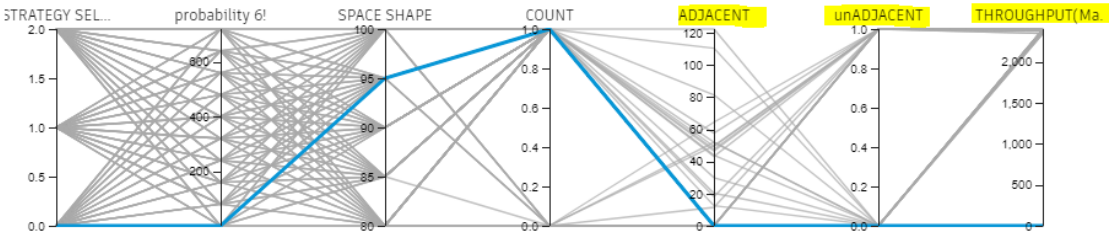
(1) การใช้งานที่เหมาะสมสอดคล้องกับการทำงานของผู้ใช้งาน สามารถออกแบบให้ผลลัพธ์มีหลักการ ค่าการใกล้กันของพื้นที่ (Adjacent) ที่ควรใกล้ตามที่ผู้ออกแบบได้กำหนดการใช้งานห้องหรือพื้นที่ต่างๆในคลังสินค้าไว้จากการวิเคราะห์จากหลักการวิธีการทำงานและความต้องการจากผู้ใช้งานจริง โดยค่าของผลลัพธ์คือค่าระยะห่างระหว่างพื้นที่ที่ผู้ออกแบบกำหนดไว้ โดยพื้นที่ที่ควรใกล้กันมีค่าของระยะห่างระหว่างกันยิ่งน้อยยิ่งดี

(2) ความปลอดภัยในการจัดเก็บวัตถุดิบแต่ละประเภท สามารถออกแบบให้ผลลัพธ์มีหลักการ ค่าการไม่ควรใกล้กันของพื้นที่ (Nonadjacent) ที่ไม่ควรใกล้ตามที่ผู้ออกแบบได้กำหนดการใช้งานห้องหรือพื้นที่ต่างๆในคลังสินค้าไว้จากการวิเคราะห์จากหลักการวิธีการจัดเก็บวัตถุดิบที่ต้องคำนึงถึงความปลอดภัย โดยค่าของผลลัพธ์คือค่าระยะห่างระหว่างพื้นที่ที่ผู้ออกแบบกำหนดไว้ โดยพื้นที่ที่ไม่ควรใกล้กันไม่ควรติดกัน กล่าวคือไม่ควรมีค่าของระยะห่างระหว่างกันเท่ากับ 0

(3) ปริมาณการจัดเก็บที่เพียงพอ โดยหลักการแล้วการคำนวนเพื่อสร้างพื้นที่การจัดเก็บของสินค้าประเภทต่างๆนั่นคำนวนจากจำนวนการจัดเก็บที่เป็นความต้องการของลูกค้าซึ่งในขั้นตอนที่ 3 การหาขนาดของพื้นที่โดยการใช้โปรแกรมในการคำนวนพื้นที่ทำให้พื้นที่ที่ได้ตรงกับความต้องการของลูกค้าอยู่แล้วแต่มีอีกกรณีคือความต้องการของลูกค้าเมื่อทำการคำนวณแล้วเกินกว่าขนาดของคลังสินค้าที่จะรับได้ ทำให้ผู้ออกแบบต้องคำนวณเพื่อนำเสนอผลลัพธ์ของปริมาณการจัดเก็บที่แท้จริงโดยสอดคล้องกับขนาดของพื้นที่จริงที่จัดเก็บได้ โดยปริมาณการจัดเก็บ (Capacity) นี้จะมีผลอย่างมากในการพิจารณาเลือกแบบการวางผังคลังสินค้าของลูกค้า



ภาพที่ 37 ภาพแสดงลูกศรเส้นประแสดงถึงค่าของผลลัพธ์ที่จะได้จาก Refinery



ภาพที่ 38 ตัวอย่างแผนภูมิค่าความสัมพันธ์ของ Input และ Output ที่ออกมาอย่างหลากหลาย

ภาพที่ 39 ตัวอย่างผลลัพธ์ที่ออกมาอย่างหลากหลายโดย Refinery

บทที่ 3
วิธีการวิจัย

3.1 แนวความคิดของโปรแกรมวางผังคลังสินค้า (Warehouse Planning Program)

การลดภาระงานที่ซับซ้อนและมีขั้นตอนการทำงานที่ถูกทำซ้ำไปมานั่นเป็นแนวความคิดหลักของกนำเอาโปรแกรมมิ่งมาช่วยในการทำงาน เช่นเดียวกันกับงานวิจัยชิ้นนี้ที่ต้องการลดภาระและปัญหาที่เกิดขึ้นจากขั้นตอนการวางผังคลังสินค้าแบบเดิมที่มีเงื่อนไขในการจัดวางผังที่ซับซ้อนและใช้เวลาในการจัดวางผังนานซึ่งไม่สอดคล้องกับระยะเวลาที่มีจำกัดในการทำงานจริง ส่งผลให้เกิดข้อผิดพลาดที่เกิดจากมนุษย์ได้ (Human Error) เนื่องจากขั้นตอนการวางผังนี้เป็นขั้นตอนซึ่งถือว่าเป็นขั้นต้นของการเริ่มงานทางสถาปัตยกรรม การจัดวางผังเป็นเสมือนรากฐานที่มีผลบาทในการกำหนดรูปแบบการใช้งานหลักของอาคาร หากขั้นตอนดังกล่าวเกิดความผิดพลาดผลลัพธ์สุดท้ายจะเกิดเป็นงานสถาปัตยกรรมที่มีปัญหาทั้งการใช้งานรวมถึงในกรณีที่อาจเกี่ยวเนื่องไปถึงความปลอดภัยของผู้ใช้งานและอาคารในอนาคตด้วยดังที่เราอาจจะได้รับทราบกรณีที่เกิดเพลิงไหม้จากการจัดเก็บสารเคมีที่ไวไฟอย่างไม่ถูกต้องจนทำให้เกิดเพลิงไหม้เป็นต้น

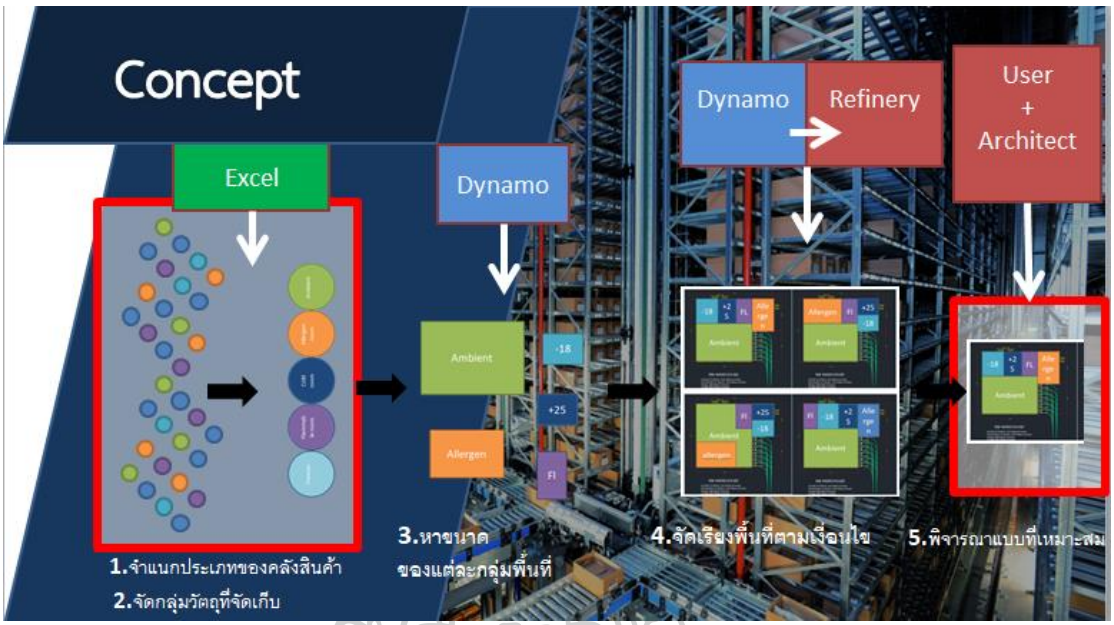
ดังนั่นแนวคิดของโปรแกรมวางผังคลังสินค้าคือการนำโปรแกรมคอมพิวเตอร์มาทดแทนขั้นตอนการวางผังคลังสินค้าในปัจจุบันโดยสถาปนิก ดังนี้

ขั้นตอนการจัดวางผังคลังสินค้าแบบเดิมที่ทำอยู่ อ้างอิงจากการทำงานของบริษัทที่ผู้วิจัยทำงานอยู่ ซึ่งมีขั้นตอนหลักดังนี้

1) ใช้ Excel สร้างโปรแกรมในขั้นการรับข้อมูลความต้องการของลูกค้าที่มีผลต่อการออกแบบวางผังคลังสินค้า (Get Requirements)

2) ใช้ Dynamo สร้างโปรแกรมในขั้นตอนการสร้างพื้นที่ (Create Space) และขั้นตอนการวางผัง (Space Planning)

3) ใช้ Refinery สร้างโปรแกรมในการสร้างและกำหนดผลลัพธ์การวางผังคลังสินค้าที่หลากหลายเพื่อนำไปสู่การพิจารณาและเลือกผลลัพธ์สุดท้ายของผังคลังสินค้าที่เหมาะสมที่สุด (Space Planning and Alternative Design)



ภาพที่ 40 แผนภาพแสดงแนวความคิดโปรแกรมวางผังคลังสินค้า

3.1.1 ใช้ Excel ในขั้นการรับข้อมูลความต้องการของลูกค้าที่มีผลต่อการออกแบบวางผังคลังสินค้า

ข้อมูลที่ความต้องการของลูกค้าที่มีผลต่อการออกแบบวางผังคลังสินค้าในขอบเขตที่ผู้วิจัยทำการศึกษานั้นสามารถกำหนด ข้อมูลที่ต้องการจากลูกค้าเพื่อใช้ในการออกแบบและวางคลังสินค้าออกมาในรูปแบบของการกรอกในตารางได้ โดยการแบ่งความต้องการเป็นสองกลุ่มใหญ่ คือ กลุ่มที่ 1 ข้อมูลความต้องการที่มีผลต่อการสร้างพื้นที่เพื่อการออกแบบวางผัง กลุ่มที่ 2 ข้อมูลความต้องการที่เป็นเงื่อนไขในการจัดวางผังซึ่งจะมีผลต่อการกำหนดเป้าหมายที่จะเกิดขึ้นของผลลัพธ์ของการจัดวางผังคลังสินค้า

3.1.1.1 ข้อมูลกลุ่มที่ 1

สามารถกำหนดโดยตรงจากข้อมูลที่ได้รับจากลูกค้า โดยมีหัวข้อดังนี้

(1) พื้นที่ขอบเขตที่ดิน (Area)

(2) จำนวนของรถชนิดต่างๆที่ถูกใช้ในการขนส่ง (No. of Each Car Type)

-รถ Pick up 4 ล้อ

-รถ Truck 6 ล้อ

-รถ Trailer 40 Feet

(3) รายชื่อพื้นที่จัดเก็บสินค้าแต่ละประเภท (Name Space)

(4) จำนวนความต้องการจัดเก็บสินค้าในแต่ละประเภท (Capacity)

(5) จำนวนการซ้อนชั้นของการจัดเก็บสินค้าแต่ละประเภท (Level)

(6) กำหนดสีเพื่อแบ่งที่การจัดเก็บสินค้าแต่ละประเภท (Color)

3.1.1.2 ข้อมูลกลุ่มที่2

สถาปนิกมีหน้าต้องวิเคราะห์ข้อมูลที่ได้รับจากลูกค้าในเรื่องของคุณสมบัติของสินค้าที่จะถูกจัดเก็บในแต่ละพื้นที่ก่อน จากนั้นจึงกำหนดคุณลักษณะของแต่ละพื้นที่ภายใต้เงื่อนไขในการจัดวางพื้นที่แต่ละพื้นที่โดยมีหัวข้อดังนี้

(1) กำหนดพื้นที่ที่ควรอยู่ใกล้กัน (Adjacent Space)

(2) กำหนดพื้นที่ต้องอยู่ไกลกัน (Nonadjacent Space)

3.1.2 ใช้ Dynamo ในขั้นตอนการสร้างพื้นที่ (Create Space) และขั้นตอนการวางผัง (Space Planning)

เนื่องจากการสร้างพื้นที่จัดเก็บสินค้าในแต่ละประเภทนั่นสามารถคำนวนได้จากข้อมูลที่ได้รับจากลูกค้าที่ถูกกรอกลงใน Excel ทำให้สามารถสร้าง พื้นที่ที่มีผลต่อการวางผังได้ดังนี้

3.1.2.1 แนวความคิดในขั้นตอนการสร้างพื้นที่ (Create Space)

พื้นที่ดังต่อไปนี้สามารถ ถูกสร้างได้อัตโนมัติจากโปรแกรมสถาปนิกไม่จำเป็นต้องเสียเวลาในการคำนวนส่วนนี้อีกต่อไป โดยมีความสัมพันธ์ในการนำข้อมูลความต้องการของลูกค้าที่ถูกกรอกลงใน Excel มาใช้ในขั้นตอนการสร้างพื้นที่ (Create Space) ด้วยโปรแกรมโดยมีหลักการดังต่อไปนี้

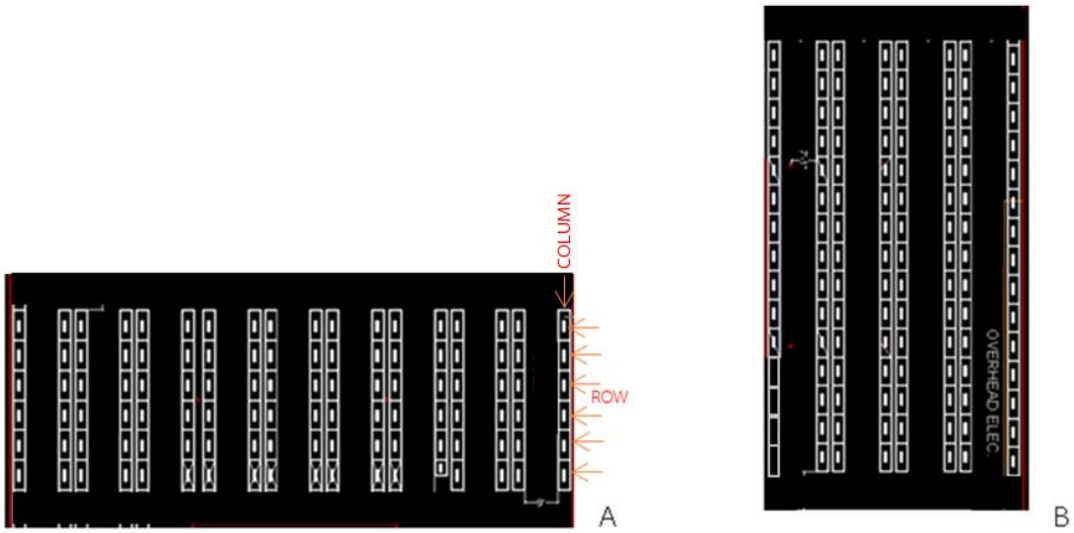
(1) หลักการหาขนาดของพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space)

<u>ขั้นตอนที่ 1</u> คำนวนหาจำนวนชั้นวางสินค้าต่อ 1 พาเลท (Locations) จาก ความต้องการในการจัดเก็บสินค้า (Capacity) และ จำนวนชั้นในการจัดเก็บ (Level) ที่ได้จากความต้องการของลูกค้าที่ถูกกรอกใน Excel

<u>Capacity ÷ Level = จำนวนชั้นวางสินค้าต่อ1 พาเลท (Locations)</u>

<u>ขั้นตอนที่ 2</u> การหารูปทรงของพื้นที่จัดเก็บสินค้า โดยพื้นที่จัดเก็บสินค้าชนิดเดียวกัน ที่สร้างจากความต้องการในการจัดเก็บสินค้า (Capacity) ในข้อ (1) จะมีความแตกต่างกันของรูปทรงโดยมีปัจจัยเรื่องของการจัดวางจำนวน แถว (Row) และ แนว (Column) ของชั้นวาง

สินค้า (Rack) ดังภาพตัวอย่างด้านล่างภาพพื้นที่จัดเก็บสินค้า A มีจำนวนแถว (Row) และ แนว (Column) และรูปแบบการวางชั้นวางสินค้า (Rack) ที่แตกต่างจาก พื้นที่จัดเก็บสินค้า B ในขณะที่มีจำนวนความต้องการในการจัดเก็บสินค้า (Capacity) เท่ากันแต่ทรงผลให้รูปทรงของ พื้นที่จัดเก็บสินค้า A และ B แตกต่างกัน



ภาพที่ 41 ภาพแสดงตำแหน่ง Row และ Column ของชั้นวางสินค้าในแปลน (Mecalux, 2018)

จำนวนแถว (Row) และ แนว (Column) ในโปรแกรมที่ออกแบบจะคำนวณโดยการสมมติค่าของจำนวนแนว (Column) ขึ้นมาก่อนโดยสมมุติให้จำนวนแถว (Row) และแนว (Column) มีจำนวนเท่ากัน โดยการหาค่าสแควรูท (Square Root) ของจำนวนชั้นวางสินค้า 1 พาเลท (Locations) ที่ได้จากขั้นตอนที่ 1

√ Locations = ค่าสมมติของจำนวนแนว (Column)

เพื่อการสร้างรูปทรงของพื้นที่ที่แตกต่างกันจึงเพิ่มเติมโปรแกรมให้สามารถปรับค่าของจำนวนแนว (Column) ได้ โดยการคูณเปอร์เซ็นเข้าไปตั้งแต่ 50%-100% โดยค่าเปอร์เซ็นจะทำให้ค่าของจำนวนแนว (Column) นั่นเท่าเดิมในกรณีที่เป็น 100% และลดลงจนเหลือครึ่งนึงของค่าสมมติของจำนวนแนว (Column) เดิม เนื่องจากว่าโดยหลักการการวางชั้นวางสินค้านั่นจำนวนของแนว (Column) จะมีจำนวนมากกว่าจำนวนแถว (Row) โดยมีนัยยะสำคัญเพราะว่าจำนวนของแนว (Column) นั่นจะมีผลต่อช่องทางสัญจรของรถยก (Forklift) ซึ่งหากจำนวนของแนว (Column) มีมากก็จะทำให้พื้นเสียไปกับพื้นที่ช่องทางสัญจรของรถยก (Forklift) มากขึ้นตามไปด้วย

ในส่วนของค่าต่ำสุดที่ครึ่งนึงหรือ50%ของค่าสมมติของจำนวนแนว (Column) เพื่อกำหนดสัดส่วนของพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) ให้ด้านกว้างต่อด้าน

ยาวไม่ต่ำกว่า 1:4 เนื่องมาจากหากจำนวนของแนว (Column) มีค่าต่ำเกินไปจะทำให้ค่าจำนวนแถว (Row) นั่นมีค่ามากซึ่งจะมีผลต่อรูปร่างของพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) ที่จะมีลักษณะผอมและยาวซึ่งปกติจะไม่พบการออกแบบพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) ลักษณะนี้ รวมทั้งยังไม่สอดคล้องกับการใช้งานอีกด้วย

(100% - 50%) x √ Locations = ค่าสมมติของจำนวนแนว (Column)

ขั้นตอนที่ 3 หาจำนวนแถว (Row) โดยใช้ค่าสมมติของจำนวนแนว (Column) หารกับจำนวนชั้นวางสินค้าต่อ 1 พาเลท (Locations)

Locations ÷ Column = จำนวนแถว (Row)

ขั้นตอนที่ 4 หากว้าง (Width) ของพื้นที่จัดเก็บสินค้าแต่ละชนิด (room space) เมื่อได้ค่าจำนวนของแนว (Column) และจำนวนแถว (Row) มาแล้วจะสามารถหาค่าความกว้างและความยาวเพื่อกำหนดพื้นที่จัดเก็บสินค้าแต่ละชนิด (room space) ได้

การคำนวณหาความกว้าง (Width) ได้จากจำนวนแถว (Row) คูณกับความกว้างต่อ1พาเลท (Pallet Size) บนชั้นวางสินค้า (Rack) โดยใช้ค่า 1.3เมตร ซึ่งเป็นขนาดที่ใหญ่สุดครอบคลุมทุกขนาดตามมาตรฐานขนาดชั้นวางสินค้า (Rack) โดยอ้างอิงจากบทที่2 ข้อ2.1.3.2 วิธีการคำนวณพื้นที่คลัง

Row x Pallet Size (1.3 เมตร) = ความกว้าง (Width)

ขั้นตอนที่ 5 หาความยาว (Length) ของพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) โดยมี 2 ปัจจัยที่มีผลต่อความยาว (Length) คือขนาดความยาวต่อ1พาเลท (Pallet Size) บนชั้นวางสินค้า (Rack) และขนาดของทางสัญจรของรถยก (Forklift Lend)

ความยาวต่อ 1 พาเลท (Pallet Size) บนชั้นวางสินค้า (Rack) ใช้ค่า 1.3 ม. ซึ่งเป็นขนาดที่ใหญ่สุดครอบคลุมทุกขนาดตามมาตรฐานขนาดชั้นวางสินค้า (Rack) โดยอ้างอิงจากบทที่ 2 ข้อ 2.1.3.2 วิธีการคำนวณพื้นที่คลัง

ความกว้างของทางสัญจรของรถยก (Forklift Lend) ปกติสำหรับทางสัญจรรถยก (Forklift Lend) ทั่วไปจะอยู่ที่ประมาณ 4.5 เมตร รถยก (Forklift) ไฟฟ้าขนาดเล็กอาจลงไปได้ถึง 3 เมตรจากประสบการณ์ของผู้วิจัยซึ่งเป็นสถาปนิกออกแบบงานคลังสินค้า โครงการต่างๆ ที่ได้ออกแบบจะมีค่าเฉลี่ยของความกว้างทางสัญจรรถยก (Forklift Lend) อยู่ที่ 3.5 เมตร ซึ่งครอบคลุมการทำงานคลังสินค้าส่วนใหญ่ในประเทศไทย

ขนาดของทางสัญจรของรถยก (Forklift Lend) จะคำนวณได้จำนวนของแนว (Column) หาร 2 เนื่องจากทางสัญจรของรถยก (Forklift Lend) จะมีทุกๆ 2 แนว (Column) เพื่อใช้งานในการจัดเก็บสินค้า โดยการหารที่เกิดเศษจะถูกปัดขึ้น

Column ÷ 2 = ขนาดของทางสัญจรของรถยก (Forklift Lend)

ดังนั่นความยาว (Length) ของพื้นที่จัดเก็บสินค้าแต่ละชนิด (room space) คำนวณได้จากขนาดความยาวต่อ1พาเลท (Pallet Size) บนชั้นวางสินค้า (Rack) บวกกับขนาดของทางสัญจรของรถยก (Forklift Lend)

[Column x Pallet Size (1.3 เมตร)] + Forklift Lend = ความยาว (Length)

ขั้นตอนที่ 6 หาขนาดของพื้นที่จัดเก็บสินค้าแต่ละชนิด (room space) ได้จากผลคูณของความกว้าง (Width) และความยาว (Length) ที่ได้จาก ขั้นตอนที่ 4 และ 5 โดยมีการเผื่อทางเดินบวกเพิ่มอีก 30 เปอร์เซ็นของพื้นที่

[Width x Length] x 1.3 = พื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space)

(2) หลักการหาขนาดของขอบเขตอาคาร (Building Space)

คำนวนได้จากพื้นที่ขอบเขตที่ดิน (Area) ลบกับระยะร่นตามกฎหมาย (Setback) โดยจะออกแบบให้สามารถกรอกระยะร่นได้เองโดยผู้ใช้งานตามแต่กฎหมายกำหนดซึ่งมีความแตกต่างกันในแต่ละพื้นที่และรูปแบบของอาคาร

ขอบเขตที่ดิน (Area) – ระยะร่นตามกฎหมาย (Setback) = ขอบเขตอาคาร (Building space)

(3) หลักการหาขนาดของพื้นที่ในการบริหารจัดการการลานโหลดสินค้า (Loading Area)

คำนวนได้จากจำนวนรถในแต่ละประเภทที่ใช้ขนถ่ายสินค้า (No. of Each Car Type) โดยมีการแบ่งรถเป็น3ประเภทอ้างอิงจาก บทที่2 ข้อ 2.1.3.1 การหาขนาดพื้นที่ขนส่งสินค้า (Loading Area) โดยมีขนาดพื้นที่ในการจอดรวมระยะวงเลี้ยวตามแต่ละประเภท (Size of each car type) ดังนี้

-รถ Pick up 4 ล้อ

โดยต้องการพื้นที่ในการจอดรวมระยะวงเลี้ยวเท่ากับ 12 x 4 ตร.ม.

-รถ Truck 6 ล้อ

โดยต้องการพื้นที่ในการจอดรวมระยะวงเลี้ยวเท่ากับ 15 x 4 ตร.ม.

-รถ Trailer 40 Feet

โดยต้องการพื้นที่ในการจอดรวมระยะวงเลี้ยวเท่ากับ 30 x 4 ตร.ม.

ดังนั้นขนาดพื้นที่ในการบริหารจัดการการขนถ่านสินค้า (Loading Area) มีวิธีการคำนวณดังนี้

No. of Each Car Type x Size of Each Car Type = พื้นที่ขนส่งสินค้า (Loading Area)

3.1.2.2 แนวความคิดในขั้นตอนการวางผัง (Space Planning)

พื้นที่ในข้อ 3.1.2.1 ถูกสร้างได้อัตโนมัติจากโปรแกรมสถาปนิกไม่จำเป็นต้องเสียเวลาในการคำนวนส่วนนี้อีกต่อไป เช่นกันกับวางผังเมื่อพื้นที่ต่างๆถูกสร้างขึ้นครบถ้วน การจัดเรียงหรือการวางผังพื้นที่ก็สามารถถูกทำโดยโปรแกรมได้เช่นเดียวกัน โดยหลักการที่เรียกว่า การบรรจุสิ่งของลง (Packing) บวกกับพื้นฐานทางคณิตศาสตร์เรื่องความน่าจะเป็น (Probability) โดยเลียนแบบการบรรจุสิ่งของ (Items) ลงกล่อง (Container) ในระนาบ 2 มิติ ในที่นี้สิ่งของ (Items) คือพื้นที่เก็บสินค้าประเภทต่างๆที่มีความแตกต่างกันซึ่งถูกจำแนกและสร้างไว้แล้ว นำมาบรรจุคือจัดเรียงลงใน ขอบเขตของอาคารที่กำหนดขึ้นซึ่งเปรียบเสมือนกล่อง (Container) โดยการจัดเรียงหรือการวางผังนี้สามารถจัดได้หลากหลายรูปแบบตามแต่ละกระบวนการในการจัดเรียงพื้นที่โดยมีปัจจัยที่มีผลต่อการจัดเรียงหรือการวางผังคลังสินค้าดังนี้

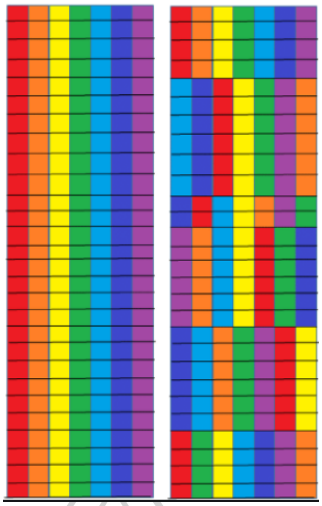
(1) ลำดับของการจัดเรียง (Permutation)

ลำดับของการนำพื้นที่มาจัดเรียงก่อนหลังจะมีผลต่อรูปแบบการวางผังที่จะเกิดขึ้น ยกตัวอย่างเช่น การจัดเรียงพื้นตามลำดับ A, C, E, D, B, F, G จะได้ผลลัพธ์ที่แตกต่างจาก B, C, F, E, G, A, D เป็นต้น โดยรูปแบบของลำดับการจัดเรียงนั่นจะใช้วิธีการ สุ่ม (Shuffle) การจัดเรียงลำดับของข้อมูล โดยอ้างอิงกับพื้นฐานทางคณิตศาสตร์ความน่าจะเป็น (Probability) เรื่องการเรียงสับเปลี่ยน (Permutation) คือ การนำพื้นที่ซึ่งมีอยู่ทั้งหมด หรือบางส่วนมาจัด โดยถือว่าลำดับมีความสำคัญ

จำนวนวิธีจัดลำดับของ n สิ่งที่แตกต่างกันเป็นแนวตรง จะสามารถจัดได้

n! = n x (n-1) x (n-2) x ... x 1 วิธี

กล่าวคือ หากมีพื้นที่จัดเก็บสินค้าโดยแบ่งเป็นทั้งหมด 7 พื้นที่ จำนวนวิธีในการลำดับการจัดเรียงจะได้ทั้งหมด 7! =7x6x5x4x3x2x1 =5040 วิธีการ โดยลำดับของการจัดเรียงในแต่ละวิธีการนั่นจะส่งผลต่อการจัดวางผังคลังสินค้าที่แตกต่างกัน และจำนวนวิธีการของลำดับการจัดเรียงนี้จะถูกนำไปใช้ในการกำหนดช่วงของ Input ใน Dynamo ที่จะส่งต่อให้ Refinery ต่อไป



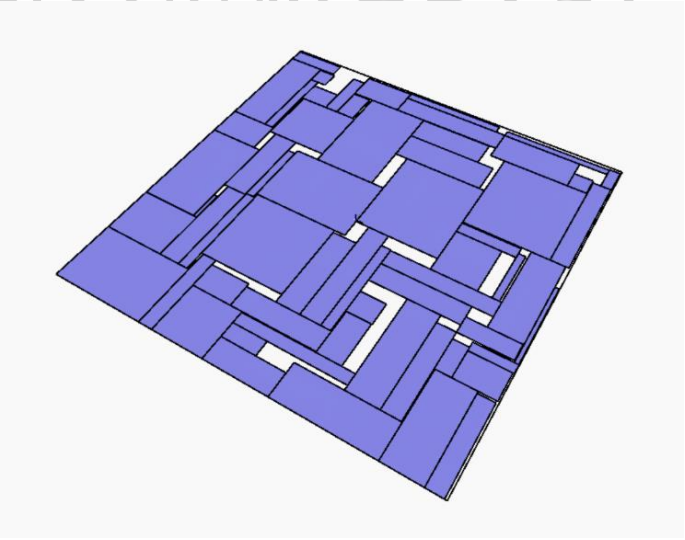
ภาพที่ 42 ภาพแสดงลำดับการจัดเรียง (Permutation) โดยสี่เหลี่ยมสีต่างๆ

(2) กระบวนการจัดเรียงพื้นที่ (Strategy)

กระบวนการในการจัดเรียงพื้นที่ (Strategy) ที่มีผลต่อผลลัพธ์ของการจัดเรียงหรือวางผังของพื้นที่โดยหลักการวางผังแบบการบรรจุสิ่งของ (Items) ลงกล่อง (container) ในระนาบ2มิติซึ่งมีกระบวนการจัดเรียง (Strategy) 3รูปแบบ (MATTERLAB, 2020)

รูปแบบที่1 (Short Side Strategy)

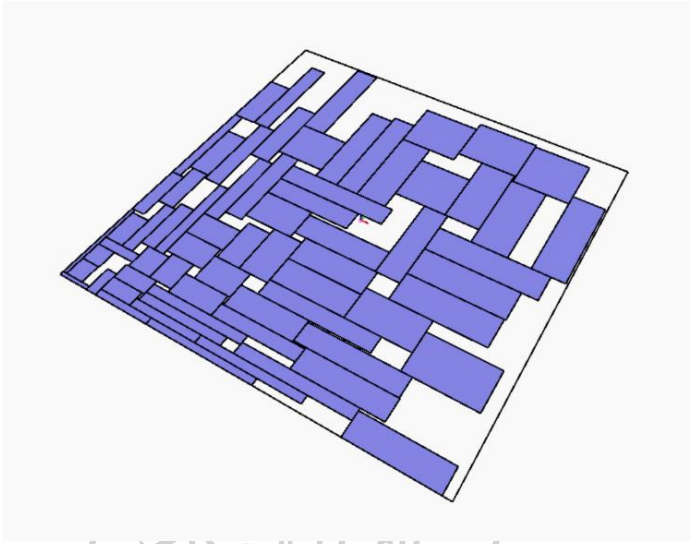
กระบวนการในการจัดเรียงพื้นที่โดยพิจารณาการจัดวางด้านกว้างของพื้นที่แต่ละพื้นที่ต่อกันก่อนเป็นความสำคัญแรก



ภาพที่ 43 ภาพการวางผังแบบ Short Side Strategy (MATTERLAB, 2020)

รูปแบบที่2 (Long Side Strategy)

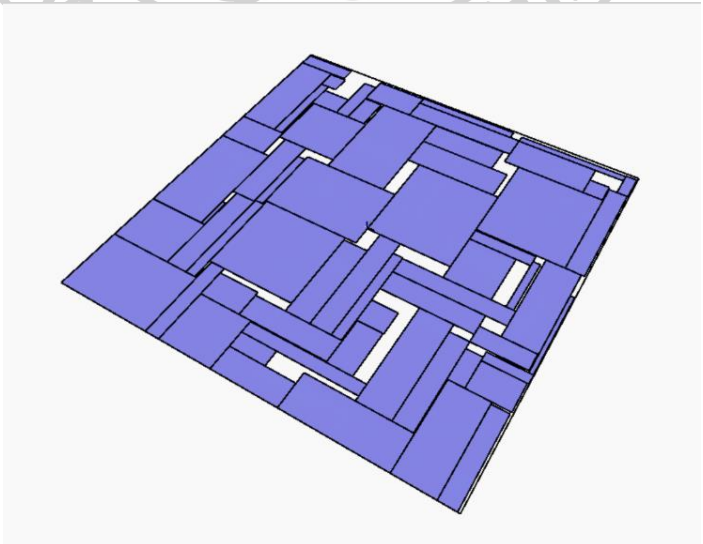
กระบวนการในการจัดเรียงพื้นที่โดยพิจารณาการจัดวางด้านยาวของพื้นที่แต่ละพื้นที่ต่อกันก่อนเป็นความสำคัญแรก



ภาพที่ 44 ภาพการวางผังแบบ Long Side Strategy (MATTERLAB, 2020)

รูปแบบที่3 (Area strategy)

กระบวนการในการจัดเรียงพื้นที่โดยพิจารณาการจัดแบบสุ่มวางทั้งด้านยาวและด้านกว้างของพื้นที่แต่ละพื้นที่ต่อกัน



ภาพที่ 45 ภาพการวางผังแบบ Area strategy (MATTERLAB, 2020)

3.1.3 ใช้ Refinery ในการสร้างและกำหนดผลลัพธ์การวางผังคลังสินค้าที่หลากหลาย

ในส่วนของการทดลองจัดวางผังคลังสินค้าในหลากหลายรูปแบบเพื่อให้เกิดตัวเลือกที่หลากหลายภายใต้เงื่อนไขที่เกี่ยวข้องของสถาปนิกนั่น จะถูกแทนที่ด้วยโปรแกรม Refinery ที่จะทำหน้าที่เป็นเครื่องมือส่วนเสริมของ Dynamo ซึ่งมีหลักการทำงานที่เรียกว่า Generative Design ซึ่งจะช่วยในการสร้างตัวเลือกของผลลัพธ์งานออกแบบโดยผู้ออกแบบสามารถกำหนดเป้าหมายของผลลัพธ์งานออกแบบได้

หน้าที่ของโปรแกรม Refinery ในงานวิจัยนี้จะทำการทดลองผลลัพธ์ของการวางผังคลังสินค้าในหลากหลายรูปแบบตามปัจจัยที่ส่งผลต่อการวางผัง ซึ่งปัจจัยที่ส่งผลต่อการวางผังคลังสินค้าจะถูกออกแบบให้เป็นค่าที่มีช่วงตั้งแต่ค่าหนึ่งไปถึงอีกค่าหนึ่ง โดยกำหนดให้เป็นค่า Input ใน Refinery ซึ่งจะนำค่าในช่วงที่กำหนดเป็น Input ไปทดลองทำให้เกิดผลลัพธ์การวางผังที่ต่างกันออกออกไป

หลังจากทำการประมวลผลลัพธ์ทั้งหมดที่เกิด Refinery จะทำการคัดเลือกเฉพาะผลลัพธ์ที่ตรงกับเป้าหมายที่ผู้แบบกำหนดไว้ ซึ่งค่าของเป้าหมายจะเป็นสิ่งที่ใช้ในการพิจารณาเลือกแบบอาคารคลังสินค้า ซึ่งเรียกว่า Output ในโปรแกรม Refinery

3.1.3.1 Input ของโปรแกรมวางผังคลังสินค้า

หลังจากเสร็จกระบวนการวางผังด้วย Dynamo เรียบร้อยในแต่ละครั้งจะเกิดผลลัพธ์ของการวางผังที่แตกต่างกัน หากปัจจัยที่มีผลต่อผลลัพธ์การวางผังคลังสินค้ามีค่าแตกต่างกันในแต่ละครั้งต่อการสร้างผลลัพธ์การวางผัง ค่าของปัจจัยที่แตกต่างกันนี้จะมีค่าเป็นช่วงที่สามารถเปลี่ยนแปลงได้โดยไม่มีผลกระทบข้อมูลความต้องการของลูกค้าที่อยู่ใน Excel โดยค่าปัจจัยที่มีผลต่อผลลัพธ์การวางผังคลังสินค้าคือค่า Input ใน Refinery มีดังนี้

(1) Input 1 ค่า Input ที่มีผลต่อพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space)

รูปทรงของพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) ที่แตกต่างกันจะส่งผลต่อผลลัพธ์การวางผังคลังสินค้าที่ต่างกัน

อ้างอิงจากข้อ 3.1.2.1 ข้อ (1) ขั้นตอนที่ 2 เพื่อการสร้างรูปทรงของพื้นที่ที่แตกต่างกันจึงเพิ่มเติมโปรแกรมให้สามารถปรับค่าของจำนวนแนว (Column) ได้ โดยการคูณเปอร์เซ็นเข้าไปตั้งแต่ 50%-100% โดยค่าเปอร์เซ็นจะทำให้ค่าของจำนวนแนว (Column) นั่นเท่าเดิมในกรณีที่เป็น 100% และลดลงจนเหลือครึ่งนึงของค่าสมมติของจำนวนแนว (Column) เดิม เนื่องจากว่าโดยหลักการการวางชั้นวางสินค้านั่นจำนวนของแนว (Column) จะมีจำนวนมากกว่า

จำนวนแถว (Row) โดยมีนัยยะสำคัญเพราะว่าจำนวนของแนว (Column) นั่นจะมีผลต่อช่องทางสัญจรของรถยก (Forklift) ซึ่งหากจำนวนของแนว (Column) มีมากก็จะทำให้พื้นเสียไปกับพื้นที่ช่องทางสัญจรของรถยก (Forklift) มากขึ้นตามไปด้วย

ค่า Input 1 = ช่วงเปอร์เซ็นความกว้าง Column (100%-50%)

(2) Input 2 ค่าInputที่มีผลต่อการจัดเรียงหรือการวางผังคลังสินค้าในแง่ของลำดับของการจัดเรียง (Permutation)

ลำดับของการนำพื้นที่มาจัดเรียงก่อนหลังที่แตกต่างกันจะส่งผลต่อผลลัพธ์การวางผังคลังสินค้าที่ต่างกัน อ้างอิงจากข้อ 3.1.2.2 ข้อ (1) การจัดเรียงใช้หลักการทางคณิตศาสตร์ความน่าจะเป็น (Probability) เรื่องการเรียงสับเปลี่ยน (Permutation) จำนวนวิธีจัดลำดับของ n สิ่งที่แตกต่างกันเป็นแนวตรง จะสามารถจัดได้

$$n! = n \times (n-1) \times (n-2) \times \dots \times 1 \text{ วิธี}$$

ทำให้เกิดวิธีจัดลำดับที่มีผลต่อผลลัพธ์การวางผังคลังสินค้าได้ n! วิธี

ค่า Input 2 = ช่วง Probability (วิธีที่ 1- วิธีที่ n!)

(3) Input 3 ค่าInputที่มีผลต่อการจัดเรียงหรือการวางผังคลังสินค้าในแง่ของกระบวนการจัดเรียงพื้นที่ (Strategy)

กระบวนการจัดเรียงพื้นที่ส่งผลต่อผลลัพธ์การวางผังคลังสินค้าที่ต่างกัน อ้างอิงจากข้อ 3.1.2.2 ข้อ (2) กระบวนการจัดเรียงพื้นที่ (Strategy) 3 รูปแบบ

-รูปแบบที่ 1 (Short Side Strategy)

-รูปแบบที่ 2 (Long Side Strategy)

-รูปแบบที่ 3 (Area strategy)

ทำให้เกิดกระบวนการจัดเรียงพื้นที่ที่มีผลต่อผลลัพธ์การวางผังคลังสินค้าได้ 3กระบวนการ

ค่า Input 3 = ช่วง Strategy (รูปแบบที่ 1 – รูปแบบที่ 3)

3.1.3.2 Output ของโปรแกรมวางผังคลังสินค้า

ผลลัพธ์การวางผังในแต่ละผลลัพธ์ที่แตกต่างกันจะให้ค่า Output ใน Refinery ที่แตกต่างกันด้วย โดยค่า Output นี้จะมีผลต่อการเลือกพิจารณาแบบที่เหมาะ อ้างอิงจากบทที่ 2 ข้อที่ 2.1.5.1 หลักเกณฑ์การพิจารณาเลือกแบบอาคารคลังสินค้า โดย Output ของโปรแกรมวางผังคลังสินค้ามีดังนี้

(1) Output 1 ค่า Output ที่มีผลต่อการเลือกพิจารณาการใช้งานที่เหมาะสมสอดคล้องกับการทำงานของผู้ใช้งาน

สามารถออกแบบให้ผลลัพธ์มีหลักการ ค่าการใกล้กันของพื้นที่ (Adjacent) ที่ควรใกล้ตามที่ผู้ออกแบบได้กำหนดการใช้งานห้องหรือพื้นที่ต่างๆในคลังสินค้าไว้จากการวิเคราะห์จากหลักการวิธีการทำงานและความต้องการจากผู้ใช้งานจริง โดยค่าของผลลัพธ์คือค่าระยะห่างระหว่างพื้นที่ที่ผู้ออกแบบกำหนดไว้ โดยพื้นที่ที่ควรใกล้กันมีค่าของระยะห่างระหว่างกันยิ่งน้อยยิ่งดี

ค่า Output1 = ค่าการใกล้กันของพื้นที่ (Adjacent)

หลักเกณฑ์การพิจารณา ค่า Adjacent ยิ่งน้อยยิ่งดี

(2) Output 2 ค่า Output ที่มีผลต่อการเลือกพิจารณาความปลอดภัยในการจัดเก็บวัตถุดิบแต่ละประเภท

สามารถออกแบบให้ผลลัพธ์มีหลักการ ค่าการไม่ควรใกล้กันของพื้นที่ (Nonadjacent) ที่ไม่ควรใกล้ตามที่ผู้ออกแบบได้กำหนดการใช้งานห้องหรือพื้นที่ต่างๆในคลังสินค้าไว้จากการวิเคราะห์จากหลักการวิธีการจัดเก็บวัตถุดิบที่ต้องคำนึงถึงความปลอดภัย โดยค่าของผลลัพธ์คือค่าระยะห่างระหว่างพื้นที่ที่ผู้ออกแบบกำหนดไว้ โดยพื้นที่ที่ไม่ควรใกล้กันไม่ควรติดกัน กล่าวคือไม่ควรมีค่าของระยะห่างระหว่างกันเท่ากับ 0

ค่า Output2 = ค่าการไม่ควรใกล้กันของพื้นที่ (Nonadjacent)

หลักเกณฑ์การพิจารณา ค่า Nonadjacent ต้องไม่เท่ากับ 0

(3) Output 3 ค่า Output ที่มีผลต่อการเลือกพิจารณาปริมาณการจัดเก็บที่เพียงพอ

โดยหลักการแล้วการคำนวนเพื่อสร้างพื้นที่การจัดเก็บของสินค้าประเภทต่างๆนั่นคำนวนจากจำนวนการจัดเก็บที่เป็นความต้องการของลูกค้า ซึ่งในขั้นตอนที่ 3 การหาขนาดของพื้นที่โดยการใช้โปรแกรมในการคำนวนพื้นที่ทำให้พื้นที่ที่ได้ตรงกับความต้องการของลูกค้าอยู่แล้ว แต่มีอีกกรณีคือ ความต้องการของลูกค้าเมื่อทำการคำนวณแล้วเกินกว่าขนาดของคลังสินค้าที่จะรับได้ ทำให้ผู้ออกแบบต้องคำนวณเพื่อนำเสนอผลลัพธ์ของปริมาณการจัดเก็บที่แท้จริงโดยสอดคล้องกับขนาดของพื้นที่จริงที่จัดเก็บได้ โดยปริมาณการจัดเก็บ (Capacity) นี้จะมีผลอย่างมากในการพิจารณาเลือกแบบการวางผังคลังสินค้าของลูกค้า

ค่า Output 3 = ค่าปริมาณการจัดเก็บ (Capacity)

หลักเกณฑ์การพิจารณา ค่า Capacity ยิ่งมากยิ่งดี

(4) Output 4 ค่า Output ที่มีผลต่อการเลือกพิจารณาขนาดอาคาร

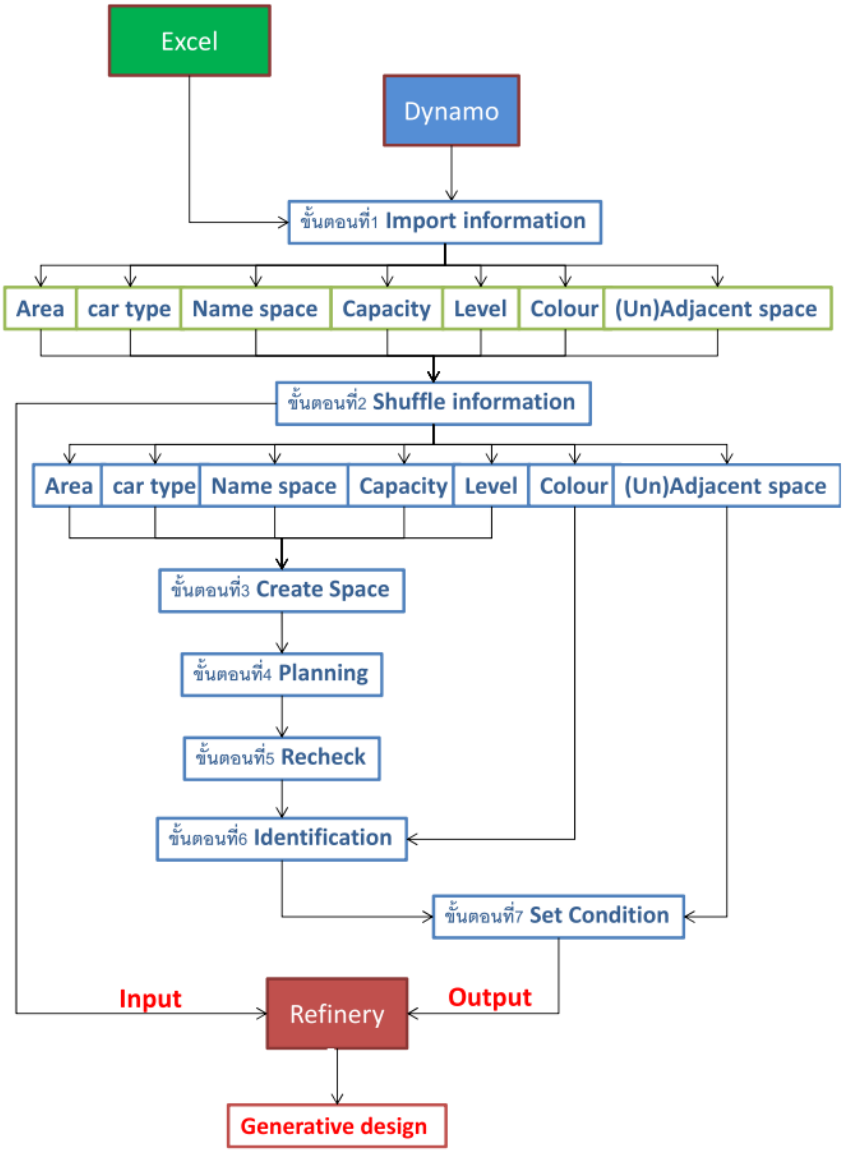
ผลลัพธ์ของปริมาณพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) ในแต่ละผลลัพธ์ของการวางผังคลังสินค้าจะแตกต่างกันอันเนื่องมาจากค่า Input ที่มีผลต่อพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) ทำให้ผลลัพธ์ของขนาดอาคาร (Building Area) มีผลในการพิจารณาเลือกแบบการวางผังคลังสินค้าของลูกค้าในแง่ของราคาในการก่อสร้างที่จะสูงขึ้นตามขนาดอาคารและพื้นที่ว่างที่เหลือจะยิ่งน้อยลง

ค่า Output 4 = ขนาดอาคาร (Building Area)

หลักเกณฑ์การพิจารณาค่า Building Area ยิ่งน้อยยิ่งดี

3.2 การออกแบบและขั้นตอนการทำงานของโปรแกรมวางผังคลังสินค้า (Warehouse Planning Program)

หลังจากเสร็จสิ้นกระบวนการวางแนวความคิดหัวข้อนี้จะกล่าวถึงรายละเอียดในการออกแบบโปรแกรมเพื่อตอบสนองแนวความในการนำโปรแกรมคอมพิวเตอร์มาใช้ในการจัดวางผังคลังสินค้า โดยจะแบ่งการออกแบบและขั้นตอนการทำงานของโปรแกรมเป็น 3 ส่วน ตามแนวความคิดในการวางผังคลังค้า



ภาพที่ 46 แผนภาพการออกแบบขั้นตอนการทำงานของโปรแกรมวางผังคลังสินค้า

3.2.1 โปรแกรมวางผังคลังสินค้าส่วน Excel

	A	B	C	D	E	F	G	H	I	J	K
1	AREA	CAR TYPE	NUMBER OF CAR TYPE	SPACE ID	SPACE NAME	CAPACITY	RACK LEVEL	/	COLOUR	ADJACENT SPACES	UNADJACENT SPACES
2	85	PICKUP	0	0	FREEZER	54	3		128,128,128	COLD	FREEZER, FLAMABLE, C
3	65	TRUCK	3	1	FLAMABLE	30	3		255,0,0	FLAMABLE	ALLERGEN
4		TRAILER	0	2	COLD	87	3		255,128,0	FREEZER	FREEZER, FLAMABLE, C
5				3	ALLERGEN	12	6		255,0,128	ALLERGEN	FLAMABLE
6				4	AMBIENT	2088	6		255,128,128	AMBIENT	FREEZER, FLAMABLE, C
7				5	STAGING	32	1		255,255,0	STAGING	FREEZER, FLAMABLE, C

ภาพที่ 47 ภาพแสดงโปรแกรมวางผังคลังสินค้าส่วน Excel

ในส่วนการออกแบบรูปแบบข้อมูลความต้องการของลูกค้าบนโปแกรม Excel นั่น ผู้วิจัยได้แบ่งกลุ่มข้อมูลตามลักษณะการกรอกข้อมูลและการดึงข้อมูลไปใช้ใน Excel เป็น 3 ประเภทดังนี้

3.2.1.1 ประเภทที่ 1 จำกัดจำนวนข้อมูลในการกรอก

จำกัดจำนวนข้อมูลในการกรอก และข้อมูลจะถูกดึงไปใช้โดยระบุตำแหน่งที่ต้องกรอกอย่างชัดเจน กล่าวคือผู้ใช้งานจำเป็นต้องกรอกข้อมูลประเภทนี้ในช่องที่กำหนดเท่านั่นเพื่อความถูกต้องในการดึงข้อมูลเพื่อไปประมวลผลต่อใน Dynamo โดยข้อมูลประเภทที่1 มีดังนี้

	A	B	C	D	E	F	G	H	I	J	K
1	AREA	CAR TYPE	NUMBER OF CAR TYPE	SPACE ID	SPACE NAME	CAPACITY	RACK LEVEL	/	COLOUR	ADJACENT SPACES	UNADJACENT SPACES
2	85	PICKUP	0	0	FREEZER	54	3		128,128,128	COLD	FREEZER, FLAMABLE, C
3	65	TRUCK	3	1	FLAMABLE	30	3		255,0,0	FLAMABLE	ALLERGEN
4		TRAILER	0	2	COLD	87	3		255,128,0	FREEZER	FREEZER, FLAMABLE, C
5				3	ALLERGEN	12	6		255,0,128	ALLERGEN	FLAMABLE
6				4	AMBIENT	2088	6		255,128,128	AMBIENT	FREEZER, FLAMABLE, C
7				5	STAGING	32	1		255,255,0	STAGING	FREEZER, FLAMABLE, C

ภาพที่ 48 ภาพแสดงข้อมูลประเภทที่ 1 จำกัดจำนวนข้อมูลในการกรอก (สีเหลือง)

(1) พื้นที่ขอบเขตที่ดิน (Area)

ข้อมูลขอบเขตที่ดิน ถูกออกแบบให้กรอกข้อมูลเพียง 2 ค่า คือความกว้างของที่ดิน และ ความยาวของที่ดิน เนื่องจากผู้วิจัยเห็นว่าโปรแกรมการวางคลังสินค้านี้อยู่ในช่วงขั้นตอนการออกแบบขั้นต้น Conceptual Design ซึ่งต้องการความรวดเร็วและไม่ต้องการรายละเอียดที่ซับซ้อนมากนัก ดังนั่นการกำหนดขอบเขตที่ดินอย่างคร่าวโดยการใช้เพียงกว้างและความยาวของที่ดินนั่นเพียงพอต่อการวางผังคลังสินค้าอย่างมีประสิทธิภาพในช่วงการออกแบบเบื้องต้นแล้ว

(2) จำนวนของรถชนิดต่างๆที่ถูกใช้ในการขนส่ง (No. of Each Car Type)

ข้อมูลของจำนวนรถที่ใช้ในการขนส่งนั่นจะถูกกำหนดให้กรอกค่าเป็นจำนวนของรถที่ใช้จริงในคลังสินค้าโดยมีหัวข้อเป็นรถ3ชนิดที่ผู้วิจัยทำการสรุปประเภทรถที่ใช้เป็นมาตรฐานของคลังสินค้าโดยทั่วไป ซึ่งมีความแตกต่างกันในด้านขนาดของตัวรถซึ่งจะส่งผลต่อพื้นที่ลานขนถ่ายสินค้า รถ3ประเภทได้แก่ รถ Trailer 40 Feet, รถ 10 ล้อ และ รถกะบะ Pick up

3.2.1.2 ประเภทที่ 2 ข้อมูลที่ไม่จำกัดปริมาณการกรอก

ข้อมูลที่ไม่จำกัดปริมาณการกรอกของข้อมูลในัวข้อนั้นๆซึ่งแต่ละลำดับของข้อมูลนั่นจะถูกไล่เรียงลงไปตาม Column โดยข้อมูลที่กรอกในแต่ละ Column จะเป็นข้อมูลค่าเดียวเท่านั่น ข้อมูลประเภทที่ 2 มีดังนี้

(1) รายชื่อพื้นที่จัดเก็บสินค้าแต่ละประเภท (Name Space)

(2) จำนวนความต้องการจัดเก็บสินค้าในแต่ละประเภท (Capacity)

(3) จำนวนการซ้อนชั้นของการจัดเก็บสินค้าแต่ละประเภท (Level)

(4) กำหนดสีเพื่อแบ่งที่การจัดเก็บสินค้าแต่ละประเภท (Colour)

	A	B	C	D	E	F	G	I	J	K
1	AREA	CAR TYPE	NUMBER OF CAR TYPE	SPACE ID	SPACE NAME	CAPACITY	RACK LEVEL	COLOUR	ADJACENT SPACES	UNADJACENT SPACES
2	85	PICKUP	0	0	FREEZER	54	3	128,128,128	COLD	FREEZER, FLAMABLE, C
3	65	TRUCK	3	1	FLAMABLE	30	3	255,0,0	FLAMABLE	ALLERGEN
4		TRAILER	0	2	COLD	87	3	255,128,0	FREEZER	FREEZER, FLAMABLE, C
5				3	ALLERGEN	12	6	255,0,128	ALLERGEN	FLAMABLE
6				4	AMBIENT	2088	6	255,128,128	AMBIENT	FREEZER, FLAMABLE, C
7				5	STAGING	32	1	255,255,0	STAGING	FREEZER, FLAMABLE, C

ภาพที่ 49 ภาพแสดงข้อมูลประเภทที่ 2 ข้อมูลที่ไม่จำกัดปริมาณการกรอก (สีเหลือง)

3.2.1.3 ประเภทที่ 3 ข้อมูลที่กรอกในแต่ละ Column จะมีมากกว่า 1 ค่า

ข้อมูลประเภทนี้จะมีลักษณะของปริมาณการกรอกคล้ายกับประเภทที่ 2 คือ ข้อมูลที่ไม่จำกัดปริมาณการกรอกของข้อมูลในหัวข้อนั้นๆซึ่งแต่ละลำดับของข้อมูลนั่นจะถูกไล่เรียงลงไปตาม Column แต่ข้อมูลที่กรอกในแต่ละ Column จะมีมากกว่า 1 ค่าโดยแต่ละค่าจะถูกคั่นด้วยสัญญาลักษณ์ คอมม่า (,) มี 2 กลุ่มข้อมูลที่มีลักษณะของข้อมูลเป็นประเภทที่ 3 คือ

(1) พื้นที่ที่ควรอยู่ใกล้กัน (Adjacent Space)

(2) พื้นที่ต้องอยู่ไกลกัน (Nonadjacent Space)

เนื่องจากพื้นที่ที่ควรใกล้กันหรือไม่ควรใกล้กันนั่นกับอีกพื้นที่หนึ่งนั่นสามารถมีได้มากกว่าหนึ่งพื้นที่ ยกตัวอย่างเช่นพื้นที่จัดเก็บวัตถุไวไฟ ไม่ควรติดกับพื้นที่จัดเก็บวัตถุ

ระเบิด และไม่ควรติดกับ พื้นที่เก็บสารที่เกิดการระคายเคืองด้วยเช่นกัน ดังนั้นทำให้ค่าของข้อมูลในหัวข้อ ทั้ง2ข้างต้นจึงมีค่าของข้อมูลมากกว่าหนึ่งค่าขึ้นไป

	A	B	C	D	E	F	G	I	J	K
1	AREA	CAR TYPE	NUMBER OF CAR TYPE	SPACE ID	SPACE NAME	CAPACITY	RACK LEVEL	COLOUR	ADJACENT SPACES	UNADJACENT SPACES
2	85	PICKUP	0	0	FREEZER	54	3	128,128,128	COLD	FREEZER, FLAMABLE, C
3	65	TRUCK	3	1	FLAMABLE	30	3	255,0,0	FLAMABLE	ALLERGEN
4		TRAILER	0	2	COLD	87	3	255,128,0	FREEZER	FREEZER, FLAMABLE, C
5				3	ALLERGEN	12	6	255,0,128	ALLERGEN	FLAMABLE
6				4	AMBIENT	2088	6	255,128,128	AMBIENT	FREEZER, FLAMABLE, C
7				5	STAGING	32	1	255,255,0	STAGING	FREEZER, FLAMABLE, C

ภาพที่ 50 ภาพแสดงข้อมูลประเภทที่ 3 ข้อมูลที่กรอกในแต่ละ Column จะมีมากกว่า 1 ค่า (สีเหลือง)

3.2.2 โปรแกรมวางผังคลังสินค้าส่วน Dynamo

ขั้นตอนการทำงานของโปรแกรมวางผังคลังสินค้าบน Dynamo นั่นถูกออกแบบให้ มีรูปแบบการทำงานเพื่อตอบสนองแนวคิดของโปรแกรมวางผังคลังสินค้าในข้อ 3.1.2 คือ การดึงข้อมูลจาก ข้อมูลความต้องการของลูกค้าจากไฟล์ Excel นำมาสร้างพื้ทที่ที่มีผลต่อการวางผังคลังสินค้า หลักจากที่สร้างพื้นที่ทั้งหมดก็ถูกนำมาสู่ขั้นตอนการจัดวางผังคลังสินค้า โดยมีการวางเงื่อนไขในการวางผังเพื่อให้เกิดค่าผลลัพธ์ภายใต้เงื่อนไขที่กำหนด และนำค่าของผลลัพธ์ไปใช้ในการกำหนดเป้าหมายของผลลัพธ์ที่ต้องการให้เกิดขึ้นต่อไปในโปรแกรม Refinery โดยมีขั้นตอนการทำงานของโปรแกรมวางผังคลังสินค้าบน Dynamo 7 ขั้นตอนหลักดังนี้

ขั้นตอนที่ 1 นำเข้าข้อมูล (Import information)

ขั้นตอนที่ 2 สลับสับเปลี่ยนข้อมูล (Shuffle Information)

ขั้นตอนที่ 3 สร้างพื้นที่จากข้อมูลขั้นตอนที่ 2 (Create Space)

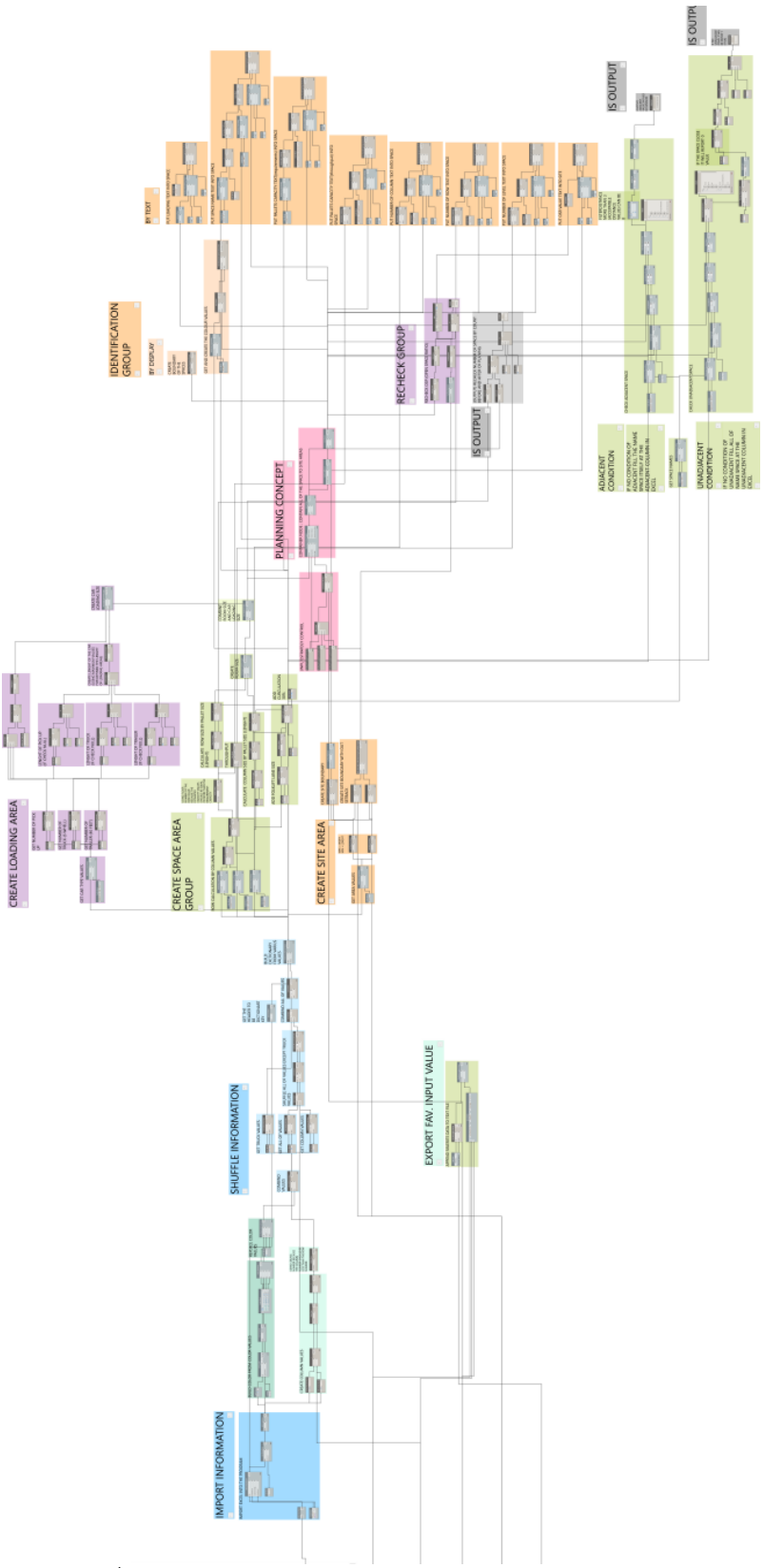
ขั้นตอนที่ 4 จัดเรียงวางผังพื้นที่จากขั้นตอนที่ 3 (Planning)

ขั้นตอนที่ 5 ตรวจสอบความถูกต้องของผังที่ถูกจัดเรียงในขั้นตอนที่ 4 (Recheck)

ขั้นตอนที่ 6 ระบุข้อมูลจำเพาะให้แสดงผลลงบนผังที่ถูกจัดเรียงในขั้นตอนที่4 (Identification)

ขั้นตอนที่ 7 สร้างเงื่อนไขเพื่อให้เกิดค่าของข้อมูลในแต่ละครั้งของผลลัพธ์ในการวางผังที่เกิดขึ้น (Set Condition)

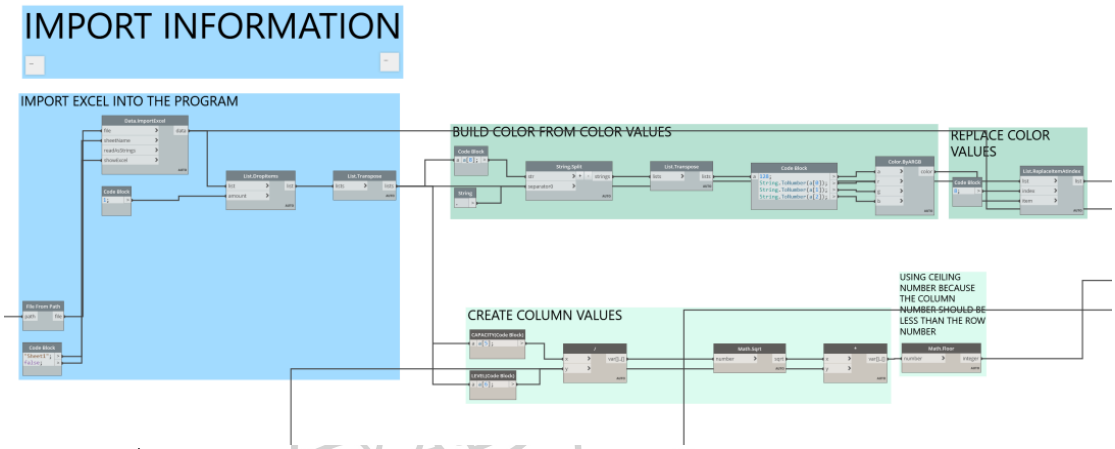
ขั้นตอนข้างต้นทั้งหมดได้ถูกนำมาสร้างเป็นโปรแกรมบน Dynamo โดยมีรายละเอียดของการเขียนโปรแกรมหรือการเลือกใช้ Node ซึ่งเป็นชุดเครื่องมือคำสั่งต่างๆที่ตรงกับความต้องการของโปรแกรมที่กำลังออกแบบในแต่ละขั้นตอน



ภาพที่ 51 ภาพแสดงโปรแกรมวางผังคลังสินค้าส่วน Dynamo

3.2.2.1 นำเข้าข้อมูล (Import Information)

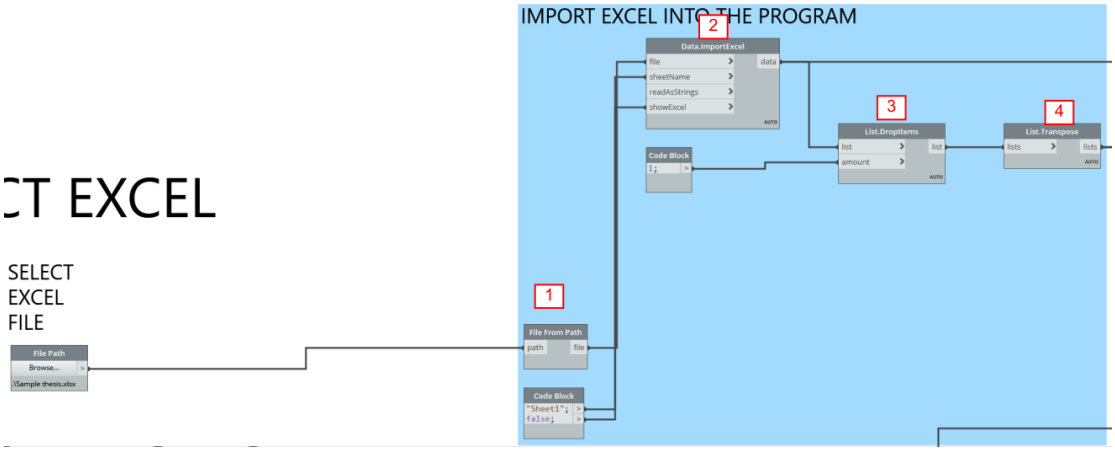
เป็นขั้นตอนแรกของโปรแกรมคือการนำเข้าข้อมูลดิบที่ถูกกรอกอยู่ใน Excel เพื่อนำมาเป็นข้อมูลตั้งต้นของโปรแกรม โดยในขั้นตอนการนำเข้าข้อมูลนี้ ข้อมูลดิบบางค่าสามารถดึงไปใช้งานต่อในขั้นตอนต่อไปของโปรแกรมได้เลย บางค่าต้องมีการนำมาแปลงค่าหรือสร้างเป็นข้อมูลชุดใหม่ก่อนที่จะดึงค่าไปใช้ในขั้นตอนต่อไปของโปรแกรม



ภาพที่ 52 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Import Information

ขั้นตอนการนำเข้าข้อมูลขั้นต้นมี3 ส่วน

(1) ส่วนที่ 1 Import Excel into program



ภาพที่ 53 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel into program

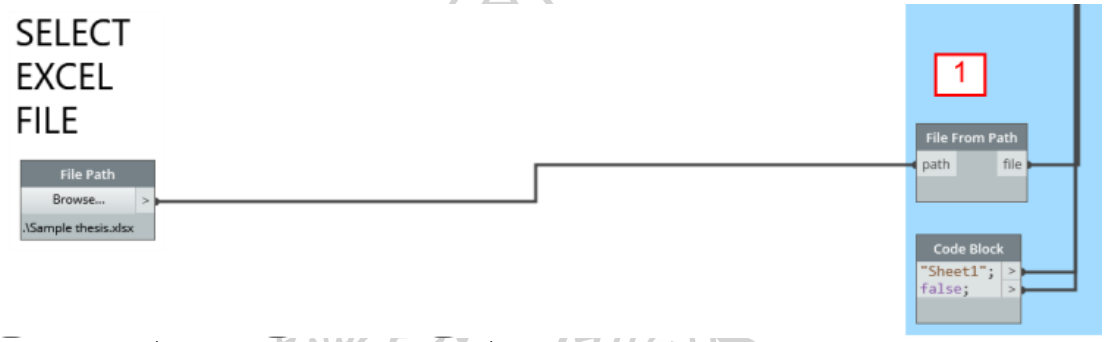
หมายเลข 1 นำเข้าไฟล์excelข้อมูลความต้องการของลูกค้าโดยวิธีการเลือกไฟล์ (Browse) ในที่อยู่ที่ไฟล์ถูกจัดเก็บไว้พร้อมทั้งระบุตำแหน่งของหน้าSheetที่ข้อมูลบรรจุอยู่

Node ที่เกี่ยวข้อง

-File Path ใช้ระบุตำแหน่งไฟล์ที่ต้องการใช้

-File From Path ใช้แปลงจากข้อมูลตำแหน่งไฟล์ (Path) เป็นค่าไฟล์ (File)

-Code Block ใช้พิมค่า String ของSheet1 และ ค่าfalse เพื่อเป็นค่า input ของ node ในข้อถัดไป

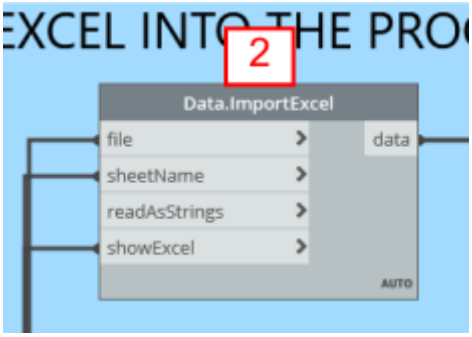


ภาพที่ 54 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel into program หมายเลข 1

หมายเลข 2 นำเข้าไฟล์excelข้อมูลความต้องการของลูกค้าจากไฟล์ที่เลือก

<u>Nodeที่เกี่ยวข้อง</u>

-Data.ImportExcel ใช้ในการดึงข้อมูลจากไฟล์ Excel ที่เลือกจากหมายเลข 1 โดยระบุทั้งตำแหน่งที่จัดเก็บไฟล์และหมายเลข Sheet ใน Excel

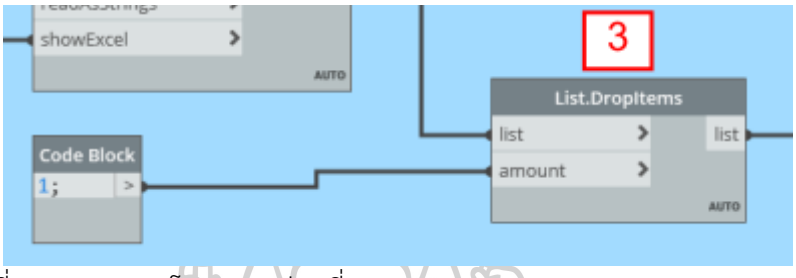


ภาพที่ 55 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel into program หมายเลข 2

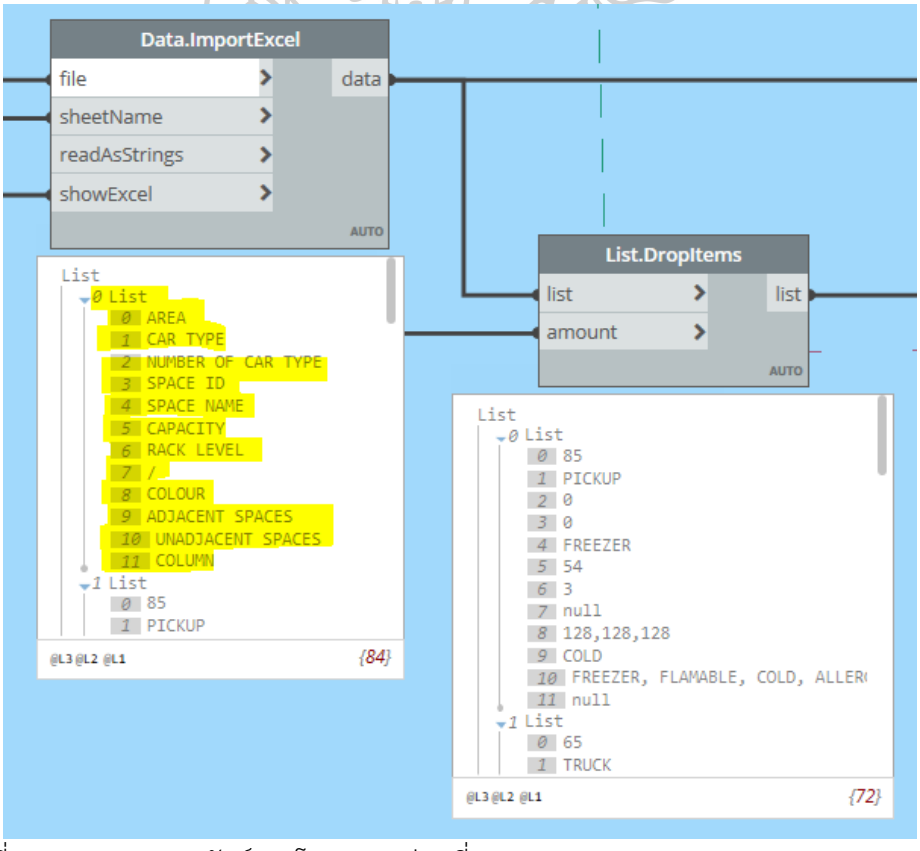
หมายเลข3 การตัดข้อมูลที่เป็นหัวข้อของความต้องการที่ใช้ในการออกแบบออก โดยให้เหลือเพียงข้อมูลของรายละเอียดในแต่ละหัวข้อนั่นๆ

Nodeที่เกี่ยวข้อง

-List.Dropitems เพื่อทำการคัดข้อมูลที่ไม่ต้องการออกจาก List ข้อมูลโดยการระบุลำดับของ List ที่ต้องการตัดออก



ภาพที่ 56 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel into program หมายเลข 3



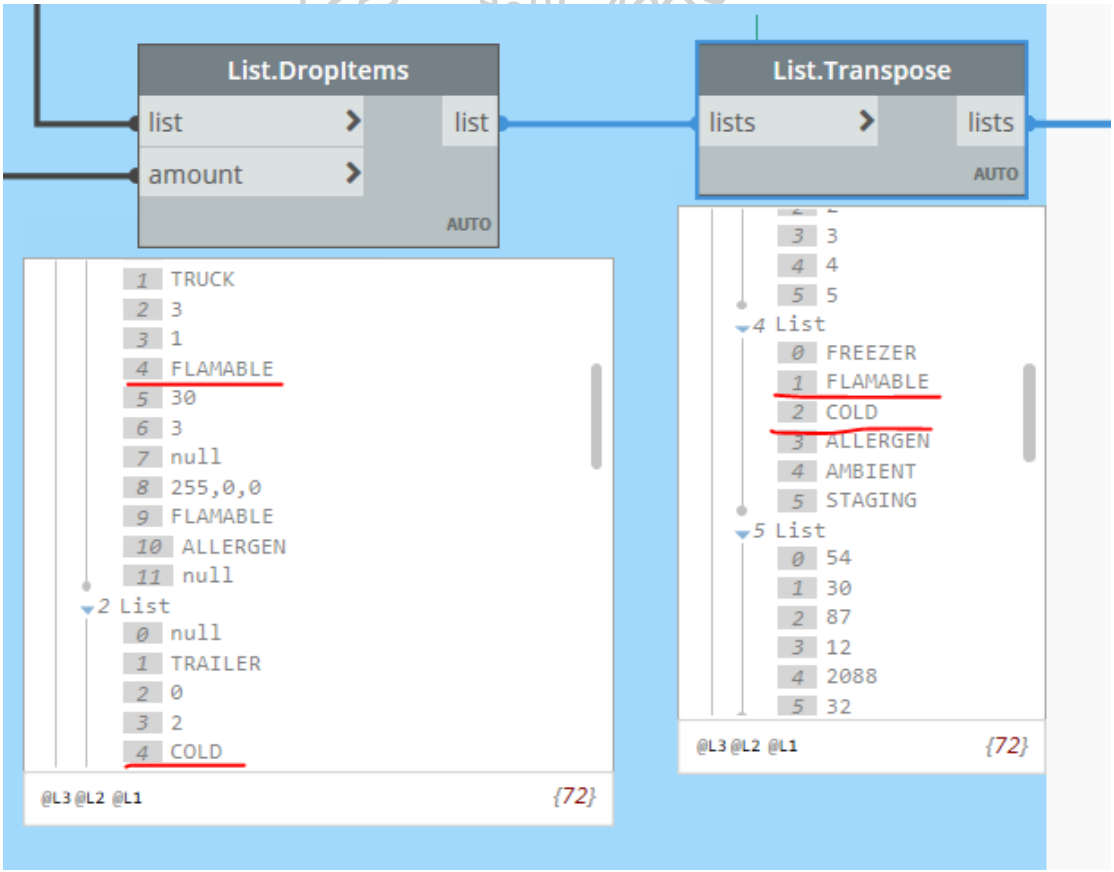
ภาพที่ 57 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 1 Import Excel into program หมายเลข 3

หมายเลข 4 ทำการจัดเรียงข้อมูลใหม่ให้ข้อมูลที่อยู่ในหัวข้อเดียวกันอยู่ใน List เดียวกัน

Node ที่เกี่ยวข้อง

-List.Tranpose ทำการไขว้ข้อมูลโดยจากเดิมคือข้อมูลความต้องการลูกค้าที่ได้มาจาก Excel เรียงจาก Row 1, 2, 3, 4 ให้กลายเป็นเรียงจาก column A, B, C, D แทน ทำให้ข้อมูลที่อยู่ในหัวข้อเดียวกันอยู่ใน List เดียวกัน

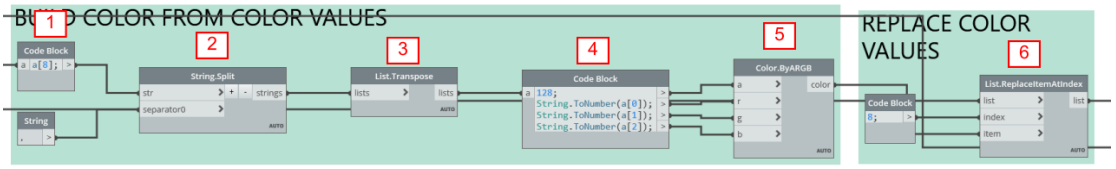
ภาพที่ 58 ภาพแสดงโปรแกรมส่วนที่ 1 Import Excel Into program หมายเลข 4



ภาพที่ 59 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 1 Import Excel Into program หมายเลข 4

(2) ส่วนที่ 2 Build Color From Color Values

เป็นขั้นตอนการนำเข้าข้อมูลค่าสีที่ถูกกรอกเป็นตัวเลขลงใน Excel มาแปลงเป็นค่าสีในระบบ RGB เพื่อนำไปใช้ในการแสดงสีต่างๆของพื้นที่ (Geometry) ที่จะถูกสร้างขึ้น

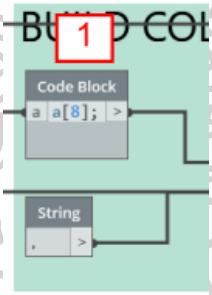


ภาพที่ 60 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values

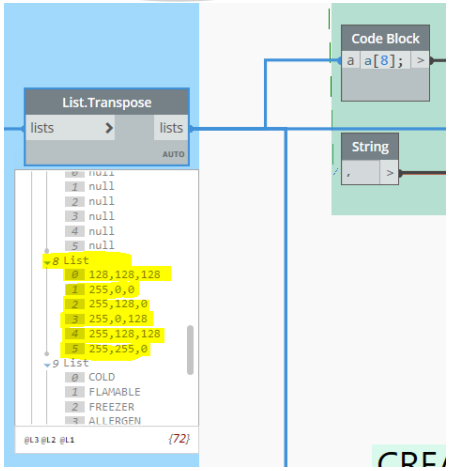
หมายเลข1 ดึงค่าของสี ซึ่งอยู่ในตำแหน่งที่ 8 ของ List

<u>Nodeที่เกี่ยวข้อง</u>

Code Block สร้างโค้ด a[8] คือการดึงค่า List ตำแหน่งที่ 8 ของข้อมูล a ซึ่งคือข้อมูลของค่า Colour ที่ถูกระบุในโปรแกรมส่วน Excel



ภาพที่ 61 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 1



ภาพที่ 62 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 1

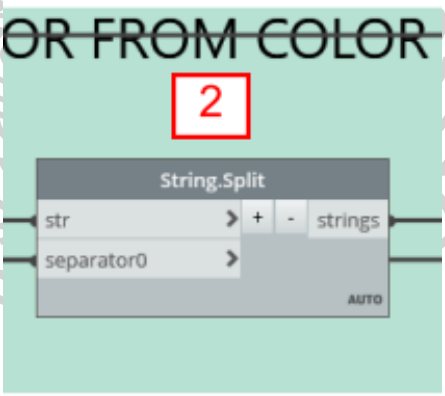
หมายเลข 2 ค่าสีที่ใส่นั่นเป็นค่า String ที่มี 3 ค่าในมาตรฐานสีแบบ RGB โดยมีการกรอกข้อมูลทั้ง 3 ค่าใน Cell เดียวกันโดยขั้นด้วย Comma (,) จึงต้องทำการแยกค่าทั้ง 3 ออกจากกัน

Nodeที่เกี่ยวข้อง

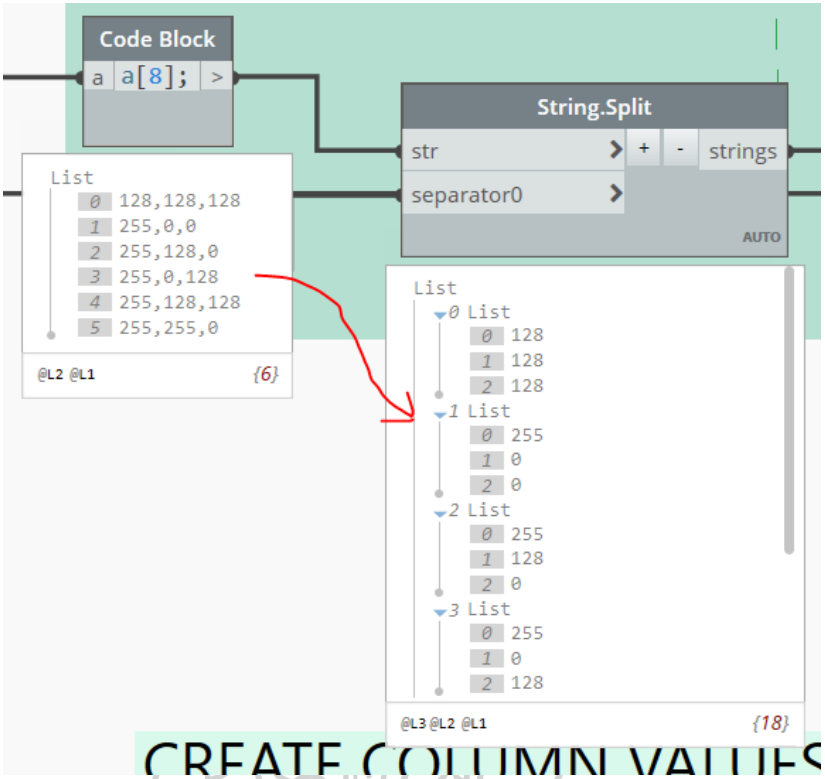
String.Split ใช้แยกค่าของข้อมูลที่เป็น String ออกจากกันโดยตัวแปร comma (,)

COLOUR
128,128,128
255,0,0
255,128,0
255,0,128
255,128,128
255,255,0

ภาพที่ 63 ตัวอย่างข้อมูลที่ถูกขั้นด้วย Comma (,)



ภาพที่ 64 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 2

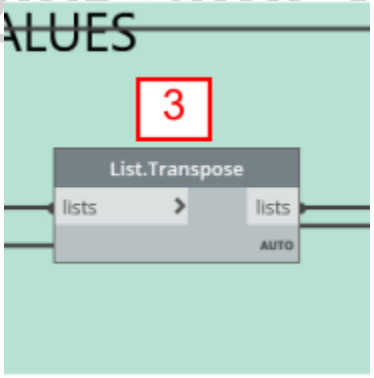


ภาพที่ 65 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 2

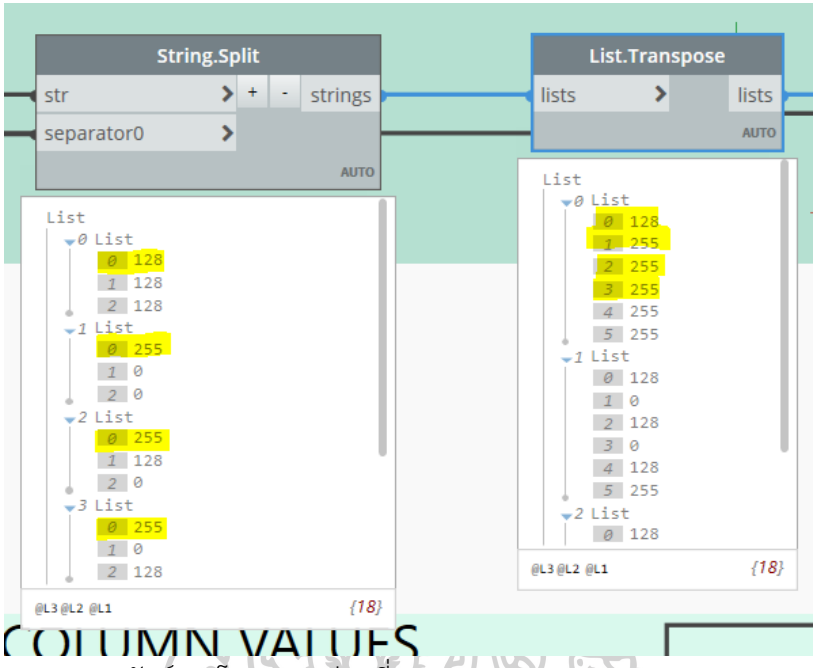
หมายเลข 3 สับเปลี่ยนข้อมูลเพื่อจัดกลุ่มข้อมูลสี RGB

Node ที่เกี่ยวข้อง

List.Tranpose ทำการไขว้ข้อมูลโดยจากเดิมข้อมูล RGB เพื่อสร้างค่า List ใหม่โดยการจัดกลุ่มของ R, G และ B เพื่อให้ข้อมูลที่อยู่ในค่าสีเดียวกันอยู่ใน List เดียวกัน



ภาพที่ 66 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 3



ภาพที่ 67 ภาพแสดงผลลัพธ์ของโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 3

หมายเลข4 แปลงค่าสีที่เป็น string เป็น ค่าตัวเลข

<u>Nodeที่เกี่ยวข้อง</u>

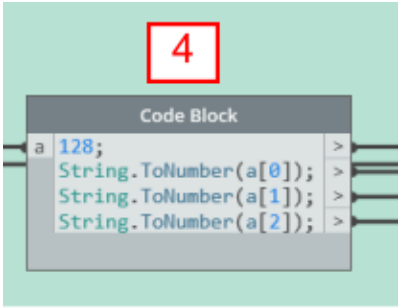
-Code block ใช้ในการเขียนโค้ดคำสั่ง

128; เป็นค่าขอ a ในระบบสี ARGB

String.ToNumber (a[0]) แปลงค่า string จาก input a ซึ่งมาจาก output (strings) ใน list ที่ 0 ของ node List.Tranpose (หมายเลข3) ซึ่งคือค่า R

String.ToNumber (a[1]) แปลงค่า string จาก input a ซึ่งมาจาก output (strings) ใน list ที่ 1 ของ node List.Tranpose (หมายเลข3) ซึ่งคือค่า G

String.ToNumber (a[2]) แปลงค่า string จาก input a ซึ่งมาจาก output (strings) ใน list ที่ 2 ของnode List.Tranpose (หมายเลข3) ซึ่งคือค่า B

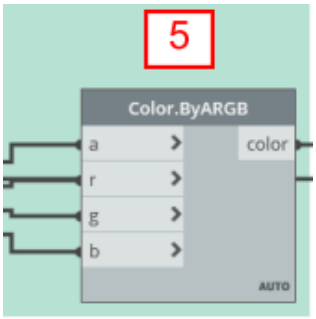


ภาพที่ 68 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 4

หมายเลข 5 สร้างสีcolorในระบบ RGB

Node ที่เกี่ยวข้อง

Color.ByARGB เพื่อประมวลผลสร้างสีในระบบ RGB จากการกำหนดค่าตามตัวเลขของค่า R, G, B ที่ได้จาก Node Code block (หมายเลข 4)



ภาพที่ 69 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 5

หมายเลข6 นำสี Colour ที่สร้างขึ้น ใส่กลับเข้าไปในตำแหน่งที่ 8 ของ List

Node ที่เกี่ยวข้อง

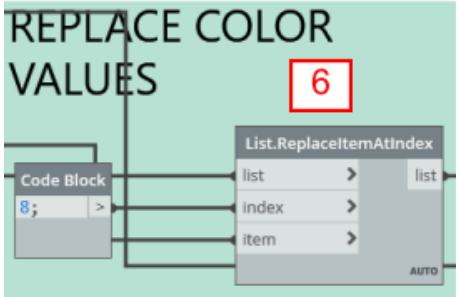
List.ReplaceItemAtIndex ใช้ในการนำข้อมูลใหม่ (Item) แทนที่เข้าไปใน List ข้อมูลเดิม

โดยมี Input ที่ใช้คือ

-List คือ ชุดข้อมูลเดิมซึ่งมาจาก Output ของ Node List.Tranpose (หมายเลข 4 ส่วนที่ 1)

-Index คือตำแหน่งของใน List ข้อมูลเดิมที่ต้องการแทนที่ด้วยข้อมูลใหม่ซึ่งในที่นี้คือตำแหน่งที่ 8 ใน List โดยใช้ Code Block กำหนดค่า 8

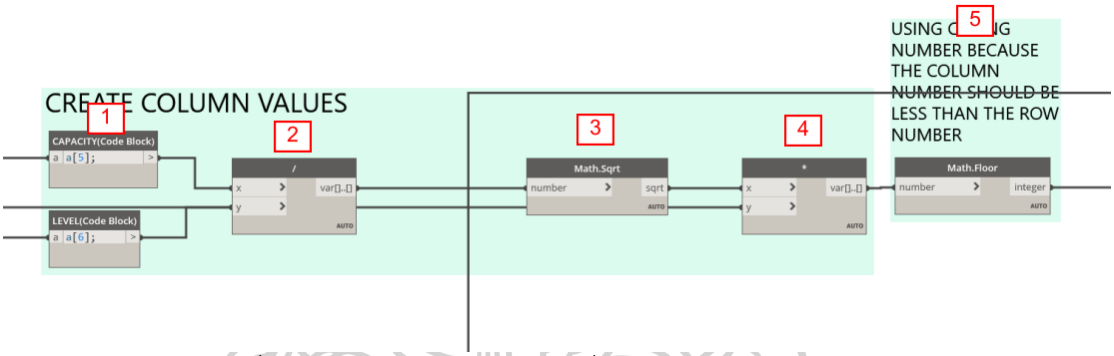
-Item คือข้อมูลใหม่ที่จะใส่แทนที่ซึ่งมาจาก Output (Colour) ของ Node Color.ByARGB (หมายเลข 5)



ภาพที่ 70 ภาพแสดงโปรแกรมส่วนที่ 2 Build Color From Color Values หมายเลข 6

(3) ส่วนที่ 3 Create Column Values

ขั้นตอนนี้เป็นการสร้างค่าของแถว (Row) และ แนว (Column) ขึ้นมาจากข้อมูลค่า Capacity และ Level เพื่อนำไปใช้เป็นข้อมูลตั้งต้นในโปรแกรมจัดวางผังคลังสินค้าในส่วนถัดไป ขนาดของพื้นที่เก็บสินค้าแต่ละชนิดนั่นมีขนาดแปรผันตามจำนวนพาเลทในชั้น 1 (Locations) และแต่ละพาเลทจะถูกวางเรียงกันเกิดเป็น แถว (Row) และ แนว (Column) จนครบจำนวนตามความต้องการของลูกค้า โดยที่รูปทรงของพื้นที่จัดเก็บสินค้าความกว้างและความยาวของรูปทรงเป็นผลจากการจัดวางจำนวน แถว (Row) และ แนว (Column) ของชั้นวางสินค้า (Rack) ในขั้นโปรแกรมส่วนนี้จะเป็นการหาค่าของแถว (Row) และ แนว (Column) ที่สัมพันธ์กับความต้องการการจัดเก็บของลูกค้า โดยวิธีการโปรแกรมดังนี้

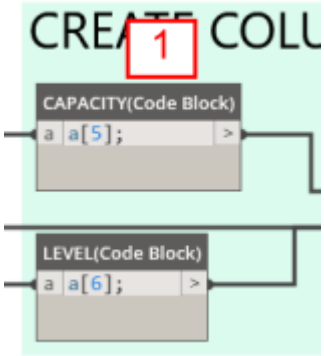


ภาพที่ 71 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values

หมายเลข1

Node ที่เกี่ยวข้อง

Code block สร้างโค้ด a[5] คือการดึงค่า List ตำแหน่งที่ 5 ของข้อมูล a ซึ่งคือข้อมูลของค่า Capacity และ Level

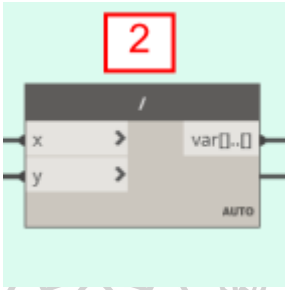


ภาพที่ 72 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 1

หมายเลข2

Node ที่เกี่ยวข้อง

-/ คือการหาร โดยดึงค่า Capacity หารกับค่า Level เพื่อให้ได้ค่าจำนวนพาเลท (locations) ที่ใช้ในพื้นที่แต่ละพื้นที่

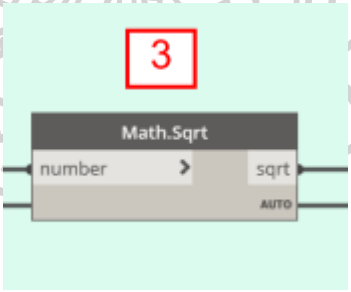


ภาพที่ 73 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 2

หมายเลข 3

Node ที่เกี่ยวข้อง

-Math.Sqrt ใช้ในการหาค่า Square Root ของจำนวนพาเลท (Locations) ทั้งหมดที่ใช้ในพื้นที่แต่ละพื้นที่โดยค่าที่ได้คือค่า แถว (Row) และ แนว (Column) โดยขั้นตอนนี้พื้นที่จะเป็นลักษณะของพื้นที่สี่เหลี่ยมจัตุรัส



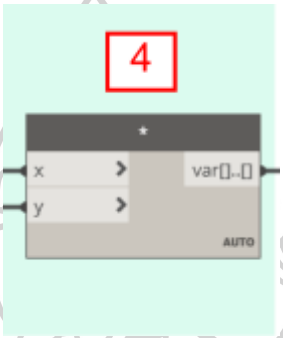
ภาพที่ 74 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 3

หมายเลข 4 หลังจากได้ค่า แถว (Row) และ แนว (Column) ของพื้นที่ลักษณะเป็นสี่เหลี่ยมจัตุรัส ซึ่งได้ค่าแถว (Row) และ แนว (Column) ที่เท่า ในขั้นตอนนี้ เป็นการคูณค่าเปอร์เซ็นเพื่อให้เกิดความเล็กใหญ่หรือกว้างยาวที่แตกต่างกันของขนาดพื้นที่ หรือกล่าวให้เข้าใจง่ายคือทำให้พื้นที่ไม่เป็นสี่เหลี่ยมจัตุรัสเพื่อให้เกิดรูปร่างที่หลากหลายของพื้นที่ โดยการคูณค่าเปอร์เซ็นให้กับ ค่าแนว (Column) หากค่า Column เปลี่ยนแปลงค่าของ Row ก็จะเปลี่ยนแปลง

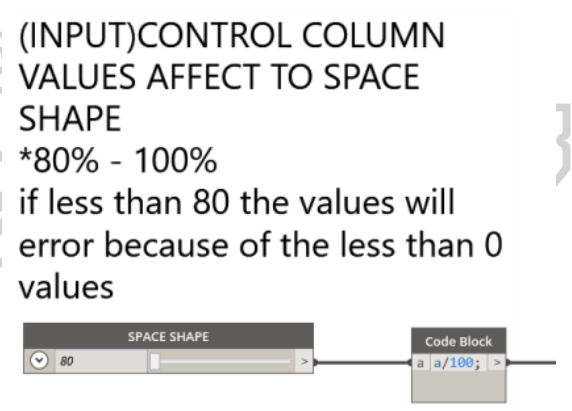
ตามเป็นสัดส่วนต่อกันและส่งผลต่อรูปร่างของพื้นที่ที่เปลี่ยนแปลงไป ซึ่งรูปร่างของพื้นที่จะมีผลในการจัดวางผังของพื้นที่ลงบนขอบเขตที่ดินโดยจะทำให้เกิดความหลากหลายในขั้นตอนการวางผัง

Node ที่เกี่ยวข้อง

-Sliding Number ใช้ node ในการปรับค่าของตัวเลขแบบเลื่อนซ้ายขวาเพื่อเพิ่มลดตัวเลขของค่าเปอร์เซ็นที่กำหนดไว้คือ 80% - 100% โดยคูณเข้าไปในค่าของ Column เพื่อทำให้ค่า Column สามารถควบคุมและปรับเปลี่ยนได้โดยผู้ใช้งาน โดยขั้นตอนนี้จะถูกกำหนดเป็น Is Input ที่ใช้ในการส่งข้อมูลต่อไปในโปรแกรม Refinery อีกด้วย



ภาพที่ 75 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 4

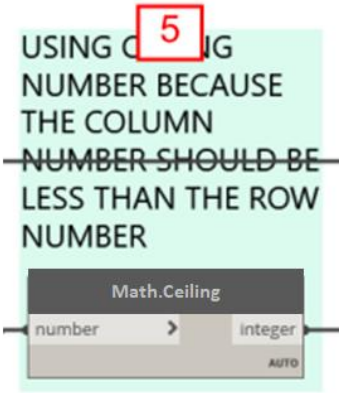


ภาพที่ 76 ภาพแสดง Is Input ของโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 4

หมายเลข 5

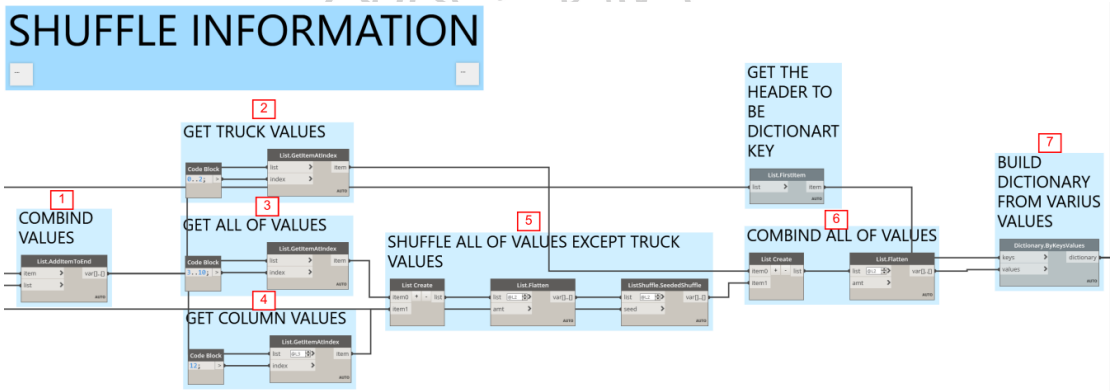
Node ที่เกี่ยวข้อง

-Math.Ceiling ใช้ปัดเศษของค่าจำนวนขึ้นให้เป็นจำนวนเต็ม เพื่อป้องกันในกรณีที่ค่าเป็น 0 เนื่องจากค่า Column มีค่าเป็น 0 ไม่ได้



ภาพที่ 77 ภาพแสดงโปรแกรมส่วนที่ 3 Create Column Values หมายเลข 5

3.2.2.2 สลับสับเปลี่ยนข้อมูล (Shuffle Information)



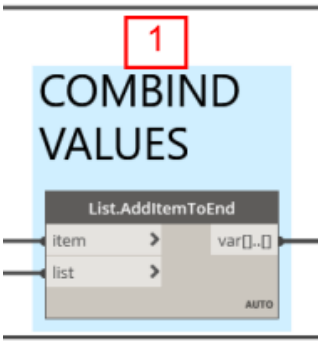
ภาพที่ 78 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Shuffle Information

ขั้นตอนนี้เป็นการนำข้อมูล จากหัวข้อ 3.2.2.1 นำเข้าข้อมูล (Import Information) มาทำการสลับและจัดเรียงลำดับของข้อมูลใหม่ใหม่ โดยหลักการในข้อ 3.1.2.2 แนวความคิดในขั้นตอนการวางผัง (Space Planning) ข้อย่อยที่ (1) ลำดับของการจัดเรียง (Permutation) เพื่อให้เกิดผลลัพธ์ของการจัดวางผังคลังสินค้าที่หลากหลายแตกต่างกันตามลำดับการจัดเรียงข้อมูลที่แตกต่างกัน

หมายเลข 1

Node ที่เกี่ยวข้อง

-List.AdditemToEnd เป็นการรวมข้อมูลส่วนที่ 1, 2, 3 ของขอ 3.2.2.1 ก่อนที่จะทำการจัดเรียงข้อมูลทั้งชุดใหม่



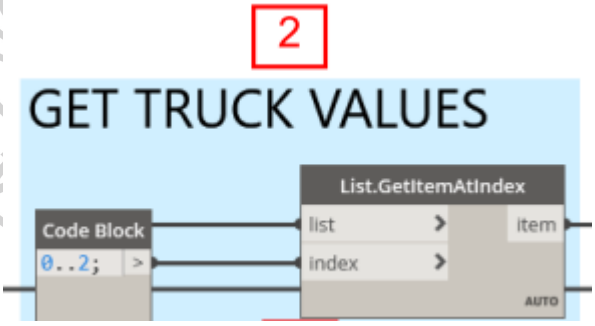
ภาพที่ 79 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 1

หมายเลข 2 – 4

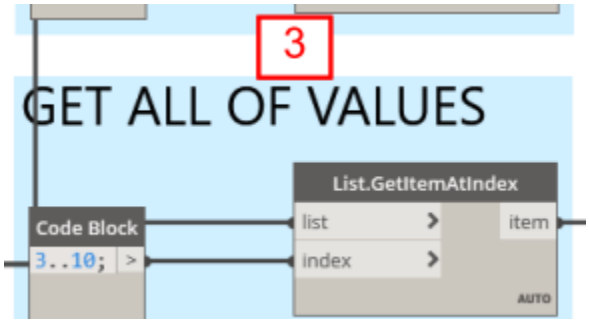
ขั้นตอนนี้เป็นการดึงข้อมูลส่วนที่ไม่ต้องจัดเรียงลำดับข้อมูลใหม่ออก คือ ข้อมูลพื้นที่ขอบเขตที่ดิน (Area) และ จำนวนของรถชนิดต่างๆที่ถูกใช้ในการขนส่ง (No. of Each Car Type) เนื่องจากไม่ใช่ข้อมูลที่มีผลต่อลำดับการจัดเรียงพื้นที่

Node ที่เกี่ยวข้อง

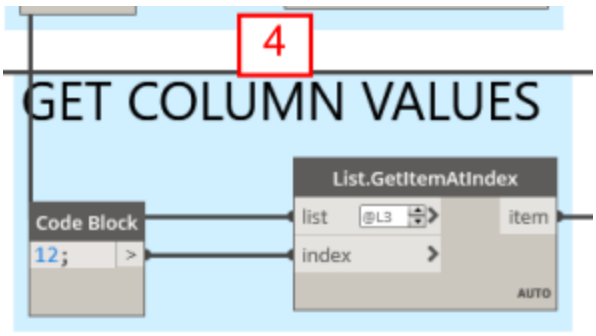
-List.GetitemAtIndex โหนดที่ใช้เลือกดึงข้อมูลแต่ละส่วนที่ต้องการโดยระบุตำแหน่งของข้อมูลที่อยู่ List



ภาพที่ 80 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 2



ภาพที่ 81 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 3

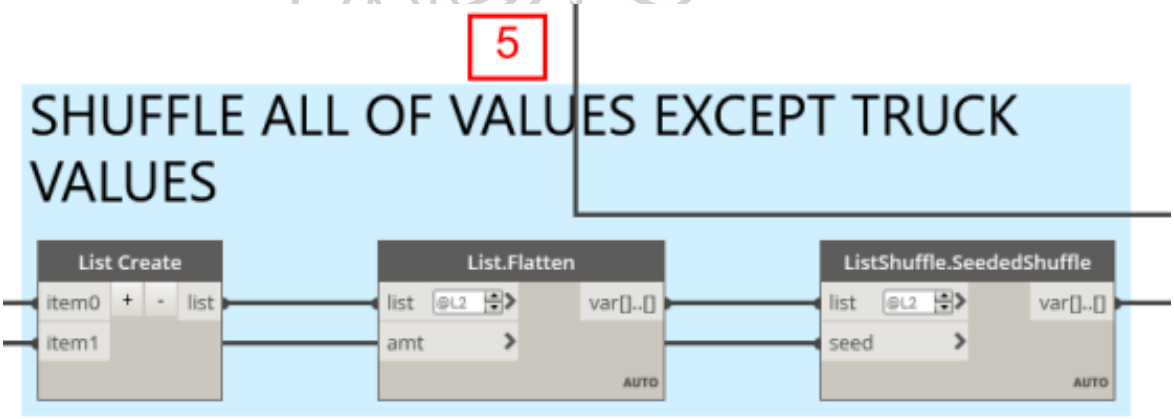


ภาพที่ 82 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 4

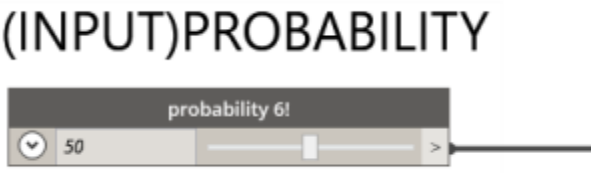
หมายเลข 5

<u>Node ที่เกี่ยวข้อง</u>

-ListShuffle.SeededShuffle ขั้นตอนนี้ของโปรแกรมจะเป็นจัดเรียงลำดับของข้อมูลที่ต้องการใหม่ โดยสามารถเลือกหรือกำหนดวิธีการของลำดับของการจัดเรียง (Permutation) ได้ในส่วน Input Seed โดยใช้ Node Number Slider เพื่อปรับจำนวนของวิธีการ



ภาพที่ 83 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 5



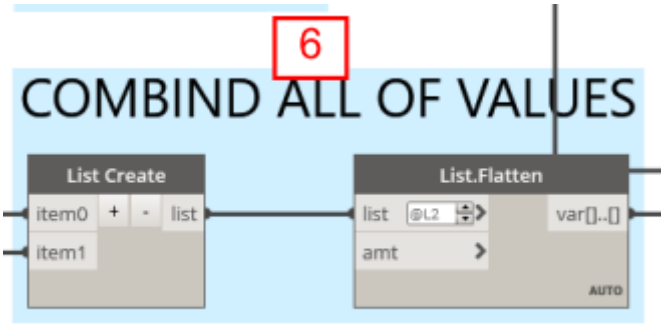
ภาพที่ 84 Input เพื่อควบคุมค่าให้เกิดความหลากหลายของผลลัพธ์ได้

หมายเลข 6

เป็นขั้นตอนในการรวมข้อมูลที่ดึงออกไปใน หมายเลข 4 รวมเข้ากับข้อมูลที่จัดเรียงใหม่จาก หมายเลข 5

Node ที่เกี่ยวข้อง

-List Create ใช้ในการสร้างชุดข้อมูลข้อมูลใหม่จากการรวมข้อมูลหลายๆชุด



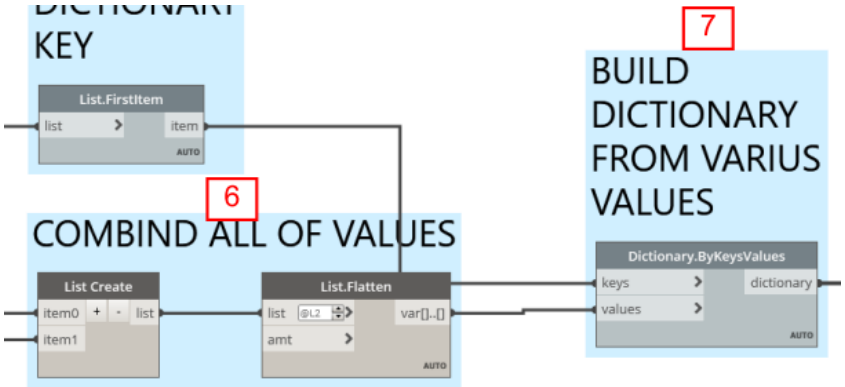
ภาพที่ 85 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 6

หมายเลข 7

ขั้นตอนนี้ของโปรแกรมเป็นการรวบรวมข้อมูลชุดสมบูรณ์ที่พร้อมจะนำไปสร้างพื้นที่และวางผังคลังสินค้าในขั้นตอนต่อไป โดยมีการสร้างหมวดหมู่ของข้อมูลในแต่ละชุดเพื่อง่ายต่อการเรียกใช้งาน

Node ที่เกี่ยวข้อง

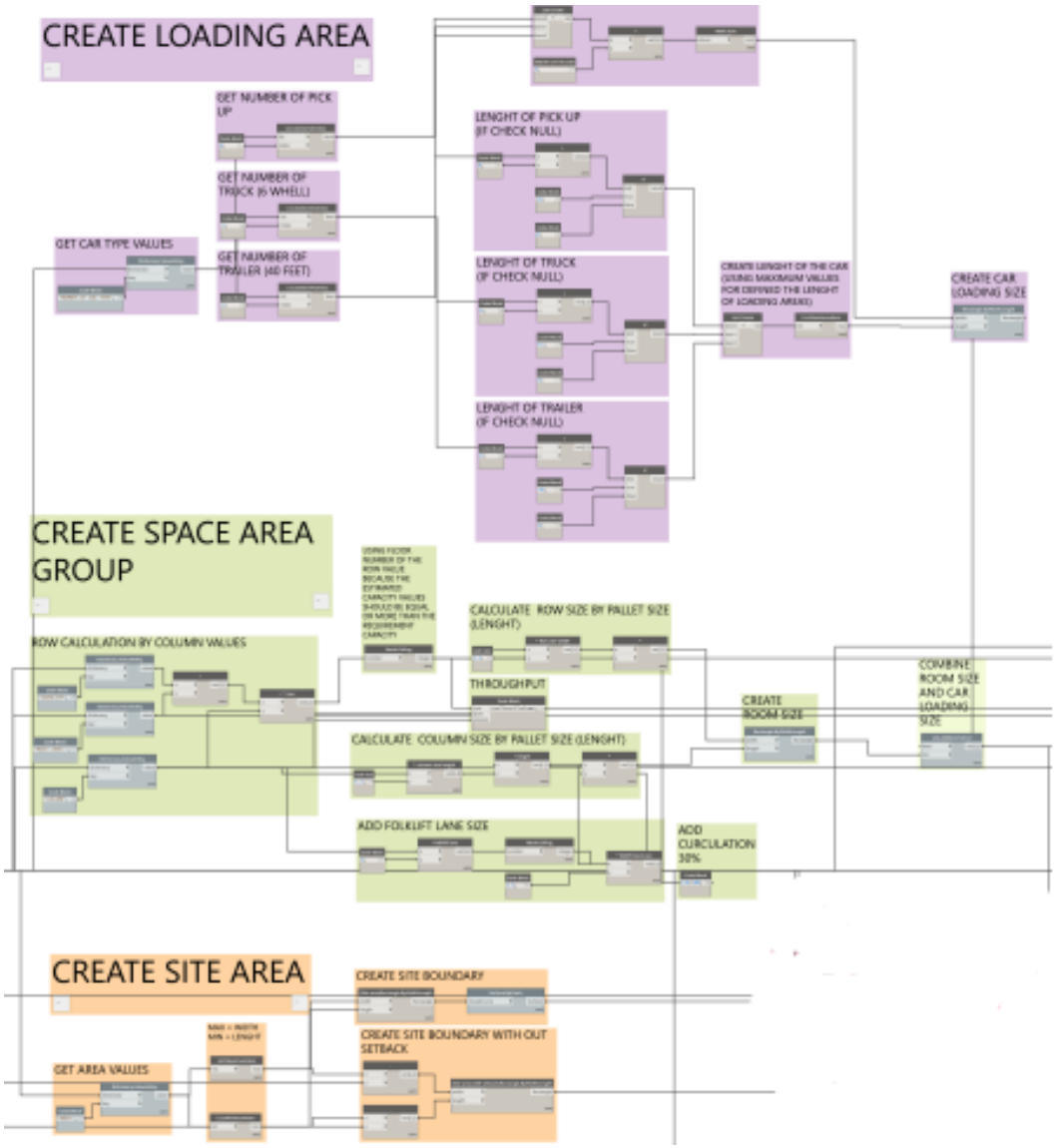
-Dictionary.ByKeysValues นำชื่อหัวข้อใน Excel ที่รับข้อมูลความต้องการลูกค้ามาสร้างเป็นหัวข้อของชุดข้อมูลทั้งหมดที่ถูกสร้างและจัดเรียงลำดับใหม่แล้ว



ภาพที่ 86 ภาพแสดงโปรแกรมส่วน Shuffle Information หมายเลข 6 - 7

3.2.2.3 สร้างพื้นที่จากข้อมูลขั้นตอนที่ 2 (Create Space)

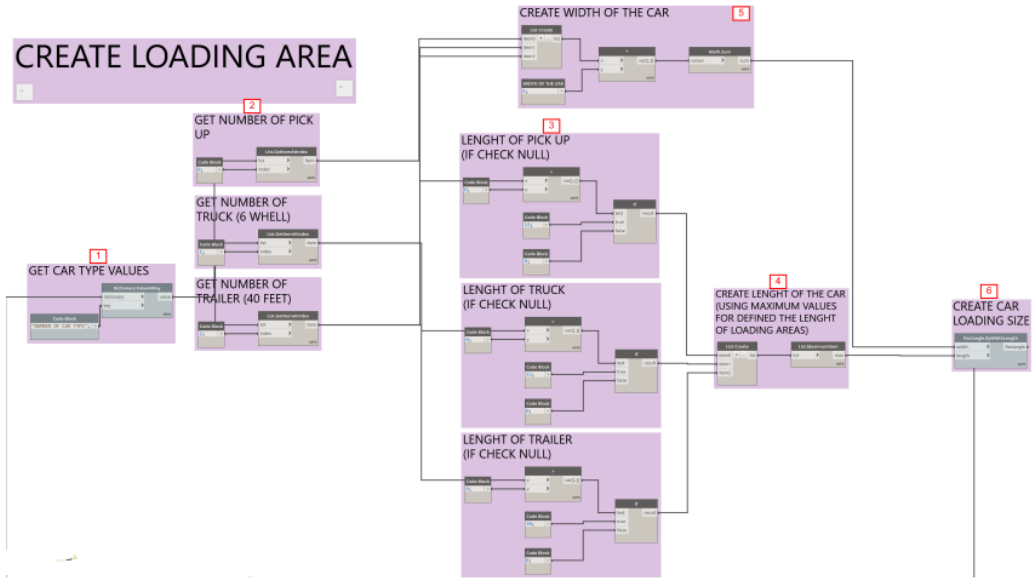
เป็นขั้นตอนการสร้างโมเดลเพื่อจำลองพื้นที่คลังสินค้า โดยสอดคล้องกับกระบวนการวางผังที่จะเกิดขึ้น ซึ่งมี 3 กลุ่มพื้นที่ ที่ถูกจำลองขึ้นโดยแบ่งตามข้อมูลความต้องการของลูกค้าที่นำมาใช้เป็นข้อมูลตั้งต้นในการสร้างพื้นที่ที่แตกต่างกัน ดังนี้



ภาพที่ 87 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Create Space

(1) กลุ่มพื้นที่ 1 Create Loading Area

ขั้นตอนการสร้างพื้นที่ของลานโหลดสินค้าของคลังสินค้าเพื่อใช้ในการวางผังคลังสินค้า โดยคำนวนการสร้างพื้นที่จากขนาดและจำนวนของรถแต่ละชนิดที่ต้องการใช้ในลานโหลดสินค้า อ้างอิงจากหัวข้อ 3.1.2.1 แนวความคิดในขั้นตอนการสร้างพื้นที่



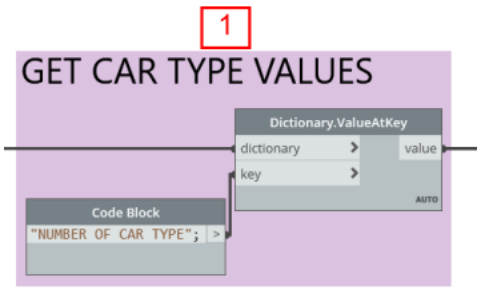
ภาพที่ 88 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area

หมายเลข 1

ดึงข้อมูลจำนวนของรถชนิดต่างๆที่ถูกใช้ในการขนส่ง เพื่อใช้ในการคำนวนสร้างพื้นที่ลานโหลดสินค้า

Node ที่เกี่ยวข้อง

-Dictionary.ValueAtKey ใช้ในการดึงข้อมูลจากหัวข้อของหมวดหมู่ที่สร้างไว้ ในขั้นตอน หมายเลข 7 ของข้อ 3.2.2.2



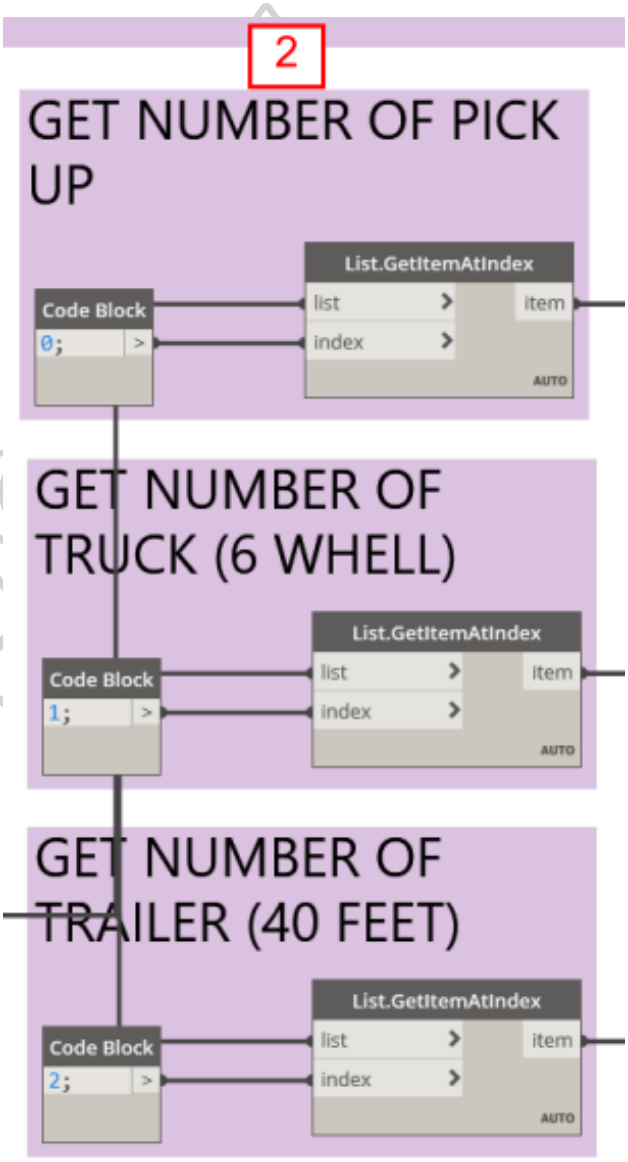
ภาพที่ 89 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 1

หมายเลข 2

แยกค่าของข้อมูลตามรถแต่ละประเภท โดยมีทั้งหมด 3 ประเภทได้แก่ รถ Trailer 40 Feet, รถ 10 ล้อ และ รถกะบะ Pick up เนื่องจากรถแต่ละประเภทมีขนาดที่แตกต่างกัน

Node ที่เกี่ยวข้อง

-List.GetitemAtIndex โหนดที่ใช้เลือกดึงข้อมูลแต่ละส่วนที่ต้องการโดยระบุตำแหน่งของข้อมูลที่อยู่ List



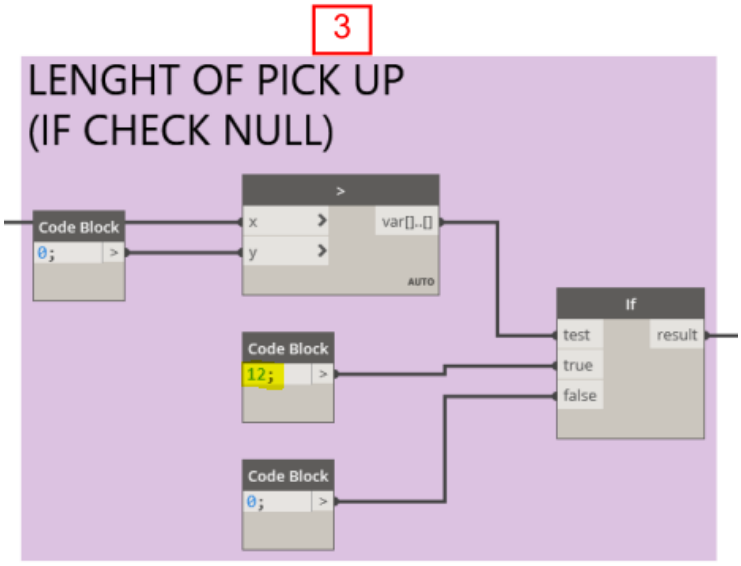
ภาพที่ 90 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 2

หมายเลข 3 - 4

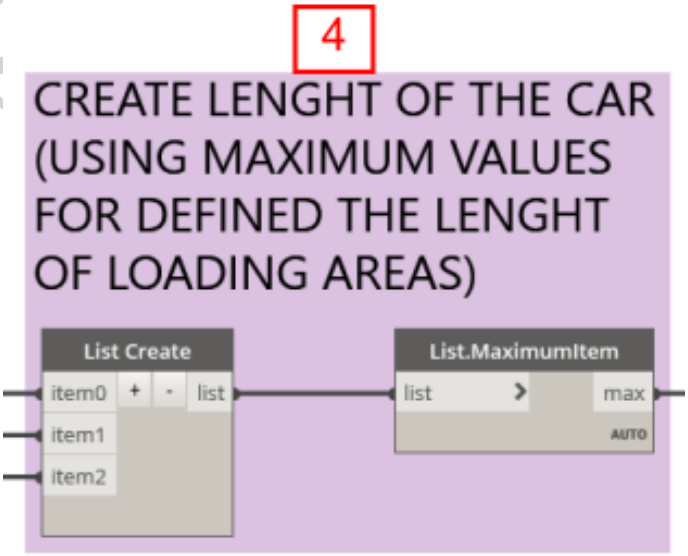
คำนวนหาค่าความยาวของพื้นที่ลานโหลดสินค้าซึ่งเท่ากับค่าความยาวของพื้นที่ซึ่งรถที่ยาวที่สุดต้องการใช้งาน

Node ที่เกี่ยวข้อง

-List.MaximumItem ใช้ในการเลือกค่าที่มากที่สุดของชุดข้อมูลออกมา



ภาพที่ 91 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 3



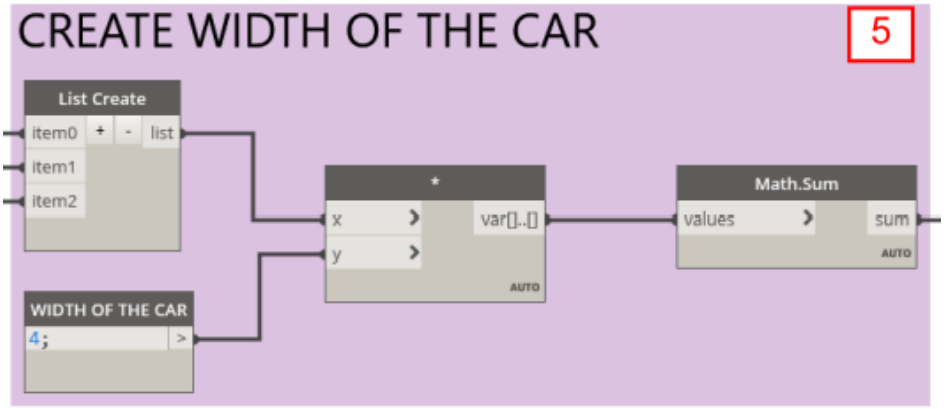
ภาพที่ 92 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 4

หมายเลข 5

คำนวนหาค่าความกว้างของพื้นที่ลานโหลดสินค้าซึ่งเท่ากับค่าความกว้างของพื้นที่ซึ่งรถแต่ละประเภทต้องการใช้งานคูณกับจำนวนรถที่ต้องใช้ในลานโหลดสินค้า

Node ที่เกี่ยวข้อง

-Math.Sum ใช้ในการบวกค่าของข้อมูลใน List เข้าด้วยกัน



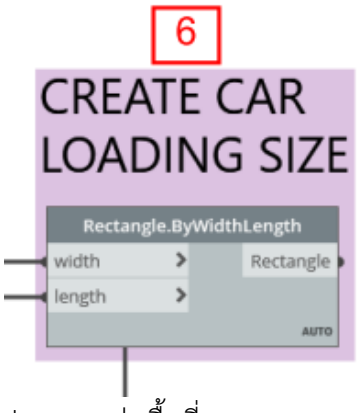
ภาพที่ 93 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 5

หมายเลข 6

คำนวนพื้นที่ลานโหลดสินค้าจากความกว้างและความยาวที่รถซึ่งใช้ในลานโหลดต้องการ

Node ที่เกี่ยวข้อง

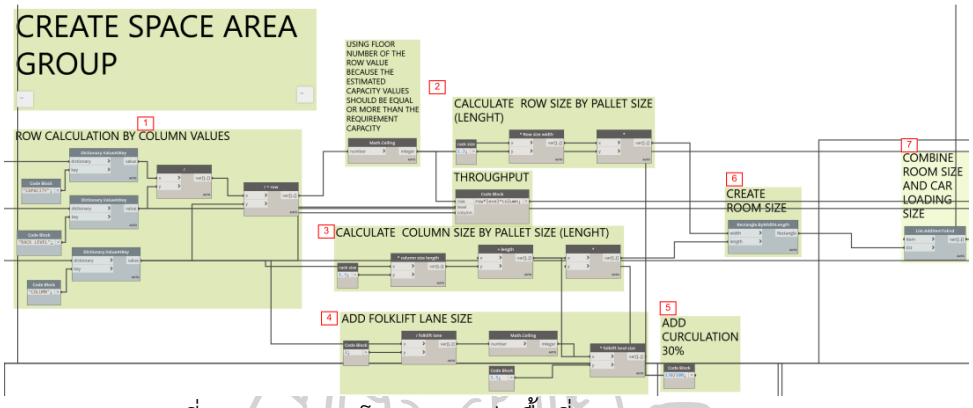
-Rectangle.ByWidthLength ใช้ในการสร้างพื้นที่สี่เหลี่ยมจากข้อมูลความกว้างและความยาว



ภาพที่ 94 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 1 Create Loading Area หมายเลข 6

(2) กลุ่มพื้นที่ 2 Create Space Area

ขั้นตอนการสร้างพื้นที่การใช้งานภายในของคลังสินค้าเพื่อใช้ในการวางผังคลังสินค้า อ้างอิงจากหัวข้อ 2.1.3.2 วิธีคำนวณพื้นที่คลังสินค้า



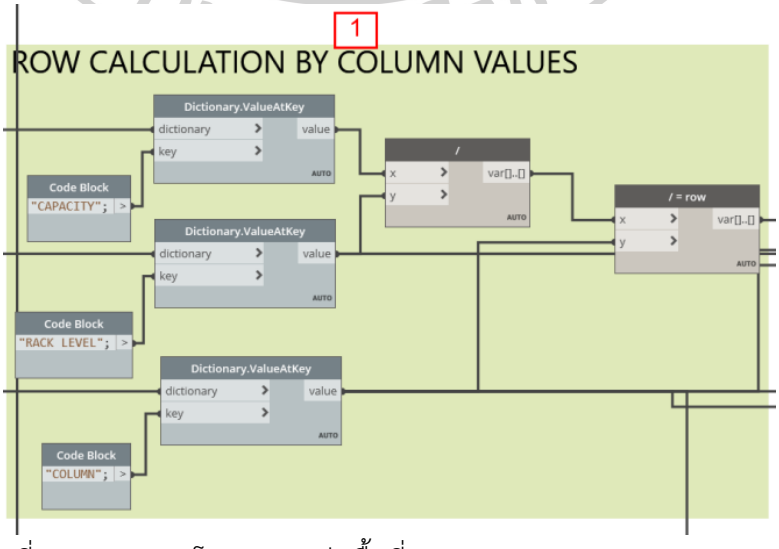
ภาพที่ 95 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area

หมายเลข 1

ดึงข้อมูลเพื่อให้ได้ค่าของ แถว (Row) และ แนว (Column) ของชั้นวางสินค้าในแต่ละพื้นที่การใช้งานเพื่อนำมาคำนวนหาพื้นที่ในแต่ละประเภทการใช้งาน

Node ที่เกี่ยวข้อง

-Dictionary.ValueAtKey ใช้ในการดึงข้อมูลจากหัวข้อของหมวดหมู่ที่สร้างไว้ ในขั้นตอน หมายเลข 7 ของข้อ 3.2.2.2



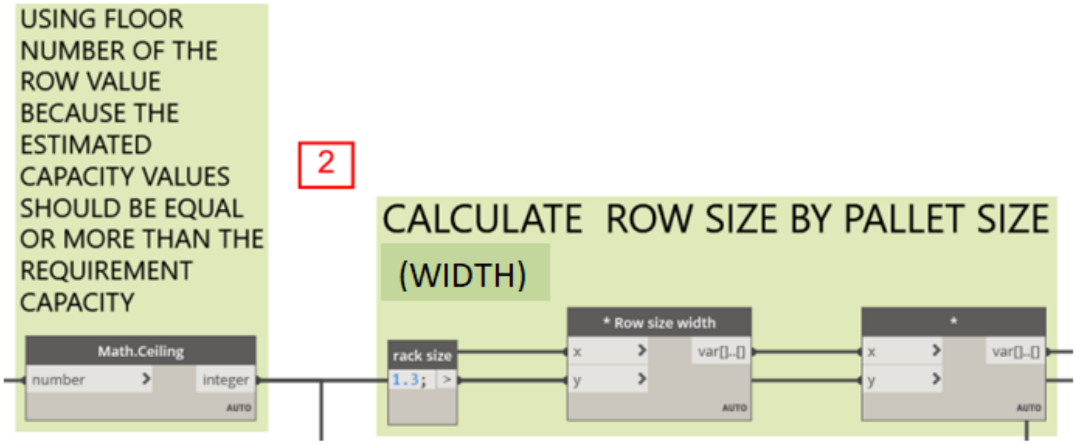
ภาพที่ 96 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 1

หมายเลข 2

คำนวนหาค่าความกว้างของพื้นที่ใช้งานภายในคลังสินค้า จากขนาดของชั้นวางสินค้าคูณกับจำนวนแถว (Row) ที่ใช้

Node ที่เกี่ยวข้อง

-Math.Ceiling ใช้ในการปัดเศษของค่าข้อมูลให้เป็นจำนวนเต็มโดยการปัดเศษขึ้น เนื่องจากค่าของ จำนวนแถว (Row) ต้องเป็นจำนวนเต็ม และเท่ากับหรือมากกว่าความต้องการของลูกค้าในเรื่องของการจัดเก็บสินค้า



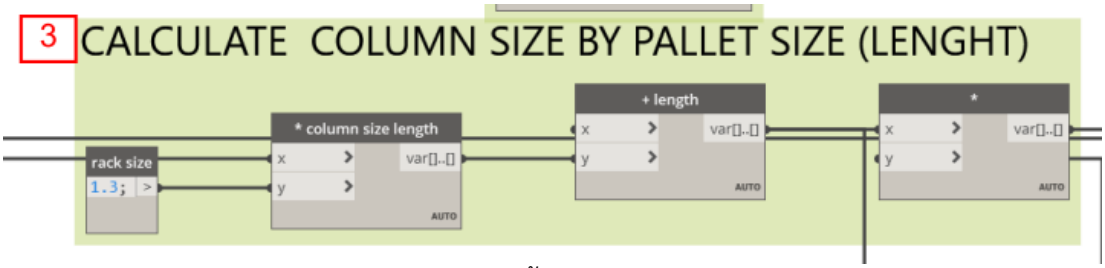
ภาพที่ 97 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 2

หมายเลข 3 – 4

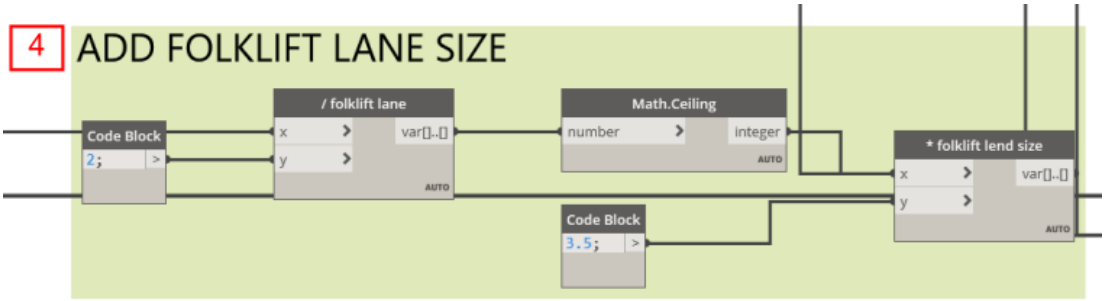
คำนวนหาค่าความยาวของพื้นที่ใช้งานภายในคลังสินค้า จากขนาดของชั้นวางสินค้าคูณกับจำนวนแนว (Column) ที่ใช้ (หมายเลข 3) และ ขนาดของพื้นที่รถยกสินค้าที่ใช้งาน (หมายเลข 4) โดยอ้างอิงจากหัวข้อ 2.1.3.2 วิธีคำนวณพื้นที่คลังสินค้า

Node ที่เกี่ยวข้อง

-Math.Ceiling ใช้ในการปัดเศษของค่าข้อมูลให้เป็นจำนวนเต็มโดยการปัดเศษขึ้น เนื่องจากค่าของ จำนวนแนว (Column) ต้องเป็นจำนวนเต็ม และเท่ากับหรือมากกว่าความต้องการของลูกค้าในเรื่องของการจัดเก็บสินค้า



ภาพที่ 98 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 3



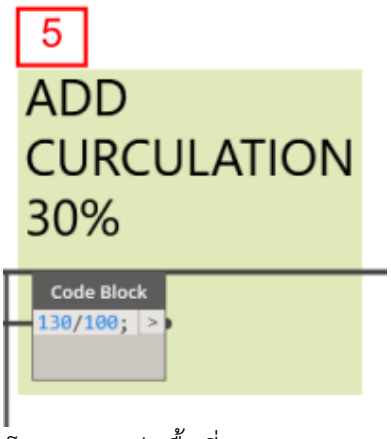
ภาพที่ 99 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 4

หมายเลข 5

ขั้นตอนนี้เป็นการเพิ่มค่า Curculation อีก 30% อ้างอิงจากข้อ 2.1.3.2 วิธีคำนวณพื้นที่คลังสินค้า

Node ที่เกี่ยวข้อง

-Code Block ใช้เขียนโปรแกรมขึ้นในกรณีที่ไม่จำเป็นต้องใช้ Node สำเร็จรูป



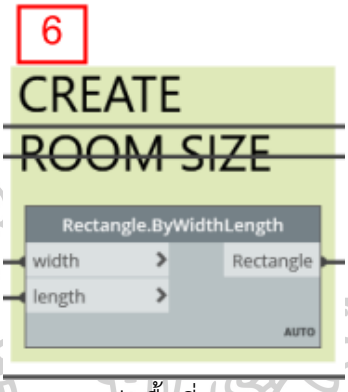
ภาพที่ 100 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 5

หมายเลข 6

คำนวนพื้นที่ใช้งานภายในคลังสินค้าจากความกว้างและความยาวที่ได้จากการคำนวน

Node ที่เกี่ยวข้อง

-Rectangle.ByWidthLength ใช้ในการสร้างพื้นที่สี่เหลี่ยมจากข้อมูลความกว้างและความยาว



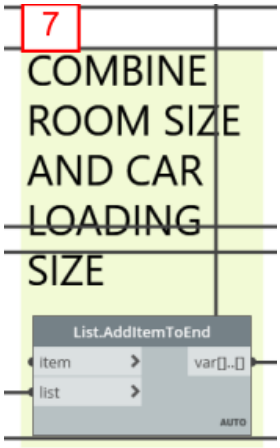
ภาพที่ 101 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 6

หมายเลข 7

รวมชุดข้อมูล กลุ่มพื้นที่ 1 และ 2 เข้าด้วยกันเพื่อนำไปจัดวางผังคลังสินค้าในขั้นตอนต่อไป

Node ที่เกี่ยวข้อง

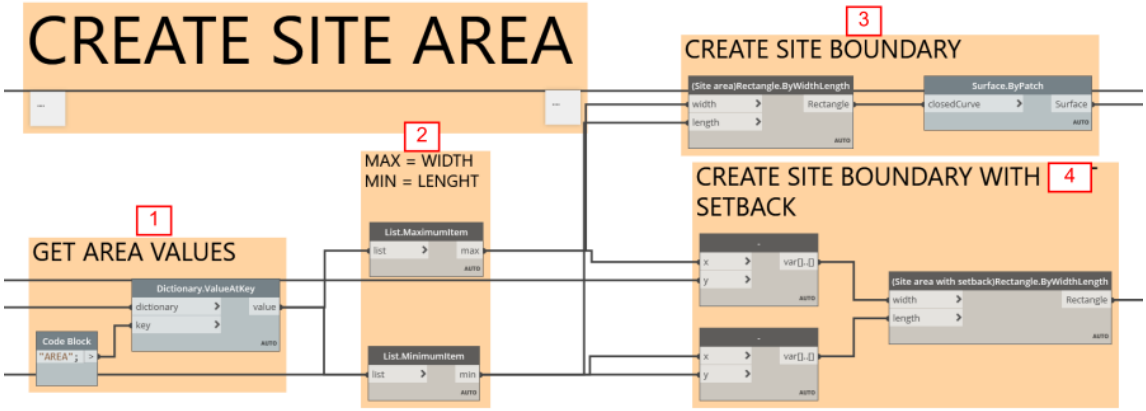
-List.AddItemToEnd ใช้ในการนำชุดข้อมูลชุดหนึ่งมาต่อท้ายชุดข้อมูลอีกชุดหนึ่ง



ภาพที่ 102 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 2 Create Space Area หมายเลข 7

(3) กลุ่มพื้นที่ 3 Create Site Area

ขั้นตอนการสร้างขอบเขตพื้นที่ดินของคลังสินค้าเพื่อใช้ในการวางผังคลังสินค้า อ้างอิงจากหัวข้อ 3.1.2.1 แนวความคิดในขั้นตอนการสร้างพื้นที่



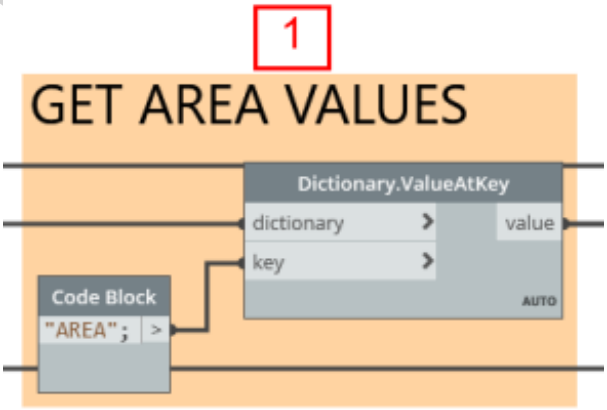
ภาพที่ 103 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area

หมายเลข 1

ดึงค่าความยาวและความกว้างของที่ดินเพื่อสร้างพื้นที่ขอบเขตที่ดิน

<u>Node ที่เกี่ยวข้อง</u>

-Dictionary.ValueAtKey ใช้ในการดึงข้อมูลจากหัวข้อของหมวดหมู่ที่สร้างไว้ ในขั้นตอน หมายเลข7 ของข้อ 3.2.2.2



ภาพที่ 104 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area หมายเลข 1

หมายเลข 2

จำแนกค่าสูงที่สุดในชุดข้อมูลเพื่อกำหนดเป็นค่า ความยาว และจำแนกค่าต่ำที่สุดในชุดข้อมูลเพื่อกำหนดเป็นค่า ความกว้าง

Node ที่เกี่ยวข้อง

-List.MaximumItem ใช้ในการหาค่าสูงที่สุดในชุดข้อมูล

-List.MinimumItem ใช้ในการหาค่าต่ำที่สุดในชุดข้อมูล

ภาพที่ 105 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area หมายเลข 2

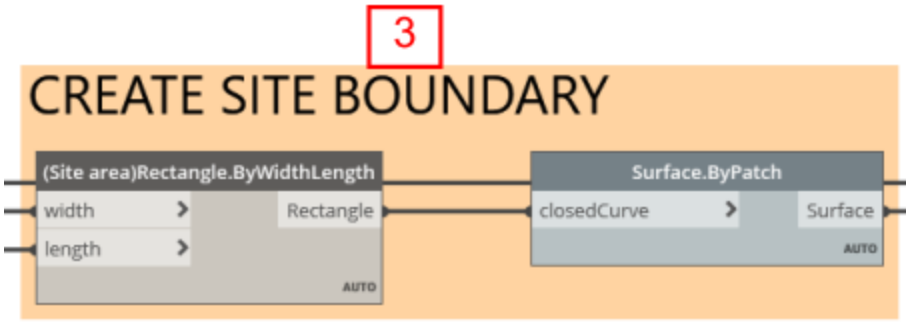
หมายเลข 3

คำนวนพื้นที่ขอบเขตที่ดินจากความกว้างและความยาวที่ได้จากการข้อมูล

Node ที่เกี่ยวข้อง

-Rectangle.ByWidthLength ใช้ในการสร้างพื้นที่สี่เหลี่ยมจากข้อมูลความกว้างและความยาว

-Surface.ByPatch ใช้ในการสร้างพื้นผิวของพื้นที่ที่กำหนด



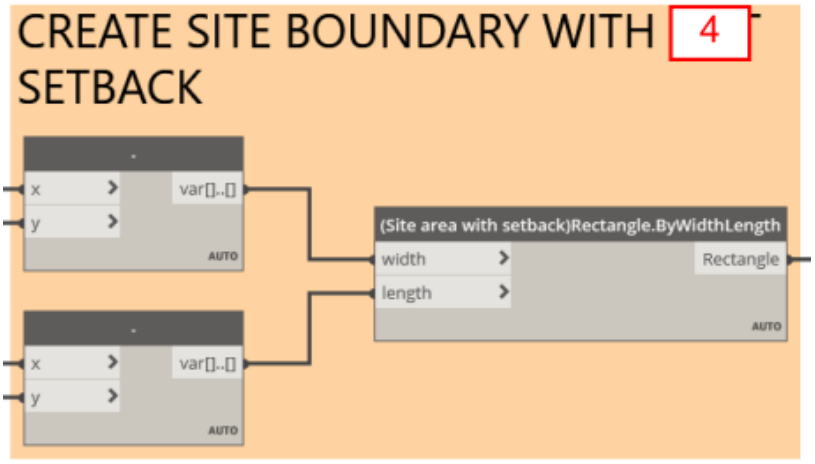
ภาพที่ 106 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area หมายเลข 3

หมายเลข 4

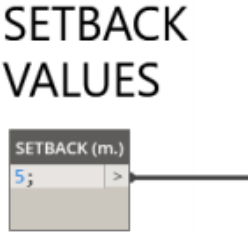
สร้างพื้นที่จริงของขอบเขตที่ดินที่หักลบกับระยะร่นตามกฎหมาย เพื่อใช้เป็นขอบเขตขอบอาคารในวางผังคลังสินค้า

Node ที่เกี่ยวข้อง

-Rectangle.ByWidthLength ใช้ในการสร้างพื้นที่สี่เหลี่ยมจากข้อมูลความกว้างและความยาวที่หักลบระยะร่นแล้ว



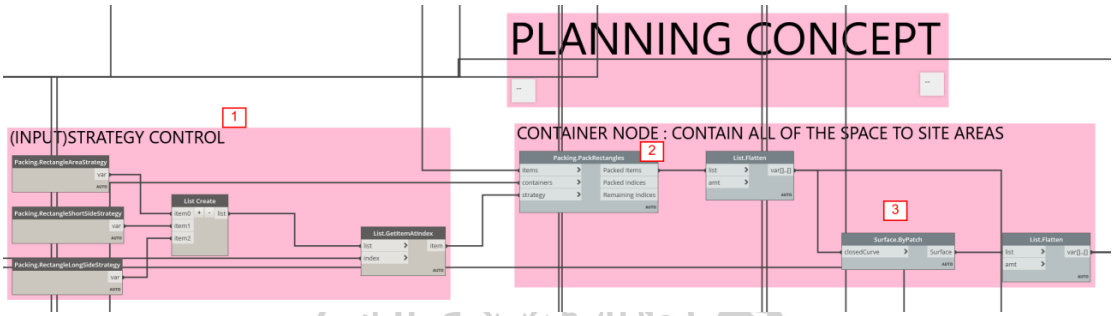
ภาพที่ 107 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area หมายเลข 4



ภาพที่ 108 ภาพแสดงโปรแกรมกลุ่มพื้นที่ 3 Create Site Area ในส่วนกรอกค่าระยะร่น

3.2.2.4 จัดเรียงวางผังพื้นที่จากขั้นตอนที่ 3 (Planning)

เป็นขั้นตอนการวางผังโดยใช้หลักการการบรรจุสิ่งของลงกล่อง กล่าวคือ พื้นที่จำลองที่ถูกสร้าง กลุ่มพื้นที่ 1 Loading Area และ กลุ่มพื้นที่ 2 Space Area เสมือนสิ่งของที่จะถูกบรรจุ ลงในกลุ่มพื้นที่ 3 Site Area ซึ่งคือตัวแทนของกล่องที่รอรับการบรรจุ อ้างอิงจากหลักการข้อ 3.1.2.2 แนวความคิดในขั้นตอนการวางผัง (Space Planning)



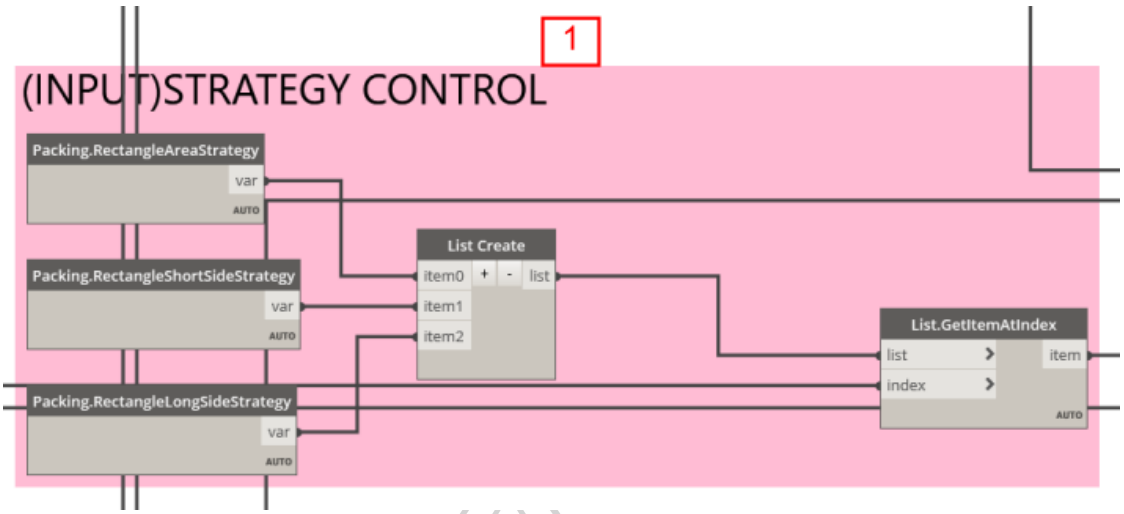
ภาพที่ 109 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Planning

หมายเลข 1

โปรแกรมในส่วนนี้เป็นการสร้างข้อมูลของค่าของกระบวนการจัดเรียงพื้นที่ (Strategy) ที่มีผลต่อผลลัพธ์ของการจัดเรียงหรือวางผังของพื้นที่ตามหลักการวางผังแบบการบรรจุสิ่งของ (Items) ลงกล่อง (Container) 3 รูปแบบ เพื่อให้เกิดเป็น Input ที่สามารถเลือกรูปแบบที่ต้องการให้เกิดกับการจัดวางผังคลังสินค้าได้ อ้างอิงจากข้อ 3.1.2.2 แนวความคิดในขั้นตอนการวางผัง (Space Planning)

<u>Node ที่เกี่ยวข้อง</u>

- Packing.RectangleAreaStrategy ใช้เพื่อเลือกกระบวนการในการจัดเรียงพื้นที่โดยพิจารณาการจัดแบบสุ่มวางทั้งด้านยาวและด้านกว้างของพื้นที่แต่ละพื้นที่ต่อกัน
- Packing. RectangleShortSideAreaStrategy ใช้เพื่อเลือกกระบวนการในการจัดเรียงพื้นที่โดยพิจารณาการจัดวางด้านกว้างของพื้นที่แต่ละพื้นที่ต่อกันก่อนเป็นความสำคัญแรก
- Packing. RectangleLongSideAreaStrategy ใช้เพื่อเลือกกระบวนการในการจัดเรียงพื้นที่โดยพิจารณาการจัดวางด้านยาวของพื้นที่แต่ละพื้นที่ต่อกันก่อนเป็นความสำคัญแรก



ภาพที่ 110 ภาพแสดงโปรแกรมส่วน Planning หมายเลข 1

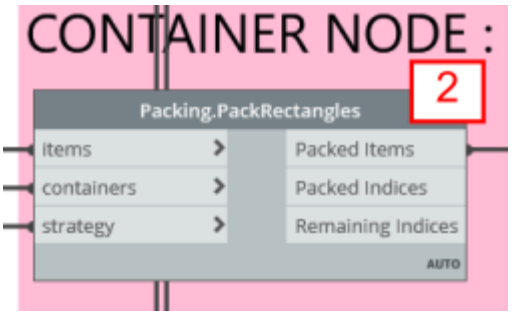
ภาพที่ 111 Input เพื่อให้ผู้ใช้สามารถกำหนดค่าในส่วน Strategy ได้

หมายเลข 2

จัดเรียงพื้นที่โดยหลักการการบรรจุสิ่งของลง (Packing) บวกกับพื้นฐานทางคณิตศาสตร์เรื่องความน่าจะเป็น (Probability) โดยเลียนแบบการบรรจุสิ่งของ (Items) ลงกล่อง (Container)

<u>Node ที่เกี่ยวข้อง</u>

-Packing.PackRectangles ใช้ในการจัดวางพื้นที่ลงในขอบเขตที่กำหนดตาม โดยสามารถกำหนดกระบวนการในการจัดเรียง (Strategy) ได้



ภาพที่ 112 ภาพแสดงโปรแกรมส่วน Planning หมายเลข 2

หมายเลข 3

สร้างพื้นผิวของพื้นที่ที่ถูกจัดวางผังเรียบร้อยแล้วเพื่อใช้ในโปรแกรมขั้นตอนต่อไป

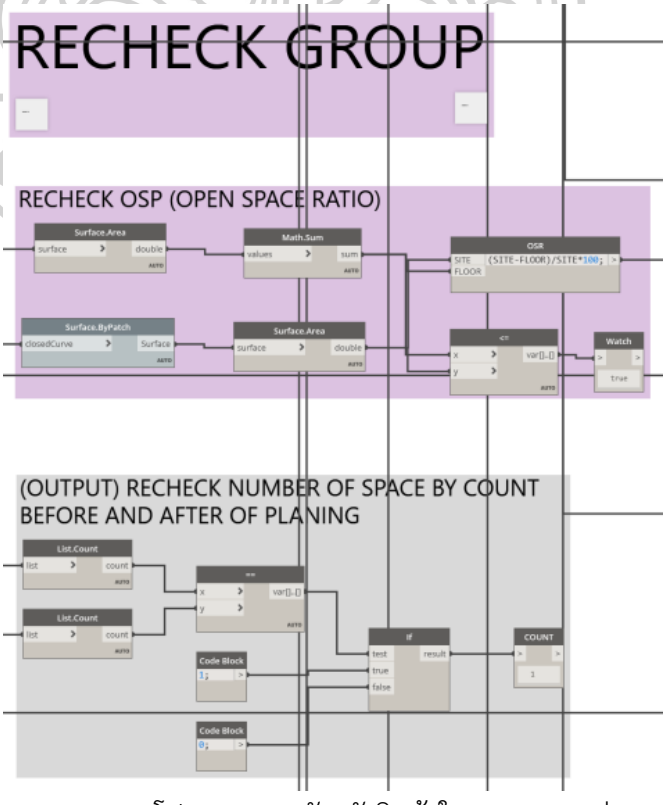
Node ที่เกี่ยวข้อง

-Surface.ByPatch ใช้ในการสร้างพื้นผิวของพื้นที่ที่กำหนด

ภาพที่ 113 ภาพแสดงโปรแกรมส่วน Planning หมายเลข 3

3.2.2.5 ตรวจสอบความถูกต้องของผังที่ถูกจัดเรียงในขั้นตอนที่ 4 (Recheck)

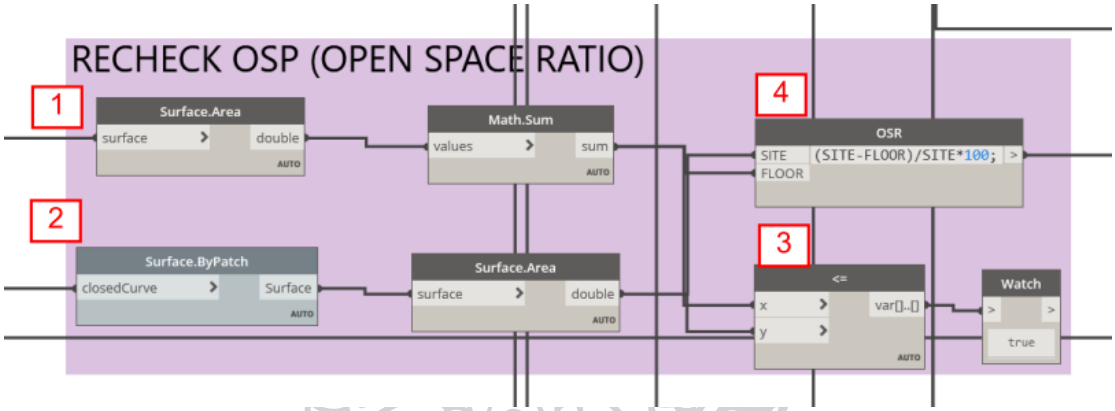
เป็นขั้นตอนในการตรวจสอบผลลัพธ์ที่เกิดขึ้นหลังจากการวางผังสำเร็จเพื่อความถูกต้องในการวางผังคลังสินค้าโดยแบ่งเป็น 2 ส่วน คือ ส่วนที่ 1 Recheck OSP (Open Space Ratio) และส่วนที่ 2 Recheck Number Of Space



ภาพที่ 114 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Recheck

(1) ส่วนที่ 1 Recheck OSR (Open Space Ratio)

เป็นการตรวจสอบค่า OSR หรือ ค่าพื้นที่เปิดโล่งต่อพื้นที่ดินตามกฎหมาย โดยการคำนวนและแสดงค่าออกมาเป็นข้อความเพื่อให้ผู้ใช้งานได้ตรวจสอบว่าการวางผังคลังสินค้าในแต่ละแบบนั่นมีค่า OSR ตามกฎหมายหรือไม่



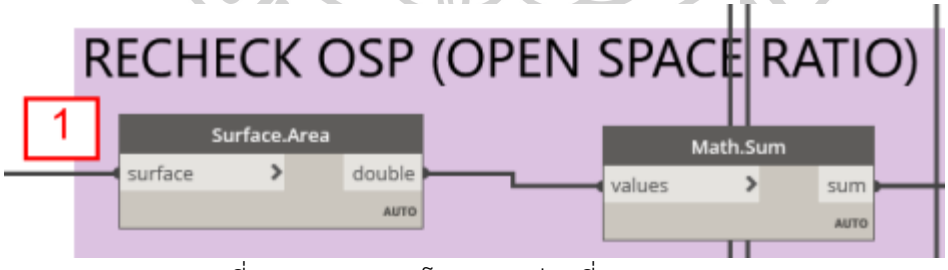
ภาพที่ 115 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR

หมายเลข 1

คำนวนหาค่าพื้นที่รวมของอาคารคลังสินค้า

Node ที่เกี่ยวข้อง

-Surface.Area ใช้ในการหาค่าพื้นที่จากพื้นผิวที่เลือก



ภาพที่ 116 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR หมายเลข 1

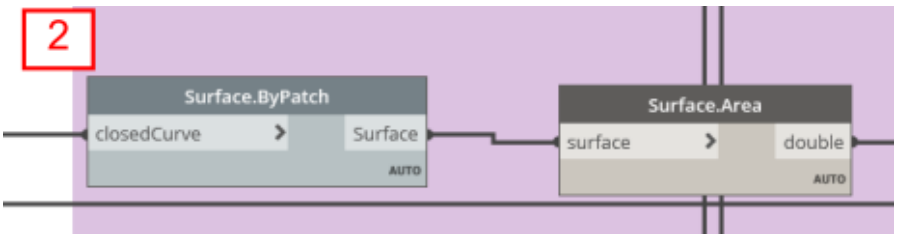
หมายเลข2

หาค่าพื้นที่ของพื้นที่ดินที่ใช้ในการวางผังคลังสินค้า

Nodeที่เกี่ยวข้อง

-Surface.ByPatch ใช้ในการสร้างพื้นผิวจากเส้นรอบรูป

-Surface.Area ใช้ในการหาค่าพื้นที่จากพื้นผิวที่เลือก



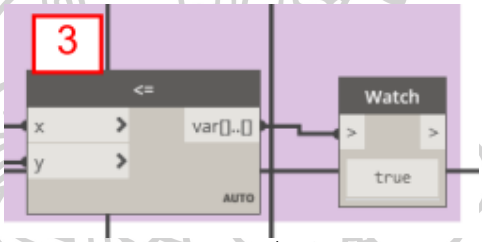
ภาพที่ 117 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR หมายเลข 2

หมายเลข 3

เช็คค่าของข้อมูลหมายเลข 1 และ 2 พื้นที่อาคารคลังสินค้า ควรจะเล็กกว่าหรือเท่ากับ พื้นที่ดิน

Node ที่เกี่ยวข้อง

-<= ใช้ในการเปรียบเทียบข้อมูลสองชุดว่ามีค่าน้อยกว่าหรือเท่ากับ กันหรือไม่



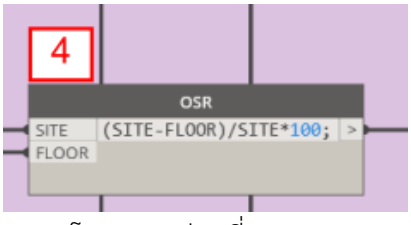
ภาพที่ 118 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR หมายเลข 3

หมายเลข4

คำนวนค่า OSR

Nodeที่เกี่ยวข้อง

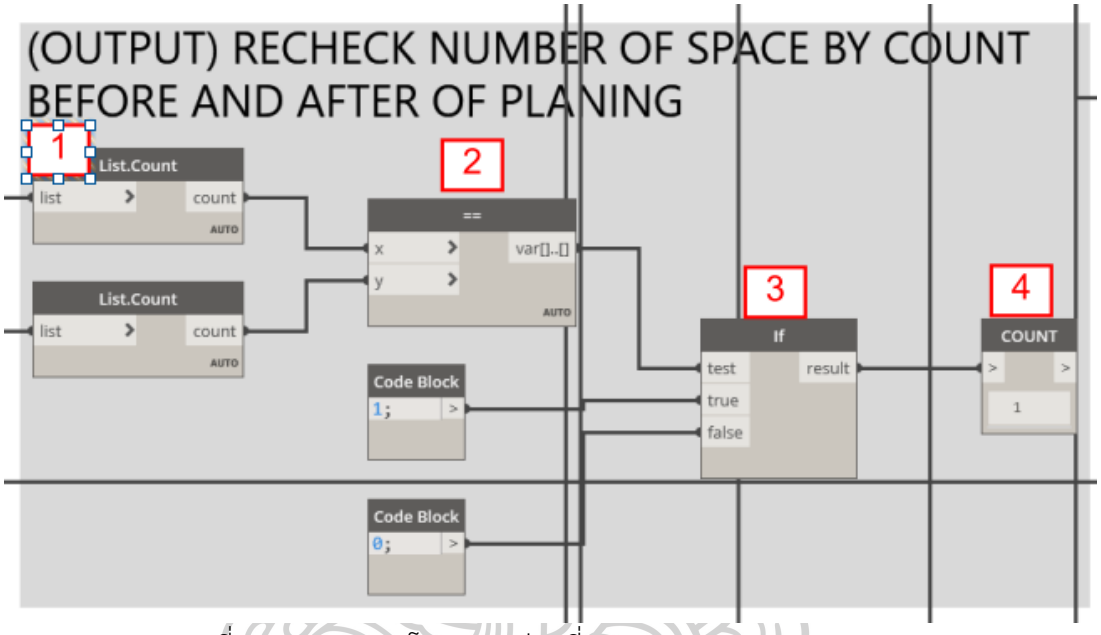
-Code Block ใช้ในการเขียนโปรแกรมเป็นสูตรการคำนวนค่า OSR โดยการนำข้อมูลหมายเลข 1 คือค่าพื้นที่อาคารคลังสินค้า หารกับ ค่าหมายเลข 2 คือค่าพื้นที่ดิน คูณ 100



ภาพที่ 119 ภาพแสดงโปรแกรมส่วนที่ 1 Recheck OSR หมายเลข 4

(2) ส่วนที่ 2 Recheck number of space

เป็นการเช็คจำนวนพื้นที่ ก่อนและหลังการวางผังว่ามีจำนวนเท่ากันหรือไม่เพื่อตรวจสอบให้ไม่เกิดกรณีที่มีพื้นที่หายไป หรือ ไม่ครบตามจำนวนพื้นที่ใช้งานภายในคลังสินค้า



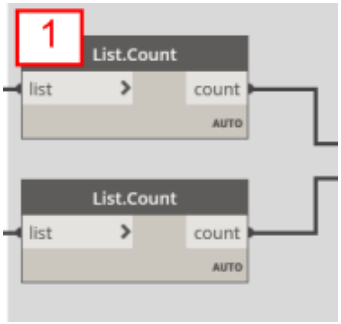
ภาพที่ 120 ภาพแสดงโปรแกรมส่วนที่ 2 Recheck number of space

หมายเลข 1

นับจำนวนพื้นที่ใช้งานภายในคลังสินค้า ก่อน และ หลัง การวางผังคลังสินค้า

Node ที่เกี่ยวข้อง

List.Count ใช้ในการนับจำนวนข้อมูลภายใน List



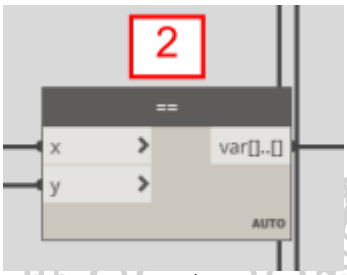
ภาพที่ 121 ภาพแสดงโปรแกรมส่วนที่ 2 Recheck number of space หมายเลข 1

หมายเลข 2

ตรวจสอบว่าจำนวนพื้นที่ใช้งานภายในคลังสินค้า ก่อน และ หลัง การวางผังคลังสินค้ามีจำนวนเท่ากันหรือไม่

Node ที่เกี่ยวข้อง

-== ใช้ในการตรวจสอบข้อมูลชุดหนึ่งกับอีกชุดหนึ่งว่ามีค่าเท่ากันหรือไม่



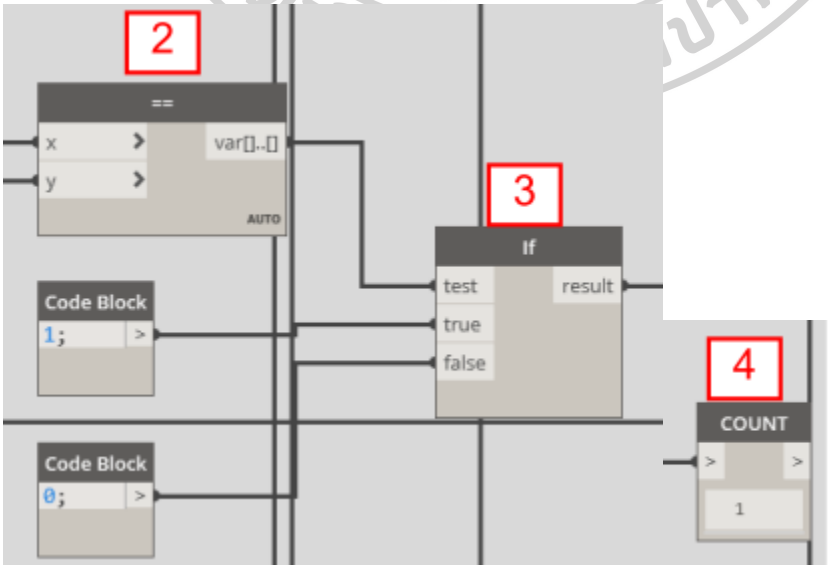
ภาพที่ 122 ภาพแสดงโปรแกรมส่วนที่ 2 Recheck number of space หมายเลข 2

หมายเลข3 – 4

รายงานผลลัพธ์ของถึงจำนวนพื้นที่ใช้งานภายในคลังสินค้า ก่อน และหลัง การวางผังคลังสินค้า ว่ามีจำนวนเท่ากันหรือไม่ โดยในส่วนหมายเลข 4 นี้ยังถูกกำหนดเป็น Is Input เพื่อส่งต่อไปประมวลผลในโปรแกรม Refinery ต่อไป

Nodeที่เกี่ยวข้อง

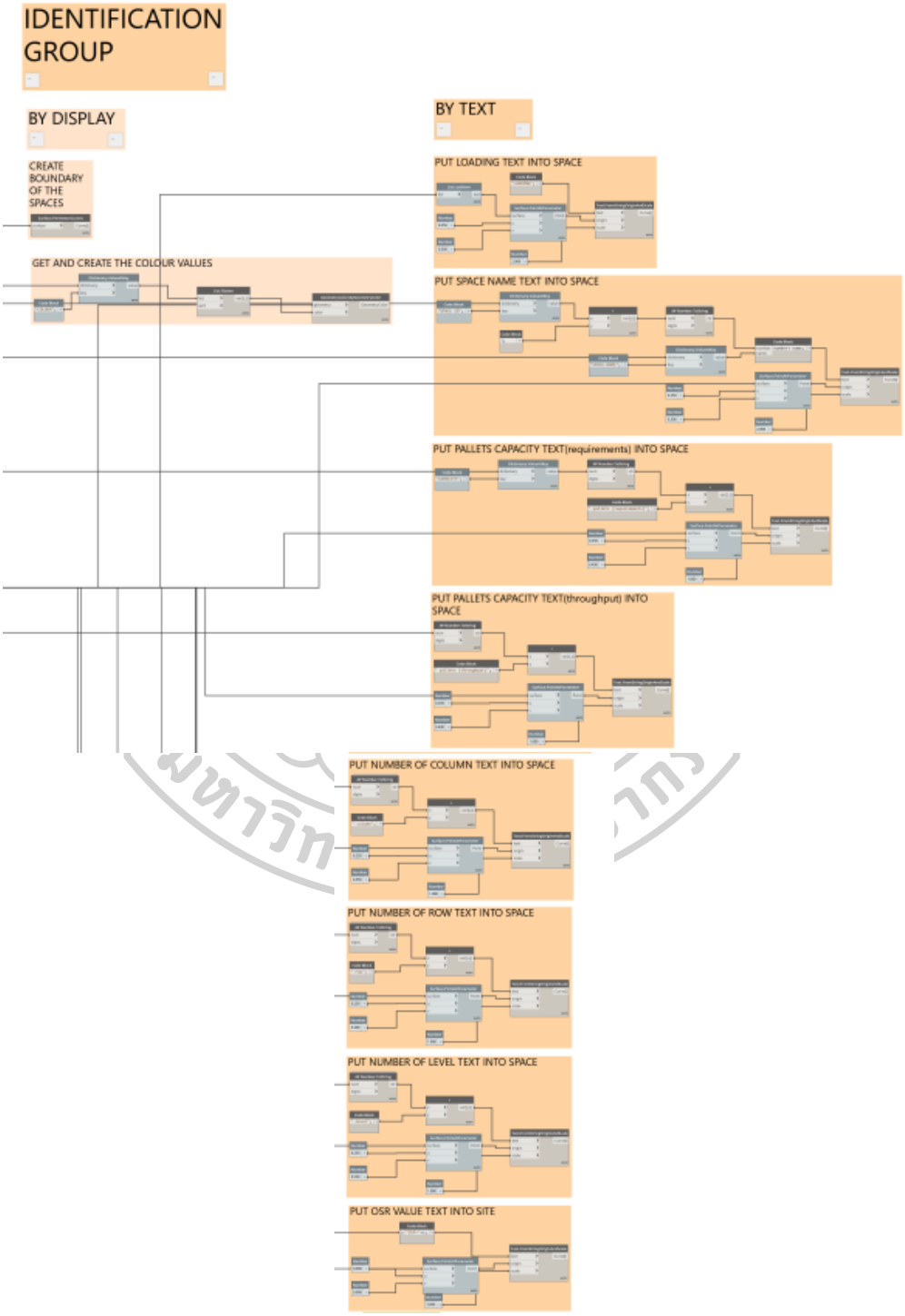
-If ใช้ในการสร้างเงื่อนไขเพื่อให้เกิดผลลัพธ์ ถูก (True) และ ผิด (False)



ภาพที่ 123 ภาพแสดงโปรแกรมส่วนที่ 2 Recheck number of space หมายเลข 2 - 4

3.2.2.6 ระบุข้อมูลจำเพาะให้แสดงผลลงบนผังที่ถูกจัดเรียงในขั้นตอนที่ 4 (Identification)

เป็นส่วนที่จะทำให้พื้นที่มีการแสดงผลเพื่อการแยกแยะแต่ละพื้นที่ออกจากกันได้ โดยการแยะแยะนี้จะแบ่งออกเป็น 2 กลุ่ม



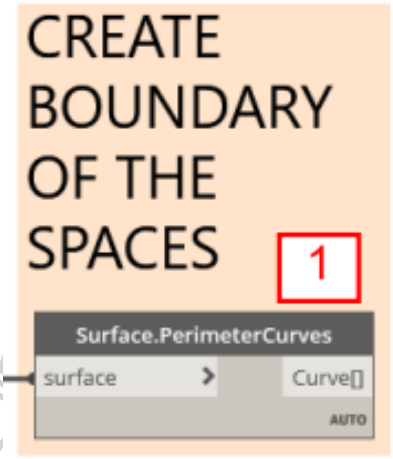
ภาพที่ 124 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Identification

(1) การแสดงผลด้วยสีและเส้น

1.1 สร้างเส้นขอบเขตของแต่ละพื้นที่ใช้งานภายในอาคารคลังสินค้า

Node ที่เกี่ยวข้อง

-Surface.PerimetryCurves ใช้ในการสร้างเส้นขอบเขตของแต่ละพื้นที่



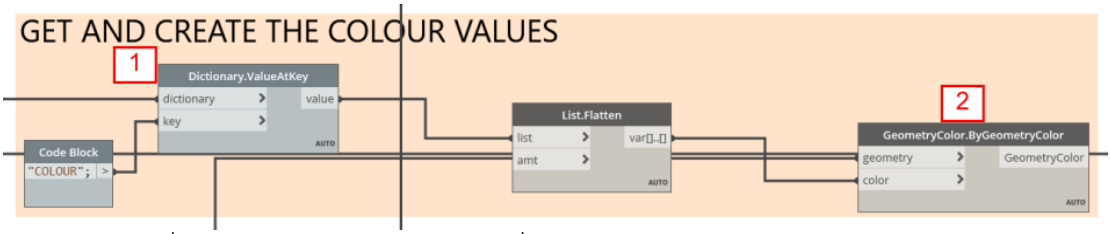
ภาพที่ 125 ภาพแสดงโปรแกรมส่วนที่ 1 การแสดงผลด้วยสีและเส้นหมายเลข 1

1.2 สร้างสีในแต่ละพื้นที่ใช้งานภายในอาคารคลังสินค้า

Node ที่เกี่ยวข้อง

-Dictionary.ValueAtKey ใช้ในการดึงค่าสีที่กำหนดไว้

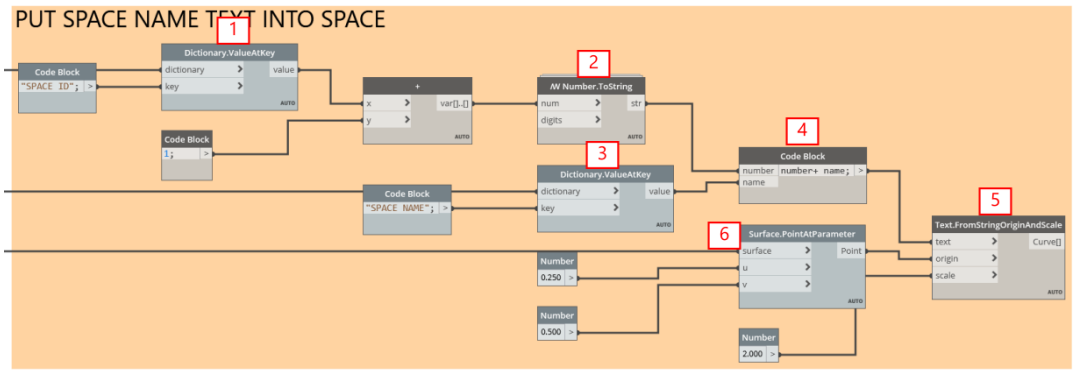
-GeometryColor.ByGeometryColor ใช้ในการแสดงผลค่าสีลงบนแต่ละพื้นที่ที่กำหนด



ภาพที่ 126 ภาพแสดงโปรแกรมส่วนที่ 1 การแสดงผลด้วยสีและเส้นหมายเลข 1-2

(2) การแสดงผลด้วยข้อความ

เนื่องจากหลักการของการแสดงตัวอักษรในโมเดลนั่นมีความเหมือนกัน ไม่ว่าจะเป็นการต้องการแสดงอักษรของค่าในส่วนใด ดังนั้นจะยกตัวอย่างอธิบายการแสดงตัวอักษรของค่าชื่อพื้นที่



ภาพที่ 127 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความ

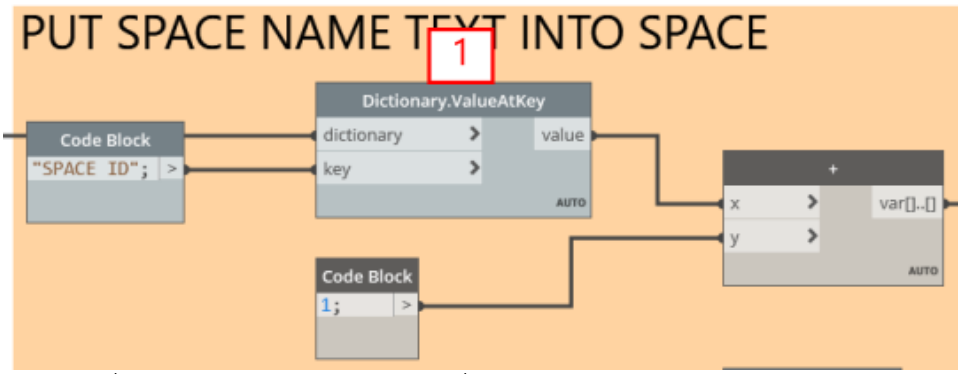
หมายเลข 1

ดึงข้อมูลที่ต้องการแสดงเป็นข้อความจากชุดข้อมูลที่เตรียมไว้ หากข้อมูลเป็นชุดตัวอักษรสามารถดึงมาแสดงผลได้เลย ส่วนในกรณีที่ข้อความที่ต้องการแสดงเป็นข้อมูลตัวเลข (Number) ต้องแปลงค่าก่อน

<u>Node ที่เกี่ยวข้อง</u>

-Dictionary.ValueAtKey ใช้ในการดึงข้อมูลจากหัวข้อของหมวดหมู่ที่สร้างไว้ ในขั้นตอน หมายเลข 7 ของข้อ 3.2.2.2

-Number.ToString ใช้ในการแปลงค่าข้อมูลตัวเลข (Number) ก่อนนำมาแสดงผลเป็นข้อความบนโปรแกรม



ภาพที่ 128 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 1

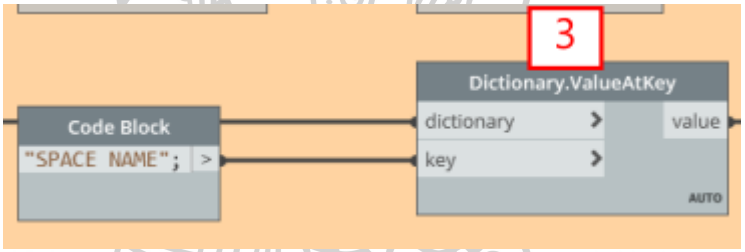
หมายเลข 2 - 3

ในกรณีที่ข้อความที่ต้องการแสดงเป็นข้อมูลตัวเลข (Number) ต้องแปลงค่าก่อน

Node ที่เกี่ยวข้อง

-Number.ToString ใช้ในการแปลงค่าข้อมูลตัวเลข (Number) ก่อนนำมาแสดงผลเป็นข้อความบนโปรแกรม

ภาพที่ 129 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 2



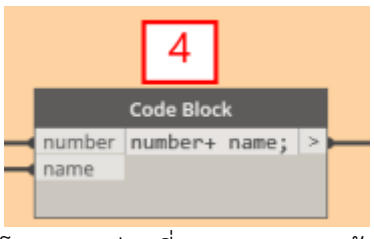
ภาพที่ 130 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 3

หมายเลข 4

นำข้อมูลที่เป็นตัวอักษรและตัวเลขมาต่อกัน

Node ที่เกี่ยวข้อง

-Code Block ใช้ในเขียนคำสั่งทางโปรแกรมในการกำหนดตัวแปรเพื่อดึงข้อมูลให้ต่อกัน



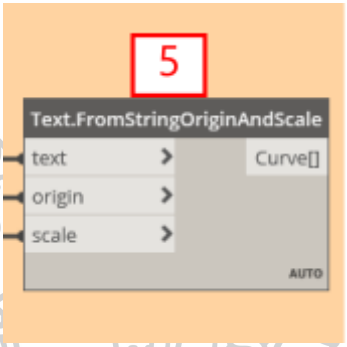
ภาพที่ 131 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 4

หมายเลข 5

สร้างการแสดงผลข้อความที่ในจุดที่ต้องการแสดงผลรวมทั้งขนาดของข้อความที่ต้องการแสดงผล

Node ที่เกี่ยวข้อง

-Text.FromStringOriginAndScale ใช้ในการสร้างการแสดงผลข้อความโดยสามารถกำหนดจุดและขนาดของข้อความที่ต้องการได้



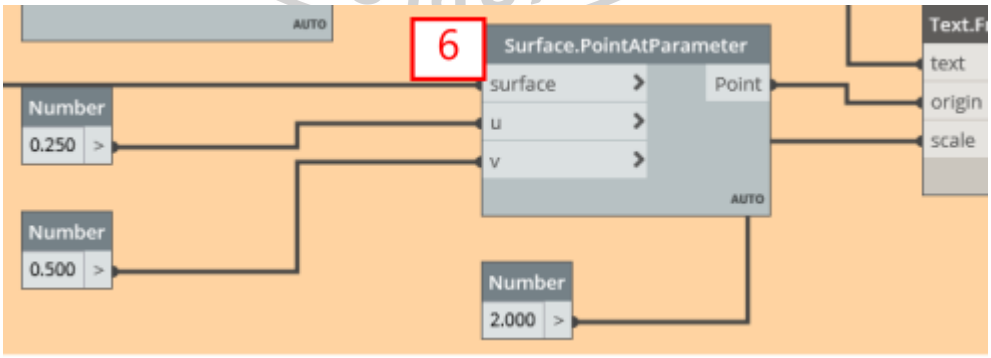
ภาพที่ 132 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 5

หมายเลข 6

ระบุตำแหน่งบนพื้นผิวของพื้นที่ในโมเดลที่ต้องการให้ข้อความแสดงผลเพื่อเป็นการกำหนดจุดให้กับโปรแกรมส่วน หมายเลข 5

Node ที่เกี่ยวข้อง

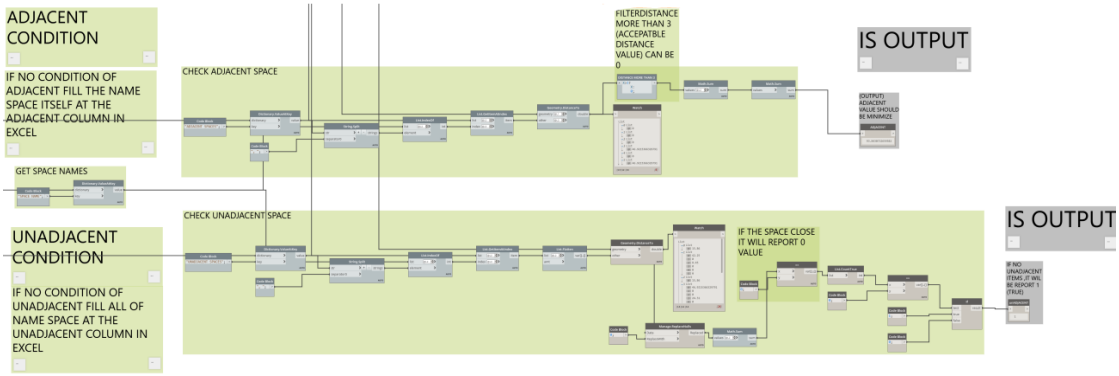
-Surface.PointAtParameter ใช้ในการระบุตำแหน่งบนพื้นผิวของพื้นที่



ภาพที่ 133 ภาพแสดงโปรแกรมส่วนที่ 2 การแสดงผลด้วยข้อความหมายเลข 6

3.2.2.7 สร้างเงื่อนไขเพื่อให้เกิดค่าของข้อมูลในแต่ละครั้งของผลลัพธ์ในการวางผังที่เกิดขึ้น (Set Condition)

ทางผู้ทำการวิจัยจะขอเรียกส่วนนี้ว่า Condition หรือเงื่อนไข ซึ่งเป็นส่วนในการสร้างเงื่อนไข ขึ้นมาจากตัวโปรแกรมเพื่อให้เกิดผลลัพธ์เพื่อนำไปกำหนดเป้าหมายในการกระบวนการ Generative Design ในโปรแกรม Refinery

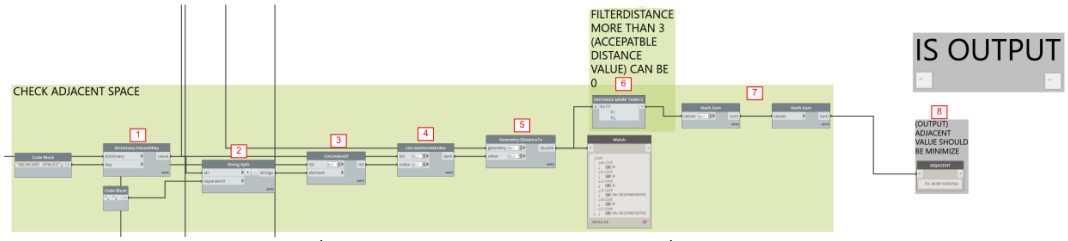


ภาพที่ 134 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Set Condition

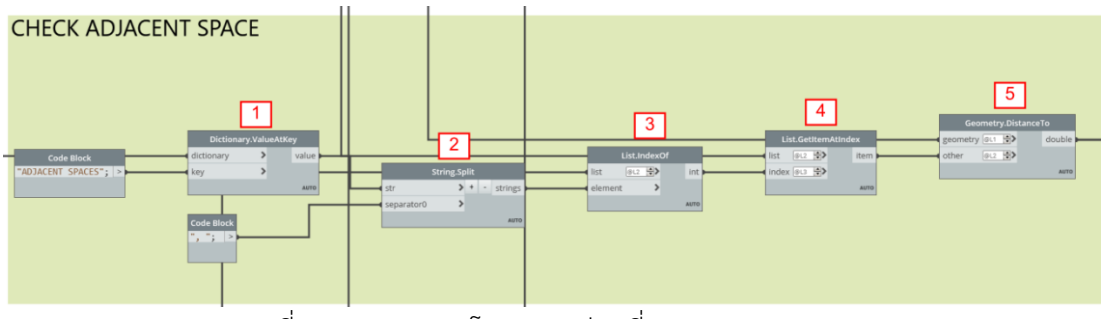
โดยเงื่อนไขนั้นจะเป็นเงื่อนไขของการวางผังคลังสินค้า โดยการจัดวางพื้นที่แต่ละพื้นที่ให้อยู่ในตำแหน่งใดนั่นมีผลเนื่องมากจาก ผลิตภัณฑ์หรือสินค้าที่ถูกจัดเก็บในห้องนั่นๆ เป็นชนิดใด กล่าวโดยสรุปคือใช้หลักการในการวางของสองสิ่งหรือมากกว่าสองสิ่ง ใกล้หรือไกลกันได้อย่างไร ด้วยคุณสมบัติและความต้องการที่แตกต่างกันของแต่ละพื้นที่

การควรใกล้กันของพื้นที่ (Adjacent) และ การไม่ควรใกล้กันของพื้นที่ (Nonadjacent) จึงถูกนำมาเป็นเงื่อนไขในการวางผังคลังสินค้าโดยให้เกิดค่าของผลลัพธ์ในแต่ละเงื่อนไขเพื่อใช้ในการกำหนดเป้าของผลลัพธ์ที่ต้องการให้เกิดได้

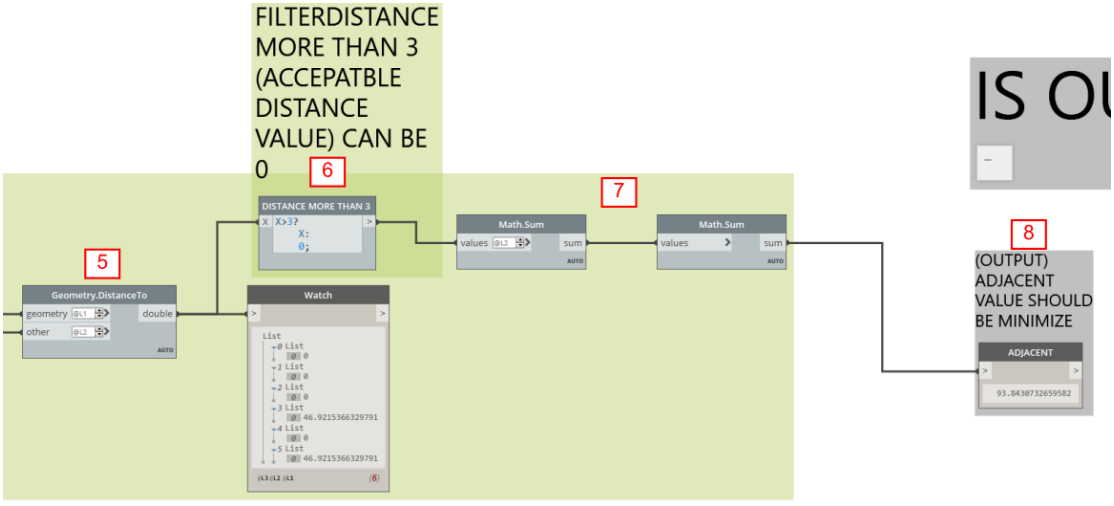
(1) Adjacent



ภาพที่ 135 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent



ภาพที่ 136 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent (ขยาย)



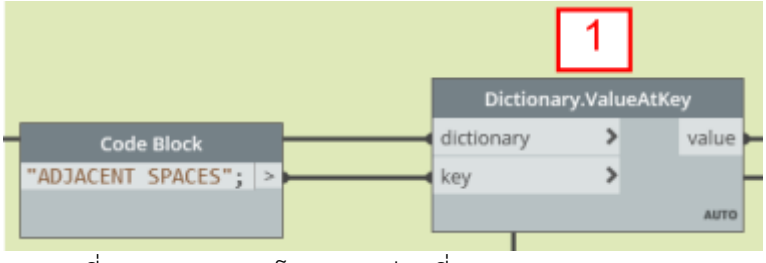
ภาพที่ 137 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent (ขยาย)

หมายเลข 1

ดึงข้อมูลค่าความต้องการของการวางผังคลังสินค้าในส่วนของความต้องการที่จะให้พื้นที่ไหนใกล้หรือติดกัน (ADJACENT)

<u>Node ที่เกี่ยวข้อง</u>

-Dictionary.ValueAtKey ใช้ในการดึงข้อมูลจากหัวข้อของหมวดหมู่ที่สร้างไว้ ในขั้นตอน หมายเลข 7 ของข้อ 3.2.2.2



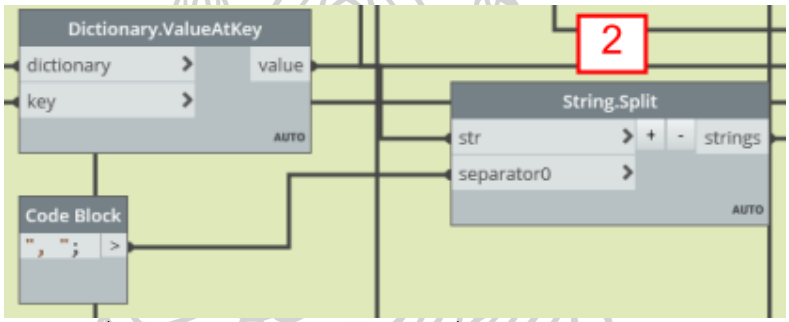
ภาพที่ 138 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 1

หมายเลข 2

เนื่องจากข้อมูลที่ดึงมากนั้นมีลักษณะข้อมูลที่ไม่จำกัดปริมาณการกรอกของข้อมูลในหัวข้อนั้นๆซึ่งแต่ละลำดับของข้อมูลนั่นจะถูกไล่เรียงลงไปตาม Column แต่ข้อมูลที่กรอกในแต่ละ Column จะมีมากกว่า 1 ค่าโดยแต่ละค่าจะถูกคั่นด้วยสัญลักษณ์ คอมม่า (,) เช่น Coldroom, Chillroom, Staging เป็นต้น อ้างอิงจากหัวข้อ 3.2.1.3 ประเภทที่ 3 จึงต้องแยกค่าข้อมูลที่กรอกออกจากกันด้วยสัญลักษณ์ที่คั่นอยู่

Node ที่เกี่ยวข้อง

-String.Split ใช้แบ่งข้อมูลเดี่ยวข้อมูลเดียวที่ได้มาให้กลายเป็นข้อมูลเดี่ยวหลายๆข้อมูล



ภาพที่ 139 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 2

หมายเลข 3 - 4

ค่าที่ได้มาจากหมายเลข 2 จะเพียงชื่อของห้องหรือพื้นที่ต่างๆที่ต้องการให้ติดกับพื้นที่ที่กำหนด ดังนั้นจึงต้องนำชื่อที่ได้มาคืนค่าให้ตรงกับข้อมูลพื้นที่หลังจากการวางผังเรียบร้อยแล้วในขั้นตอน หมายเลข7 ของข้อ 3.2.2.2

Node ที่เกี่ยวข้อง

-List.IndexOf ใช้ระบุตำแหน่งของข้อมูลที่อยู่ใน List

-List.GetItemAtIndex ใช้ดึงข้อมูลโดยการระบุตำแหน่งที่อยู่ใน List

ภาพที่ 140 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 3

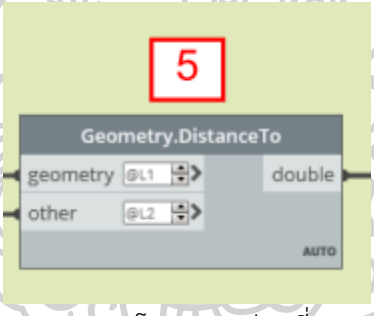
ภาพที่ 141 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 4

หมายเลข 5

โปรแกรมส่วนนี้จะทำการวัดระยะระหว่างพื้นที่ที่ต้องการให้ใกล้กันจากผลลัพธ์การวางผังที่เกิดขึ้น

Node ที่เกี่ยวข้อง

-Geometry.DistanceTo ใช้ในการวัดระยะทางจากพื้นที่หนึ่งไปยังอีกพื้นที่หนึ่ง



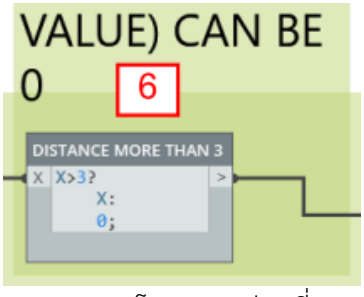
ภาพที่ 142 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 5

หมายเลข 6

จากระยะทางที่เป็นผลลัพธ์จากหมายเลข 5 ถ้ามีระยะระหว่างพื้นที่ไม่เกิน 3 ยังถือว่าพื้นที่นั่นมีความใกล้กันอยู่โดยทนค่าเป็น 0 แทน

Node ที่เกี่ยวข้อง

-Code Block ใช้ในการเขียนคำสั่งเงื่อนไขในการคัดกรอกค่าที่ข้อมูลต้องมากกว่า 3 ถึงจะส่งออกไปได้ หากน้อยกว่า 3 ให้ส่งออกเป็นค่า 0



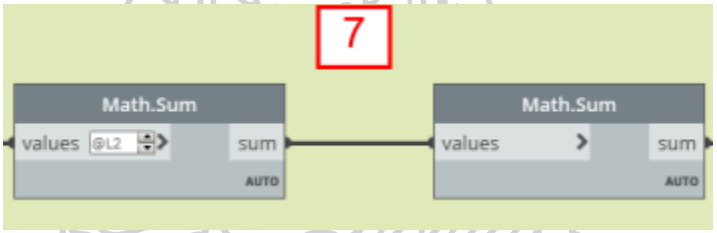
ภาพที่ 143 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 6

หมายเลข 7

บวกค่าระยะห่างทั้งหมดของพื้นที่ที่กำหนดเพื่อเป็นค่าของผลลัพธ์นั่นๆ

Node ที่เกี่ยวข้อง

-Math.Sum ใช้ในการรวมค่าของข้อมูล



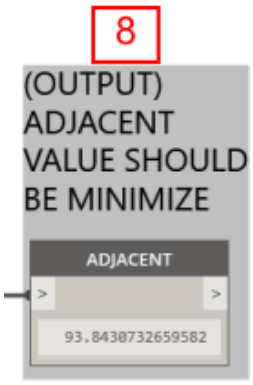
ภาพที่ 144 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 7

หมายเลข 8

แสดงค่าของผลลัพธ์ที่ทำได้ตามเงื่อนไข การควรใกล้กันของพื้นที่ (ADJACENT) ซึ่งในส่วนนี้จะถูกกำหนดเป็น Output ใน Refinery ต่อไป

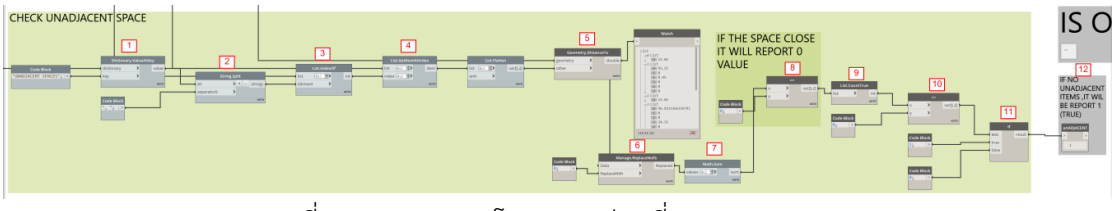
Node ที่เกี่ยวข้อง

-Watch ใช้ในการเรียกดูผลลัพธ์ที่ทำการประมวลผลมาได้ในแต่ละส่วน

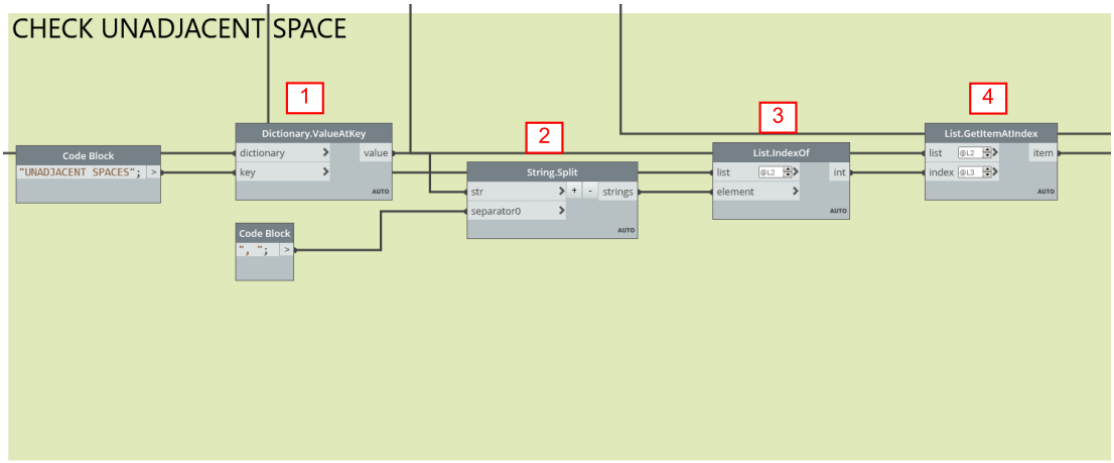


ภาพที่ 145 ภาพแสดงโปรแกรมส่วนที่ 1 Adjacent หมายเลข 8

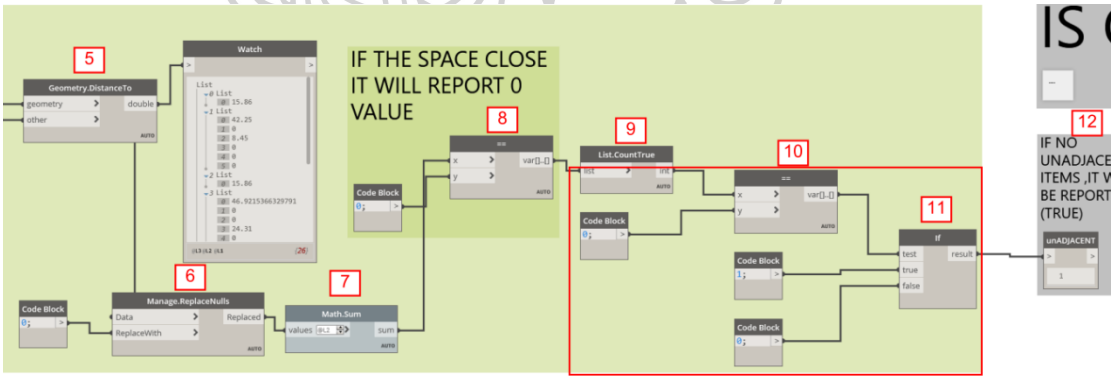
(2) Nonadjacent



ภาพที่ 146 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent



ภาพที่ 147 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent (ขยาย)



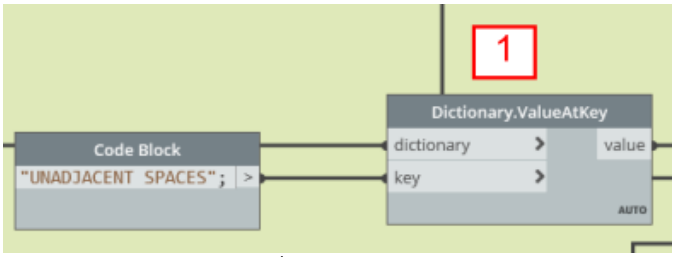
ภาพที่ 148 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent (ขยาย)

หมายเลข 1

ดึงข้อมูลค่าความต้องการของการวางผังคลังสินค้าในส่วนของความต้องการที่จะให้พื้นที่ไหนห้ามใกล้หรือติดกัน (Nonadjacent)

<u>Node ที่เกี่ยวข้อง</u>

-Dictionary.ValueAtKey ใช้ในการดึงข้อมูลจากหัวข้อของหมวดหมู่ที่สร้างไว้ ในขั้นตอน หมายเลข7 ของข้อ 3.2.2.2



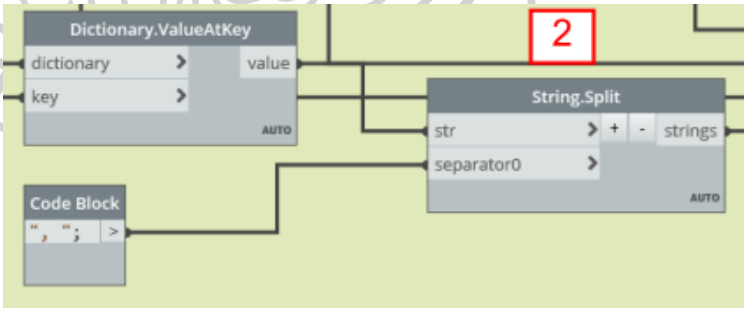
ภาพที่ 149 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 1

หมายเลข 2

เนื่องจากข้อมูลที่ดึงมานั้นมีลักษณะข้อมูลที่ไม่จำกัดปริมาณการกรอกของข้อมูลในหัวข้อนั้นๆซึ่งแต่ละลำดับของข้อมูลนั่นจะถูกไล่เรียงลงไปตาม Column แต่ข้อมูลที่กรอกในแต่ละ Column จะมีมากกว่า 1 ค่าโดยแต่ละค่าจะถูกคั่นด้วยสัญลักษณ์ คอมม่า (,) เช่น Coldroom, Chillroom, Staging เป็นต้น อ้างอิงจากหัวข้อ 3.2.1.3 ประเภทที่ 3 จึงต้องแยกค่าข้อมูลที่กรอกออกจากกันด้วยสัญลักษณ์ที่คั่นอยู่

Node ที่เกี่ยวข้อง

-String.Split ใช้แบ่งข้อมูลเดี่ยวข้อมูลเดียวที่ได้มาให้กลายเป็นข้อมูลเดี่ยวหลายๆข้อมูล



ภาพที่ 150 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 2

หมายเลข 3 - 4

ค่าที่ได้มาจากหมายเลข 2 จะเพียงชื่อของห้องหรือพื้นที่ต่างๆที่ต้องการให้ติดกับพื้นที่ที่กำหนด ดังนั่นจึงต้องนำชื่อที่ได้มาคืนค่าให้ตรงกับข้อมูลพื้นที่หลังจากการวางผังเรียบร้อยแล้วในขั้นตอน หมายเลข7 ของข้อ 3.2.2.2

Node ที่เกี่ยวข้อง

-List.IndexOf ใช้ระบุตำแหน่งของข้อมูลที่อยู่ใน List

-List.GetItemAtIndex ใช้ดึงข้อมูลโดยการระบุตำแหน่งที่อยู่ใน List

ภาพที่ 151 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 3

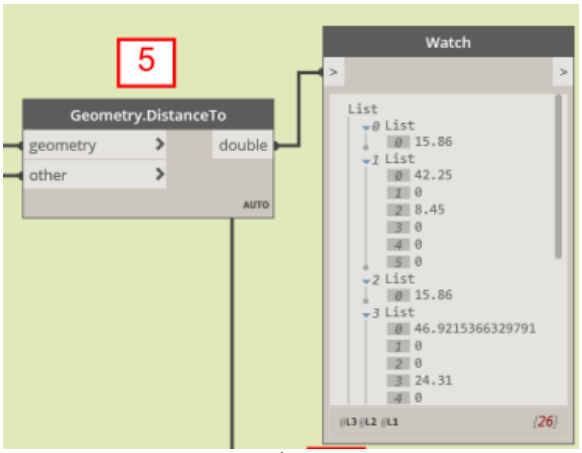
ภาพที่ 152 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 4

หมายเลข 5

โปรแกรมส่วนนี้จะทำการวัดระยะระหว่างพื้นที่ที่ห้ามใกล้กันจากผลลัพธ์การวางผังที่เกิดขึ้น

Node ที่เกี่ยวข้อง

-Geometry.DistanceTo ใช้ในการวัดระยะทางจากพื้นที่หนึ่งไปยังอีกพื้นที่หนึ่ง



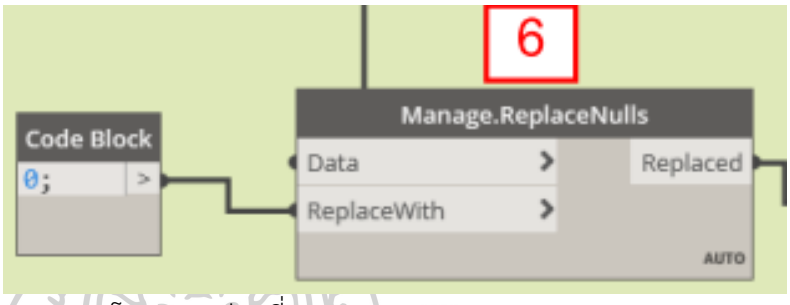
ภาพที่ 153 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 5

หมายเลข 6

ในกรณีที่ข้อมูลเป็นค่า Null คือไม่มีค่าของข้อมูล ถูกดึงมาจำเป็นต้องใส่ค่า 0 เพื่อป้องกันการทำงานผิดปกติของโปรแกรม

Node ที่เกี่ยวข้อง

-Manage.ReplaceNulls ใช้ในการกำหนดค่าแทนค่า Null



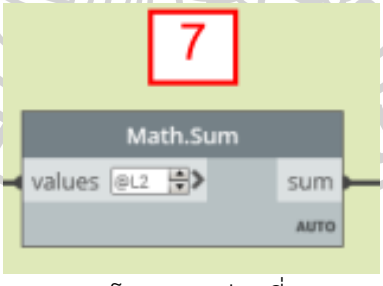
ภาพที่ 154 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 6

หมายเลข 7

บวกค่าระยะห่างทั้งหมดของพื้นที่ที่กำหนดเพื่อเป็นค่าของผลลัพธ์นั่นๆ

Node ที่เกี่ยวข้อง

-Math.Sum ใช้ในการรวมค่าของข้อมูล



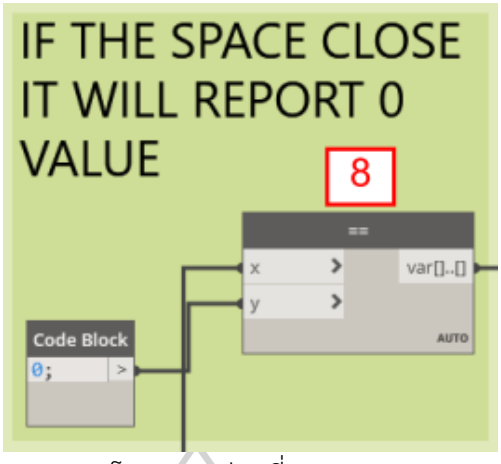
ภาพที่ 155 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 7

หมายเลข 8

โปรแกรมส่วนนี้เป็นการตรวจสอบว่ามีพื้นที่ใดที่ห้ามติดกัน มีการติดกันหรือไม่โดยผลลัพธ์จะออกมาเป็น True และ False หากผลลัพธ์จากหมายเลข 7 เป็น 0 จะได้ผล True คือมีพื้นที่ที่ผิดเงื่อนไข

Node ที่เกี่ยวข้อง

-== ใช้ในการตรวจสอบการเท่ากันของข้อมูลสองข้อมูลที่กำหนด



ภาพที่ 156 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 8

หมายเลข 9 - 10

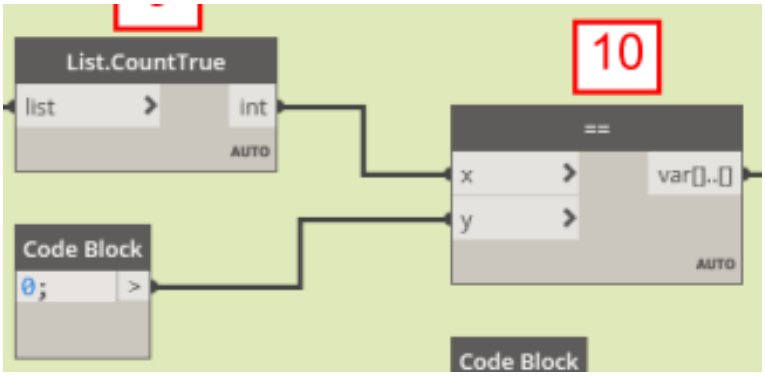
ตรวจสอบพื้นที่ทั้งหมดที่กำหนดว่ามี ค่า True จากหมายเลข 8 หรือไม่ ซึ่งตามเงื่อนไขไม่ควรมีค่า True

<u>Node ที่เกี่ยวข้อง</u>

-List.CountTrue ใช้นับจำนวนค่าผลลัพธ์ที่เป็น True

-== ใช้ในการตรวจสอบการเท่ากันของข้อมูลสองข้อมูลที่กำหนด

ภาพที่ 157 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 9



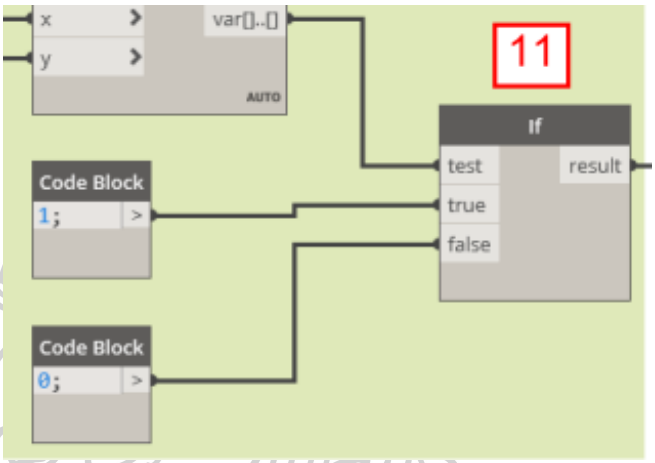
ภาพที่ 158 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 10

หมายเลข 11

จากหมายเลข 9 - 10 หากค่าที่ตรวจสอบได้เท่ากับ 0 จะให้ผลลัพธ์ของเงื่อนไขเป็นค่า 1 เท่ากับว่าไม่มีพื้นที่ผิดเงื่อนไข หากค่าที่ตรวจสอบได้มากกว่า 0 จะให้ผลลัพธ์ของเงื่อนไขเป็นค่า 0 คือผิดเงื่อนไขมีพื้นที่ต้องห้ามอยู่ติดกัน

Node ที่เกี่ยวข้อง

โดยมี Input ที่ใช้คือ



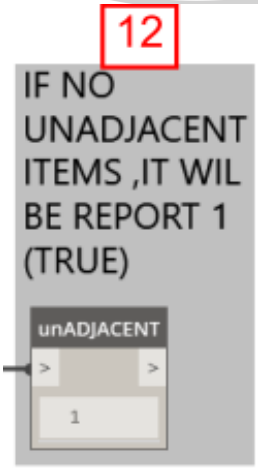
ภาพที่ 159 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 11

หมายเลข12

แสดงค่าของผลลัพธ์ที่ทำได้ตามเงื่อนไข การไม่ควรใกล้กันของพื้นที่ (NONADJACENT) ซึ่งในส่วนนี้จะถูกกำหนดเป็น Output ใน Refinery ต่อไป

Nodeที่เกี่ยวข้อง

-Watch ใช้ในการเรียกดูผลลัพธ์ที่ทำการประมวลผลมาได้ในแต่ละส่วน



ภาพที่ 160 ภาพแสดงโปรแกรมส่วนที่ 2 Nonadjacent หมายเลข 12

3.2.2.7 บันทึกค่า Input ของผลลัพธ์ที่ผู้ใช้งานพึงพอใจ (Export Favorite Input)

โปรแกรมวางผังคลังสินค้ามีวิธีการใช้งานที่ให้ผู้ใช้สามารถกำหนดค่า input ต่างๆได้อ้างอิงตามข้อ 3.1.2.3 ข้อ (1) Input ของโปรแกรมวางผังคลังสินค้า

-ค่า Probability

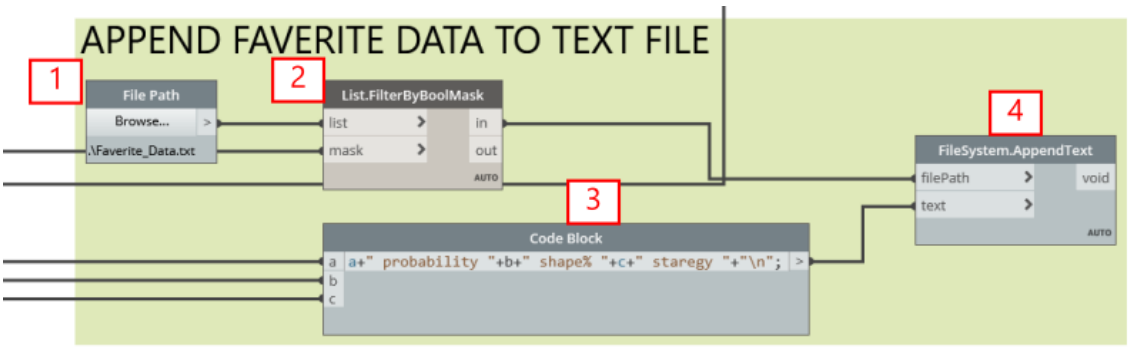
-ค่าเปอร์เซ็นความกว้าง Column

-ค่า Strategy

ซึ่งการกำหนดค่าที่แตกต่างกันในแต่ละหลังจะส่งผลให้ผลลัพธ์ที่เกิดขึ้นแตกต่างกัน โดยการกำหนดค่าในแต่ละครั้งไม่ได้ถูกบันทึกไว้ จึงมีการเพิ่มเติมโปรแกรมในส่วนของการบันทึกค่า Input ที่ผู้ใช้งานอยากบันทึกไว้ขึ้น ทำให้ในกรณีที่ค่าinput ทำให้เกิดผลลัพธ์ที่เราพึงพอใจจะสามารถเลือกเพื่อให้บันทึกค่า Input ทั้งหมดลงในไฟล์ Text แยกได้

โดยแนวคิดในการเขียนโปรแกรมในส่วนนี้สามารถเขียนเพื่อให้ตัว ค่า Input เหล่านี้ส่งออกไปบันทึกกลับใน Excel ไฟล์เดียวกันกับที่บรรจุข้อมูลดิบที่เป็น User Requirement ได้ แต่จากการทดลองพบว่ามีโอกาสทำให้ไฟล์ข้อมูลเสียหายเนื่องจากการบันทึกข้อมูลกลับในปริมาณมาก โดยเป็นผลมาจาก Node ที่เลือกใช้นั้นมีการเปิดไฟล์ Excel ทุกครั้งที่โปรแกรมมีการประมวลผลทำให้เมื่อมีการประมวลผลเพื่อให้เกิดผลลัพธ์ที่หลากหลายใน Refinery ทำให้เกิดความล้าช้าจนทำให้ไฟล์ Excel เสียหายได้

ดังนั่นทางผู้วิจัยจึงเลือกใช้การส่งออกค่า Input ดังกล่าวออกเป็น Text File ในรูปแบบของนามสกุลไฟล์ .txt แยกออกจากไฟล์ Excel รายละเอียดในส่วนExport Favorite Inputมีดังนี้



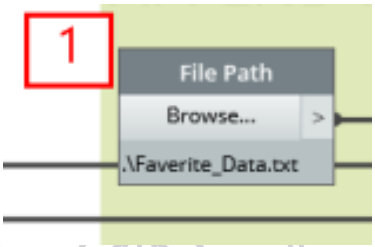
ภาพที่ 161 ภาพแสดงโปรแกรมวางผังคลังสินค้าใน Dynamo ส่วน Export Favorite Input

หมายเลข 1

ทำการเลือกที่อยู่ของไฟล์ที่ต้องการส่งข้อมูลออกไปจัดเก็บ

Node ที่เกี่ยวข้อง

-File Path ใช้ในการเลือกที่อยู่ของไฟล์



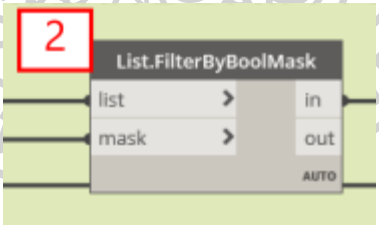
ภาพที่ 162 ภาพแสดงโปรแกรมส่วน Export Favorite Input หมายเลข 1

หมายเลข 2

โปรแกรมส่วนนี้จะเป็นส่วนที่ทำให้สามารถควบคุมการส่งหรือไม่ส่งข้อมูลออกไปบันทึกในไฟล์ที่เลือกได้

Node ที่เกี่ยวข้อง

-List.FilterByBoolMask ใช้ในการเลือกกรองข้อมูลที่จะส่งออกไปด้วยการเลือก True หรือ False



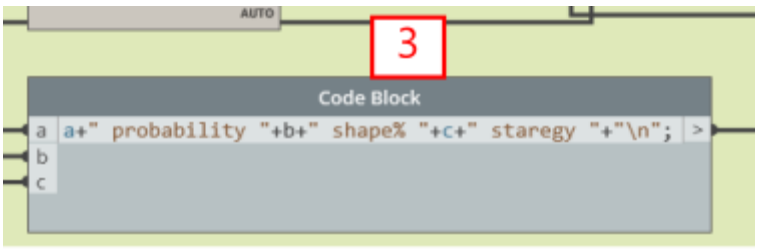
ภาพที่ 163 ภาพแสดงโปรแกรมส่วน Export Favorite Input หมายเลข 2

หมายเลข 3

รวมข้อมูล Input ทั้งหมด เข้าเป็นข้อมูลชุดเดียว

Node ที่เกี่ยวข้อง

-Code Block ใช้ในการเขียนคำสั่งเพื่อรวมข้อมูล Input ทั้งหมดเป็นชุดเดียวกัน



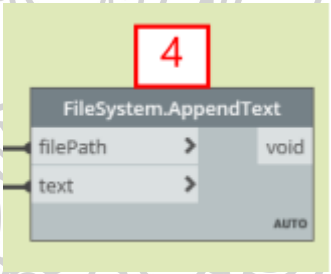
ภาพที่ 164 ภาพแสดงโปรแกรมส่วน Export Favorite Input หมายเลข 3

หมายเลข 4

บันทึกข้อมูล Input ที่ตรงตามความต้องการลงในไฟล์ที่ระบุโดยการต่อท้ายข้อมูลเดิมโดยไม่ลบหรือบันทึกทับข้อมูลก่อนหน้า

Nodeที่เกี่ยวข้อง

FileSystem.AppendText ใช้ในการบันทึกข้อความลงในไฟล์ที่ระบุโดยการต่อท้ายข้อมูลเดิมที่มีอยู่แล้ว



ภาพที่ 165 ภาพแสดงโปรแกรมส่วน Export Favorite Input หมายเลข 4

3.2.3 โปรแกรมวางผังคลังสินค้าส่วน Refinery

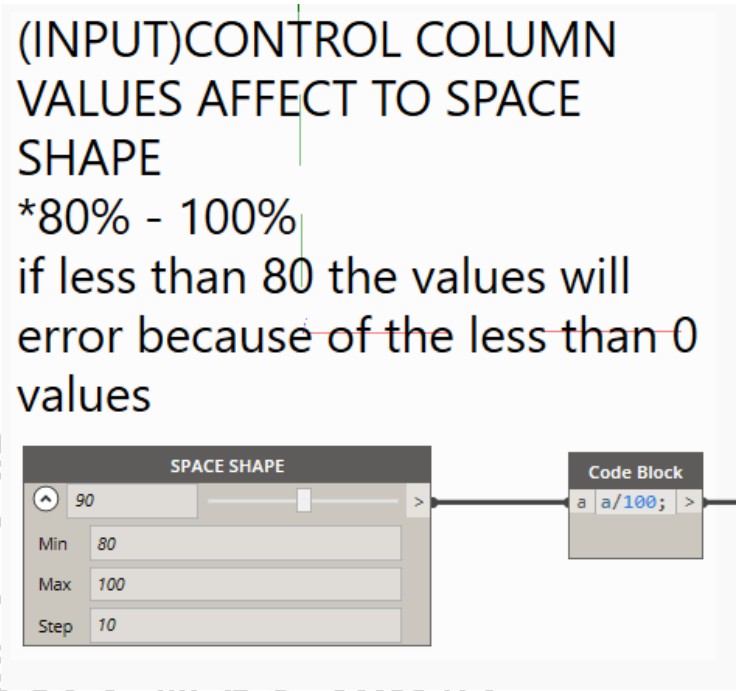
การส่งข้อมูลไปเข้าสู่กระบวนการที่เรียกว่า Generative Design ต่อในโปรแกรม Refinery นั่น จำเป็นต้องมีการกำหนดข้อมูลก่อนส่ง เป็น 2 ส่วนหลักด้วยกัน นั่นคือ ข้อมูลเข้า (IS INPUT) และ ข้อมูลออก (IS OUTPUT)

3.2.3.1 ข้อมูลเข้า (IS INPUT) ในโปรแกรมวางผังคลังสินค้า

โดยหลักการของข้อมูลที่จะเป็น IS INPUT นั่นจะเป็นข้อมูลที่มีผลต่อการเปลี่ยนแปลงผลลัพธ์ของการวางผังคลังสินค้า ในตัวโปรแกรมมีดังนี้

(1) Input 1 ช่วงเปอร์เซ็นความกว้าง Column (100%-50%)

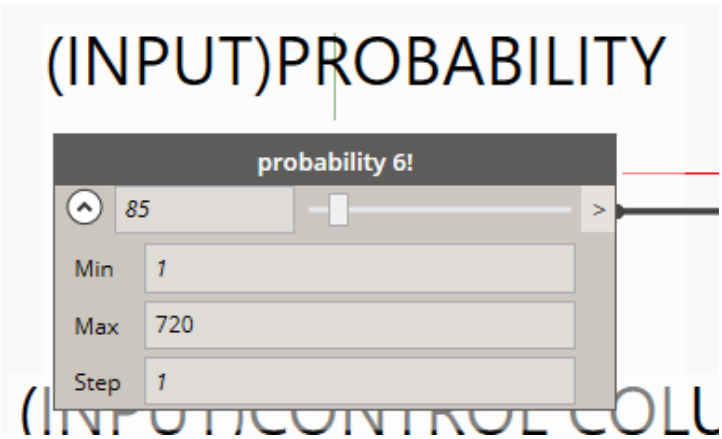
ใช้ในการปรับแต่งและกำหนดรูปทรงของพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) ให้แตกต่างกันในแต่ละผลลัพธ์การวางผังคลังสินค้า



ภาพที่ 166 ภาพแสดงโปรแกรมส่วน Input 1 ช่วงเปอร์เซ็นความกว้าง Column

(2) Input 2 ช่วง Probability (วิธีที่ 1- วิธีที่ n!)

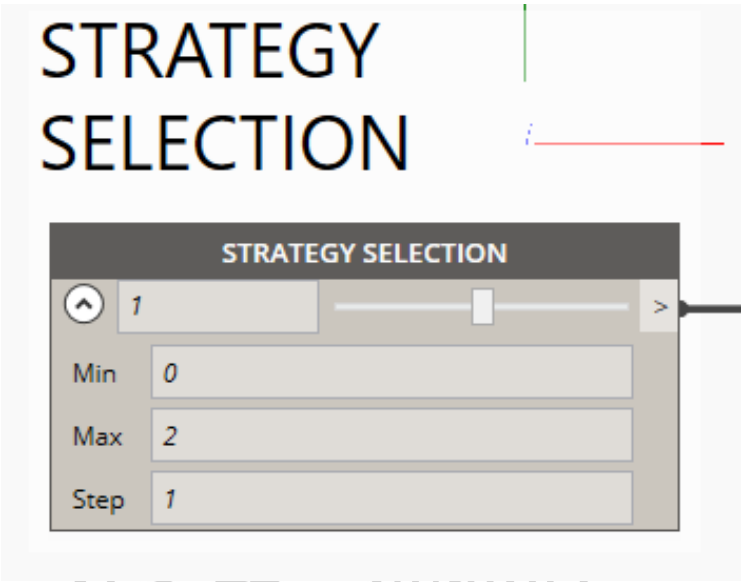
ใช้ในการปรับแต่งและกำหนดวิธีการของลำดับของการจัดเรียง (Permutation)ให้แตกต่างกันในแต่ละผลลัพธ์การวางผังคลังสินค้า



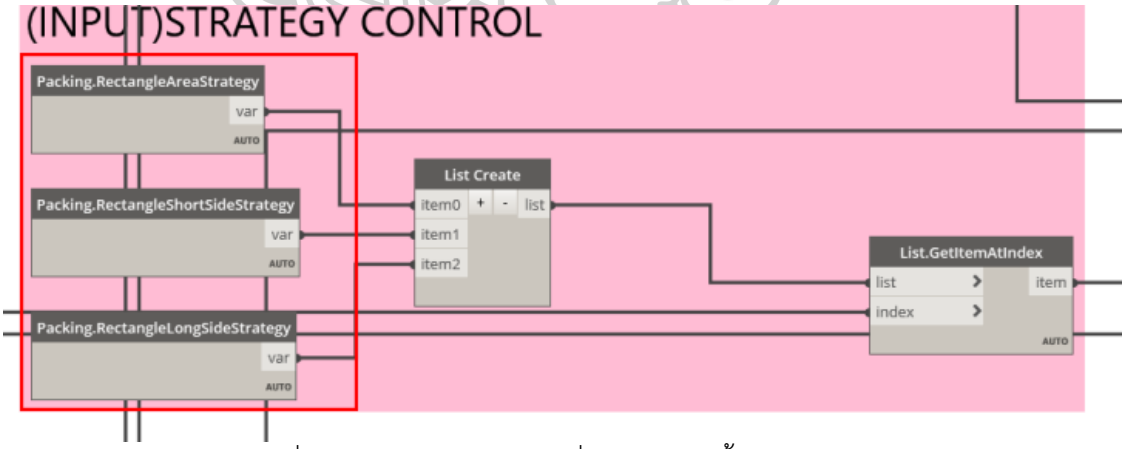
ภาพที่ 167 ภาพแสดงโปรแกรมส่วน Input2 ช่วงprobability

(3) Input 3 ช่วง Strategy (รูปแบบที่1 - รูปแบบที่3)

ใช้ในการปรับแต่งและกำหนดกระบวนการจัดเรียงพื้นที่ (Strategy) 3 รูปแบบ คือ รูปแบบที่ 1 (Short Side Strategy), รูปแบบที่ 2 (Long Side Strategy), รูปแบบที่ 3 (Area Strategy) ให้แตกต่างกันในแต่ละผลลัพธ์การวางผังคลังสินค้า



ภาพที่ 168 ภาพแสดงโปรแกรมส่วน Input 3 ช่วง Strategy



ภาพที่ 169 ภาพแสดง Node ที่เป็น Input ทั้ง 3 Strategy

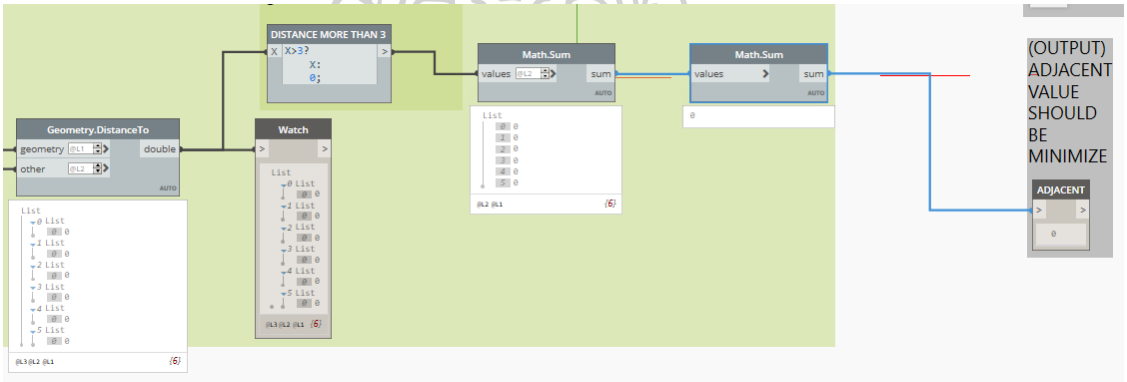
3.2.3.2 ข้อมูลออก (IS OUTPUT) ในโปรแกรมวางผังคลังสินค้า

โดยหลักการของข้อมูลที่จะเป็น IS OUTPUT นั่นจะเป็นข้อมูลที่มีที่เป็นผลลัพธ์ที่เกิดจากเงื่อนไขที่ได้ออกแบบโปรแกรมไว้ โดย IS OUTPUT ให้ข้อมูลที่จะถูกใช้ในการกำหนดเป้าหมายต่อไปในโปรแกรม Refinery

(1) Output 1 ค่าผลลัพธ์การใกล้กันของพื้นที่ (Adjacent)

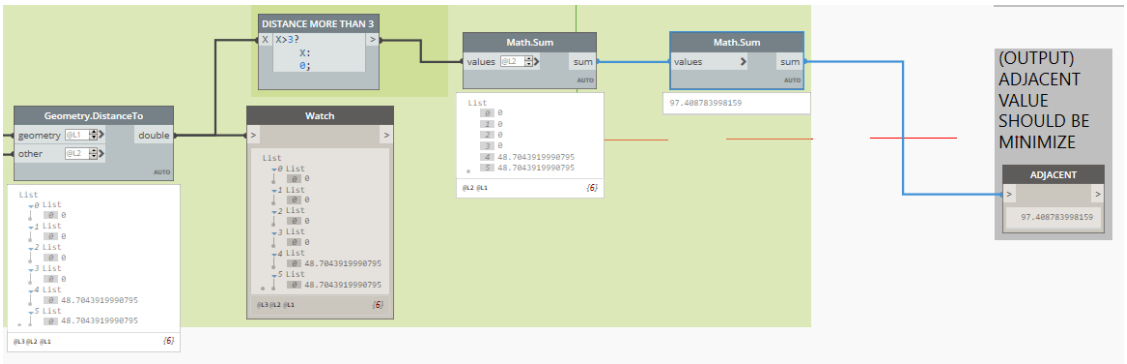
ค่าผลลัพธ์ที่เป็น Output จะออกไปในรูปแบบของผลรวมของระยะห่างระหว่างพื้นที่ที่กำหนดให้มีความใกล้กัน (Adjacent) ดังนั่นเลขที่เกิดขึ้นจะเป็นได้ตั้งแต่ 0 ขึ้นไป ดังนั่นผลลัพธ์จะมีหลากหลายค่า n โดยผลลัพธ์ที่เกิดขึ้นจะมี 2 กลุ่มคือ

-กลุ่มที่ 1 ผลลัพธ์ที่เป็นไปตามเงื่อนไขที่กำหนดใน Excel ตั้งต้น คือพื้นที่ที่ต้องการให้ใกล้กันนั้น ทุกพื้นที่ได้ใกล้กันหมด ซึ่งค่าจะออกมาเป็น 0



ภาพที่ 170 ภาพแสดงผลลัพธ์การใกล้กันของพื้นที่กลุ่ม 1

-กลุ่มที่ 2 คือกลุ่มที่ผลลัพธ์มากกว่า 0 แสดงว่ามีพื้นที่ที่ควรใกล้กันตามเงื่อนไขที่กำหนดใน excel นั่นไม่ได้ใกล้กันตามที่กำหนด ซึ่งยิ่งค่าของผลลัพธ์มีค่ามากเท่าไหร่ พื้นที่ที่ควรใกล้กันนั่นยิ่งน้อยลงไป



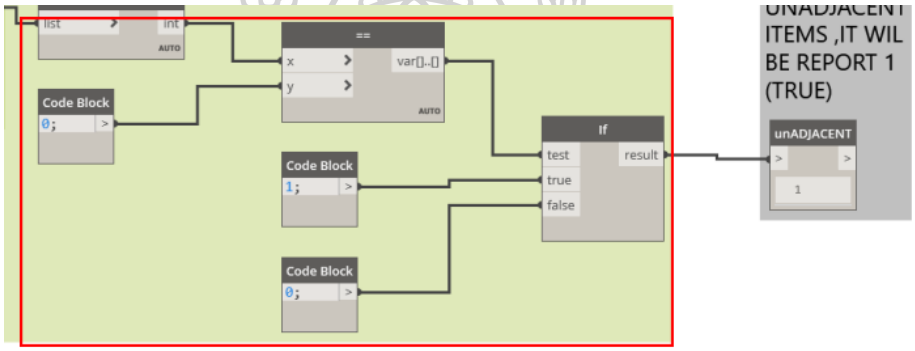
ภาพที่ 171 ภาพแสดงผลลัพธ์การใกล้กันของพื้นที่กลุ่ม 2

(2) Output2 ค่าผลลัพธ์การไม่ควรใกล้กันของพื้นที่ (Nonadjacent)

ค่าผลลัพธ์ที่เป็น Output จะออกไปในรูปแบบของของ True / False เพียง 2 ค่า ซึ่งจะแตกต่างกับผลลัพธ์ของเงื่อนไข Adjacent เนื่องจากพื้นที่ที่ไม่ควรติดกันนั่น ในการวางผังคลังสินค้า จะไม่ควรเกิดขึ้นได้เลย ดังนั้นหากมีพื้นที่ที่ไม่ควรใกล้กันนั้น มีอย่างน้อย 1 พื้นที่ได้ใกล้กัน ค่าจะออกมาเป็น False

โดยผลลัพธ์จะเป็นไปตามเงื่อนไขของพื้นที่ที่ไม่ควรใกล้กัน ซึ่งถูกระบุไว้ใน Excel ตั้งต้น หากโปรแกรมแสดงผลลัพธ์มีพื้นที่ที่ไม่ควรใกล้กันเป็นไปตามเงื่อนไขค่าจะขึ้น True

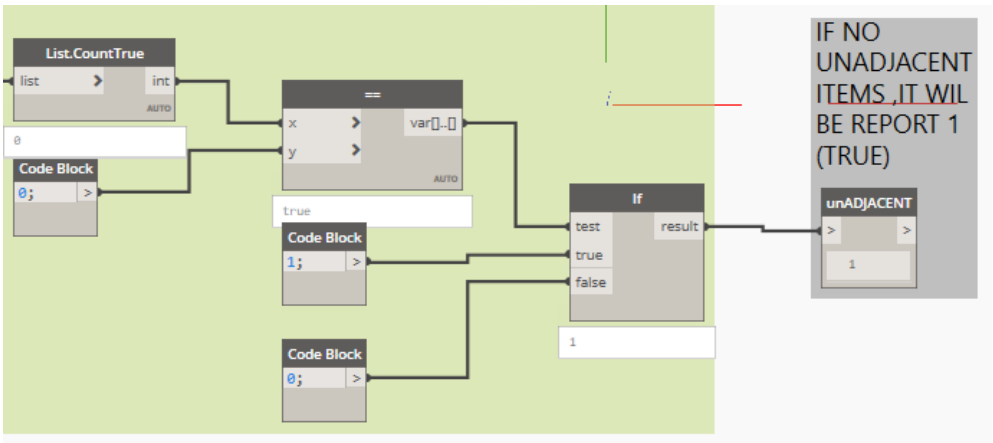
หลักการในการนำค่าไปใช้ในโปรแกรม Refinery นั่นจะไปในลักษณ์ของผลลัพธ์เชิงปริมาณ ทำให้ต้องแปลงผลลัพธ์ True = 1 และ False = 0 เพื่อจะส่งไปทำการประมวลผลต่อใน Refinery



ภาพที่ 172 ภาพแสดงโปรแกรมส่วน Output2 ค่าผลลัพธ์การไม่ควรใกล้กันของพื้นที่

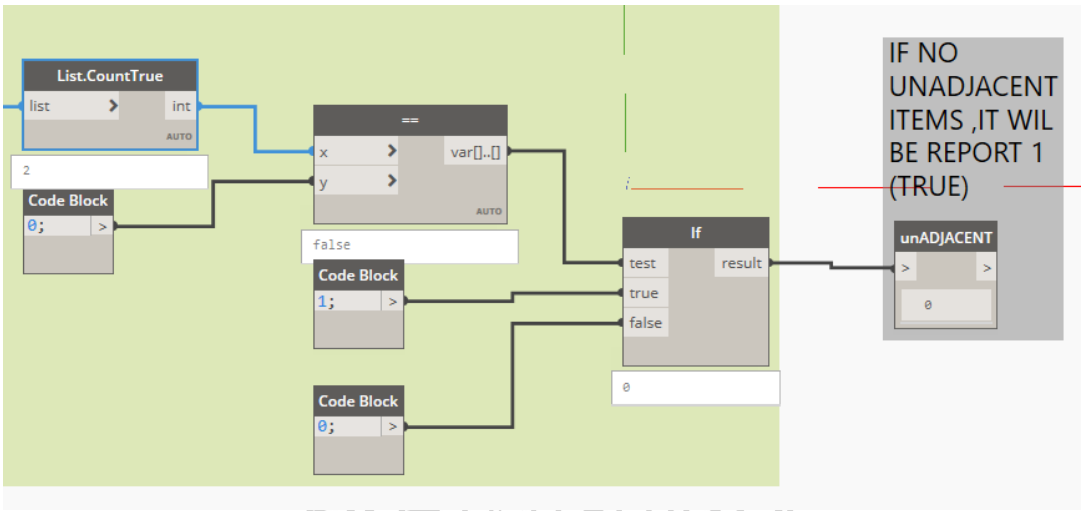
โดยผลลัพธ์ที่เกิดขึ้นจะมี2กลุ่มคือ

-กลุ่มที่ 1 ผลลัพธ์ที่เป็นไปตามเงื่อนไขที่กำหนดใน Excel ตั้งต้น คือพื้นที่ที่ไม่ควรใกล้กันนั้น ทุกพื้นที่ได้ไกลกันหมด ซึ่งค่าจะออกมาเป็น 1 = True



ภาพที่ 173 ภาพแสดงผลลัพธ์การไม่ควรใกล้กันของพื้นที่กลุ่ม 1

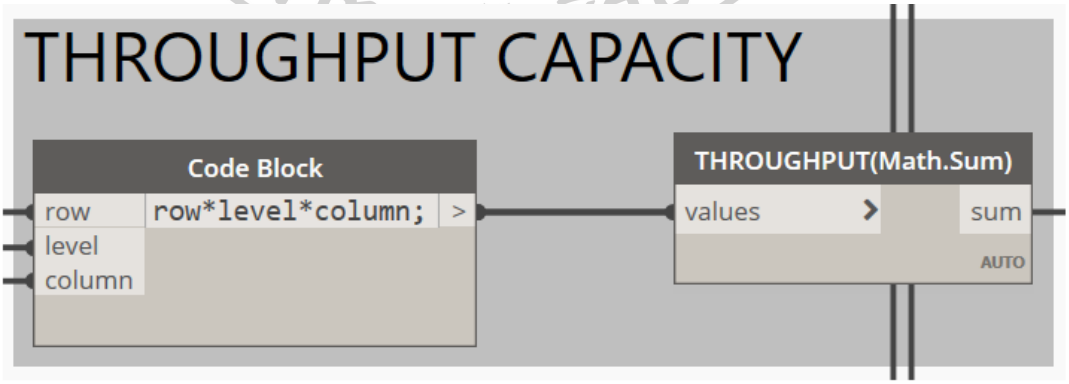
-กลุ่มที่ 2 ผลลัพธ์ที่ไม่เป็นไปตามเงื่อนไขที่กำหนดใน Excel ตั้งต้น คือ พื้นที่ที่ไม่ควรใกล้กันนั้น มีอย่างน้อย 1 พื้นที่ได้ใกล้กัน ซึ่งค่าจะออกมาเป็น 0 = False



ภาพที่ 174 ภาพแสดงผลลัพธ์การไม่ควรใกล้กันของพื้นที่กลุ่ม 2

(3) Output 3 ค่าผลลัพธ์ปริมาณการจัดเก็บ (Capacity)

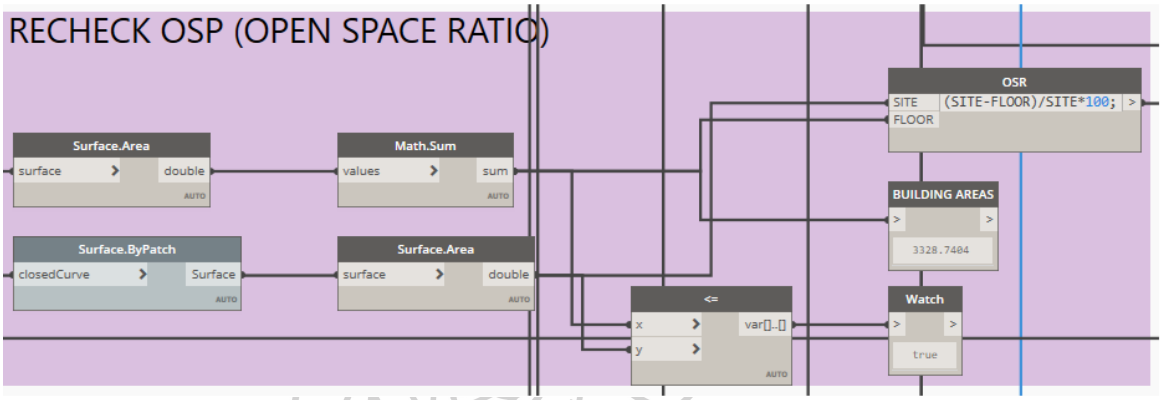
ปริมาณการจัดเก็บของแต่ละพื้นที่หลังจากวางผังคลังสินค้าสำเร็จจะถูกคำนวนและแสดงผลลัพธ์ออกมาโดยโปรแกรมโดยในส่วน Refinery ผู้ใช้งานสามารถกำหนดเป้าหมายของผลลัพธ์ที่จะเกิดขึ้นได้ อย่างเช่นต้องการตัวเลือกของผังคลังสินค้าที่มีปริมาณการจัดเก็บที่เยอะที่สุดจากผังคลังสินค้ารูปแบบอื่นๆ



ภาพที่ 175 ภาพแสดงโปรแกรมส่วน Output 3 ค่าผลลัพธ์ปริมาณการจัดเก็บ

(4) Output 4 ค่าผลลัพธ์ขนาดอาคาร (Building Area)

ขนาดพื้นที่ของอาคารทั้งหมดหลังจากวางผังคลังสินค้าสำเร็จ จะแตกต่างกันในแต่ละผลลัพธ์ของการวางผัง โดยโปรแกรมโดยในส่วน Refinery ผู้ใช้งานสามารถกำหนดเป้าหมายของผลลัพธ์ที่จะเกิดขึ้นได้ อย่างเช่นต้องการตัวเลือกของผังคลังสินค้าที่มีขนาดอาคารที่เล็กที่สุดจากผังคลังสินค้ารูปแบบอื่นๆเนื่องจากขนาดคลังสินค้าที่เล็กกว่าจะส่งผลต่อราคาค่าก่อสร้างที่น้อยกว่าตามไปด้วย

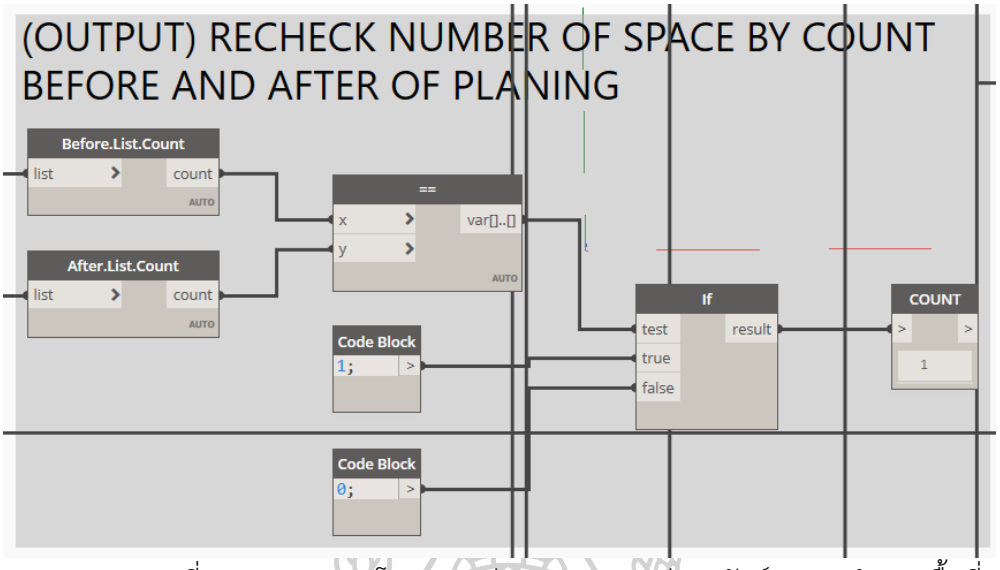


ภาพที่ 176 ภาพแสดงโปรแกรมส่วน Output4 ค่าผลลัพธ์ขนาดอาคาร

(5) Output 5 ค่าผลลัพธ์ผลรวมจำนวนพื้นที่ (COUNT SPACE)

การทำงานของโปรแกรมในส่วนของการจัดวางผังนั่นมีโอกาสที่พื้นที่จัดเก็บสินค้าที่มีทั้งหมดจะถูกจัดเรียงลงไปในพื้นที่ของเขตอาคารไม่ครบ ในกรณีที่พื้นที่ขอบเขตอาคารมีขนาดเล็กกว่าพื้นที่รวมของพื้นที่จัดเก็บสินค้าทั้งหมด หรือ รูปทรงของพื้นที่จัดเก็บสินค้าในแต่ละพื้นที่ไม่สามารถถูกจัดวางลงไปในพื้นที่ขอบเขตอาคารได้หมด ซึ่งจะทำให้เกิดผลลัพธ์ของการจัดวางผังอาคารที่ผิดพลาดได้ คือได้พื้นที่เก็บสินค้าไม่ครบ พื้นที่อาคารที่ได้ ไม่สามารถตอบสนองปริมาณการเก็บ (Capacity) ที่ลูกค้าต้องการได้

ดังนั่นจึงต้องมีการตรวจเช็คว่าจำนวนพื้นที่จัดเก็บสินค้า ก่อนที่จะถูกจัดเรียงลงไปในพื้นที่ขอบเขตอาคารนั่น มีจำนวนเท่ากันกับ หลังจากที่ถูกจัดเรียงลงไปในพื้นที่ขอบเขตอาคารแล้ว



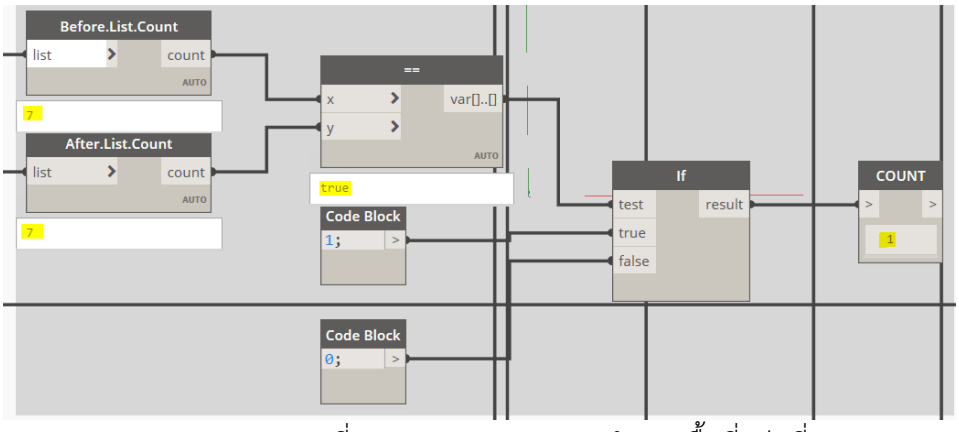
ภาพที่ 177 ภาพแสดงโปรแกรมส่วน Output5 ค่าผลลัพธ์ผลรวมจำนวนพื้นที่

โดยผลลัพธ์ที่ได้จะออกมาในรูปแบบ True / False ซึ่งจะมีเพียง 2 คำตอบ เพราะต้องการกำจัดผลลัพธ์ที่ จำนวนพื้นที่จัดเก็บสินค้าก่อนที่จะถูกจัดเรียงลงไปในพื้นที่ขอบเขตอาคารนั่น มีจำนวนไม่เท่ากับ หลังจากที่ถูกจัดเรียงลงไปในพื้นที่ขอบเขตอาคารแล้ว กรณีดังกล่าวไม่ควรเกิดขึ้น

หลักการในการนำค่าไปใช้ในโปรแกรม Refinery นั่นจะไปในลักษณ์ของผลลัพธ์เชิงปริมาณ ทำให้ต้องแปลงผลลัพธ์ True = 1 และ False = 0 เพื่อจะส่งไปทำการประมวลผลต่อใน Refinery

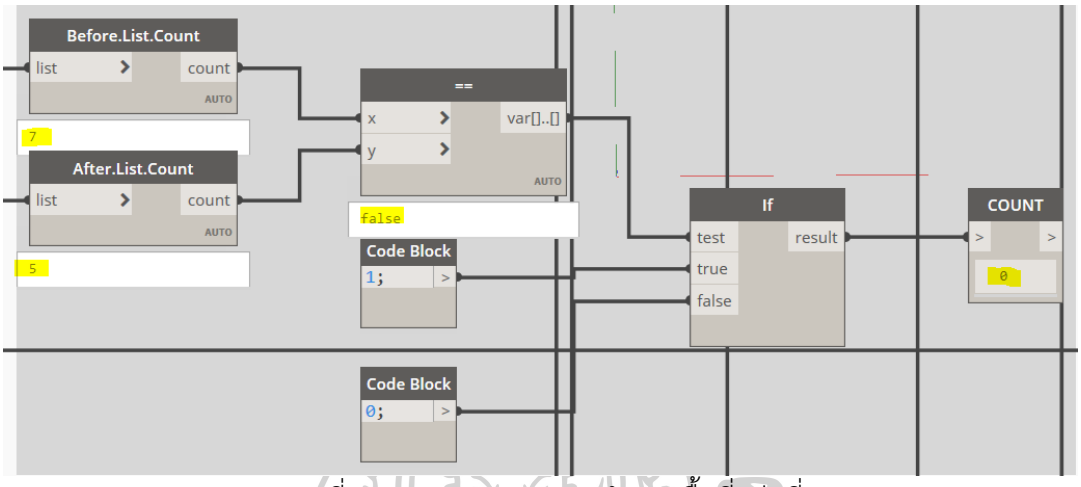
โดยผลลัพธ์ที่เกิดขึ้นจะมี 2 กลุ่มคือ

-กลุ่มที่ 1 ผลลัพธ์ที่ถูกต้อง คือ จำนวนพื้นที่จัดเก็บสินค้าก่อนที่จะถูกจัดเรียงลงไปในพื้นที่ขอบเขตอาคารนั่น มีจำนวนเท่ากับ หลังจากที่ถูกจัดเรียงลงไปในพื้นที่ขอบเขตอาคารแล้ว ซึ่งค่าจะออกมาเป็น 1 = True



ภาพที่ 178 ภาพแสดงผลรวมจำนวนพื้นที่กลุ่มที่ 1

-กลุ่มที่ 2 ผลลัพธ์ที่ไม่ถูกต้อง คือ จำนวนพื้นที่จัดเก็บสินค้าก่อนที่จะถูกจัดเรียงลงไปในพื้นที่ขอบเขตอาคารนั่น มีจำนวนไม่เท่ากับ หลังจากที่ถูกจัดเรียงลงไปในพื้นที่ขอบเขตอาคารแล้ว ซึ่งค่าจะออกมาเป็น 0 = False



ภาพที่ 179 ภาพแสดงผลรวมจำนวนพื้นที่กลุ่มที่ 2

มหาวิทยาลัยศิลปากร

บทที่ 4
ผลการวิจัยและอภิปรายผล

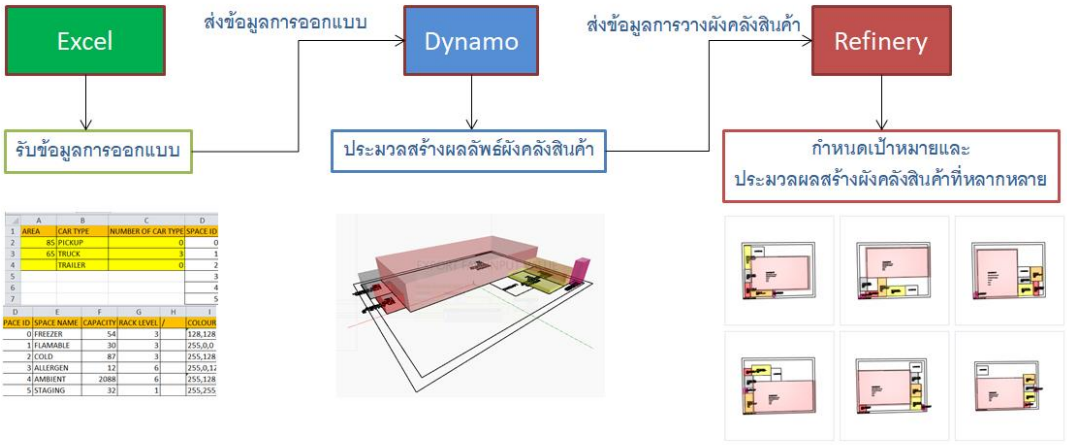
4.1 โปรแกรมวางผังคลังสินค้าและวิธีการใช้งาน

ลักษณะหลักการทำงานของโปรแกรมวางผังคลังสินค้าแบ่งออกเป็น 3 ขั้นตอนหลักตามโปรแกรมที่เปลี่ยนไปในแต่ละขั้นตอน คือ

ขั้นตอนที่ 1 ใช้โปรแกรมวางผังคลังสินค้าส่วน Excel รับข้อมูลที่เกี่ยวกับออกแบบจากลูกค้าโดยการกรอกข้อมูลใส่ตาราง Excel ที่เตรียมไว้

ขั้นตอนที่ 2 ใช้โปรแกรมวางผังคลังสินค้าส่วน Dynamo เพื่อนำเข้าข้อมูลจากขั้นตอนที่ 1 มาประมวลผลเพื่อสร้างพื้นที่และผลลัพธ์การวางผังคลังสินค้าอย่างอัตโนมัติ

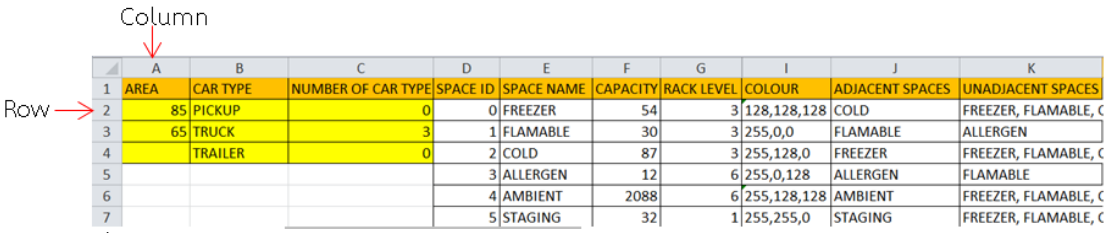
ขั้นตอนที่ 3 ใช้โปรแกรมวางผังคลังสินค้าส่วน Refinery ในการกำหนดเป้าหมายของผลลัพธ์การวางผังคลังสินค้า เพื่อให้ได้ผังคลังสินค้าที่หลากหลายและนำไปสู่การพิจารณาและเลือกผลลัพธ์สุดท้ายของผังคลังสินค้าที่เหมาะสมที่สุด



ภาพที่ 180 ภาพแสดงขั้นตอนการทำงานด้วยโปรแกรมวางผังคลังสินค้า

4.1.1 โปรแกรมวางผังคลังสินค้าส่วน Excel

ขั้นตอนแรกของหลักการทำงานของโปรแกรมวางผังคลังสินค้าจะเป็นการรับความต้องการของลูกค้าที่เกี่ยวกับการออกแบบคลังสินค้าโดยไฟล์ Excel ที่เตรียมไว้โดยจะถูกแบ่งหัวข้อในการกรอกเป็นแต่ละแนว (Column) และข้อมูลกลุ่มเดียวกันจะถูกกรอกไล่ลงมาแต่ละแถว (Row) ในแนว (Column)เดียวกัน



	A	B	C	D	E	F	G	I	J	K
1	AREA	CAR TYPE	NUMBER OF CAR TYPE	SPACE ID	SPACE NAME	CAPACITY	RACK LEVEL	COLOUR	ADJACENT SPACES	UNADJACENT SPACES
2	85	PICKUP	0	0	FREEZER	54	3	128,128,128	COLD	FREEZER, FLAMABLE, C
3	65	TRUCK	3	1	FLAMABLE	30	3	255,0,0	FLAMABLE	ALLERGEN
4		TRAILER	0	2	COLD	87	3	255,128,0	FREEZER	FREEZER, FLAMABLE, C
5				3	ALLERGEN	12	6	255,0,128	ALLERGEN	FLAMABLE
6				4	AMBIENT	2088	6	255,128,128	AMBIENT	FREEZER, FLAMABLE, C
7				5	STAGING	32	1	255,255,0	STAGING	FREEZER, FLAMABLE, C

ภาพที่ 181 ภาพแสดงตำแหน่ง Row และ Column ในโปรแกรมวางผังคลังสินค้าส่วน Excel

4.1.1.1 วิธีการใช้งานโปรแกรมวางผังคลังสินค้าส่วน Excel

ไฟล์ Excel ถูกสร้างขึ้นไว้สำหรับรับข้อมูลความต้องการของลูกค้าจะถูกแบ่งเป็นหัวข้อทั้งหมด 9หัวข้อ อ้างอิงจากบทที่3 ข้อ 3.1.1 ใช้Excel ในขั้นการรับข้อมูลความต้องการของลูกค้าที่มีผลต่อการออกแบบวางผังคลังสินค้า ในแต่ละแนว(Column)ดังนี้

(1) Area

ในหัวข้อแรกเป็นการกรอกข้อมูลในส่วนของขนาดที่ดินที่จะใช้ในการออกแบบอาคาร ซึ่งเป็นลักษณะการกรอกข้อมูล2ค่า คือ ความกว้าง และ ความยาว โดยจะกรอกไล่ลงมาที่ช่องแนว (Column) A แถว (Row) ที่ 2 และ 3 ไล่ลงมาตามลำดับ โดยข้อมูลที่กรอกมีหน่วยเป็น เมตร

	A
1	AREA
2	85
3	65

ภาพที่ 182 ภาพแสดงส่วนกรอกข้อมูล Area

(2) Car Type

ในหัวข้อชนิดรถที่จะใช้ในการออกแบบอาคาร โดยจะถูกระบุเป็นค่าตั้งต้นไว้3ชนิด คือ Pick up, Truck และ Trailer โดยมีคำอธิบายของรถแต่ละชนิดดังนี้

-รถ Pick up = รถกระ4ล้อ

-รถ Truck = รถ6ล้อ-10ล้อ

-รถ Trailer = รถขนส่ง 40 ฟุต

จำนวนรถแต่ละชนิดจะถูกกรอกใน ข้อถัดไป

	B
1	CAR TYPE
2	PICKUP
3	TRUCK
4	TRAILER

ภาพที่ 183 ภาพแสดงส่วนกรอกข้อมูล Car Type

(3) Number Of Car Type

ในหัวข้อนี้เป็นการกรอกจำนวนรถบรรทุกตามชนิดที่ระบุไว้ตามความต้องการของลูกค้า

	C
1	NUMBER OF CAR TYPE
2	0
3	3
4	0

ภาพที่ 184 ภาพแสดงส่วนกรอกข้อมูล Number Of Car Type

(4) Space ID

ในหัวข้อนี้เป็นเพียงลำดับของพื้นที่โดยเริ่มจาก 0

	D
1	SPACE ID
2	0
3	1
4	2
5	3
6	4
7	5

ภาพที่ 185 ภาพแสดงส่วนกรอกข้อมูล Space ID

(5) Space Name

ในหัวข้อนี้เป็นการกรอกชื่อพื้นที่จัดเก็บสินค้าที่ต้องการนำมาจัดวางผัง

	E
1	SPACE NAME
2	FREEZER
3	FLAMABLE
4	COLD
5	ALLERGEN
6	AMBIENT
7	STAGING

ภาพแสดงส่วนกรอกข้อมูล Space Name

(6) Capacity

ในหัวข้อนี้เป็นการกรอกจำนวนการจัดเก็บที่ต้องการในแต่ละพื้นที่โดยมีหน่วยเป็น Pallet

	F
1	CAPACITY
2	54
3	30
4	87
5	12
6	2088
7	32

ภาพแสดงส่วนกรอกข้อมูล Capacity

(7) Rack Level

ในหัวข้อนี้เป็นการกรอกจำนวนการซ้อนชั้นของวัตถุที่จะจัดเก็บ

	G
1	RACK LEVEL
2	3
3	3
4	3
5	6
6	6
7	1

ภาพแสดงส่วนกรอกข้อมูล Rack Level

(8) Colour

ในหัวข้อนี้เป็นการระบุสีเพื่อจำแนกแต่ละพื้นที่เป็นค่า RGB

	I
1	COLOUR
2	128,128,128
3	255,0,0
4	255,128,0
5	255,0,128
6	255,128,128
7	255,255,0

ภาพแสดงส่วนกรอกข้อมูล Colour

(9) Adjacent Spaces

ในหัวข้อนี้เป็นการใส่เงื่อนไขของการใกล้กันของพื้นที่โดยใส่ชื่อพื้นที่ที่ต้องการให้ใกล้กันใน Row เดียวกัน ชื่อต้องพื้นที่ต้องเหมือนกันกับในหัวข้อ Space Name

	J
1	ADJACENT SPACES
2	COLD
3	FLAMABLE
4	FREEZER
5	ALLERGEN
6	AMBIENT
7	STAGING

ภาพแสดงส่วนกรอกข้อมูล Adjacent Spaces

(9) Nonadjacent Spaces

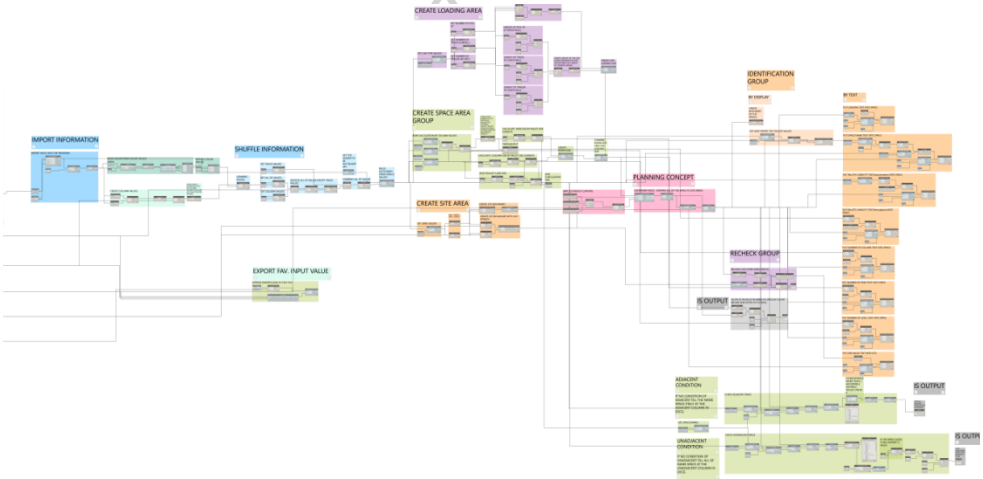
ในหัวข้อนี้เป็นการใส่เงื่อนไขของการห้ามใกล้กันของพื้นที่โดยใส่ชื่อพื้นที่ที่ต้องการห้ามใน Row เดียวกัน ชื่อต้องพื้นที่ต้องเหมือนกันกับในหัวข้อ Space Name

	K
1	UNADJACENT SPACES
2	FREEZER, FLAMABLE, C
3	ALLERGEN
4	FREEZER, FLAMABLE, C
5	FLAMABLE
6	FREEZER, FLAMABLE, C
7	FREEZER, FLAMABLE, C

ภาพแสดงส่วนกรอกข้อมูล Nonadjacent Spaces

4.1.2 โปรแกรมวางผังคลังสินค้าส่วน Dynamo

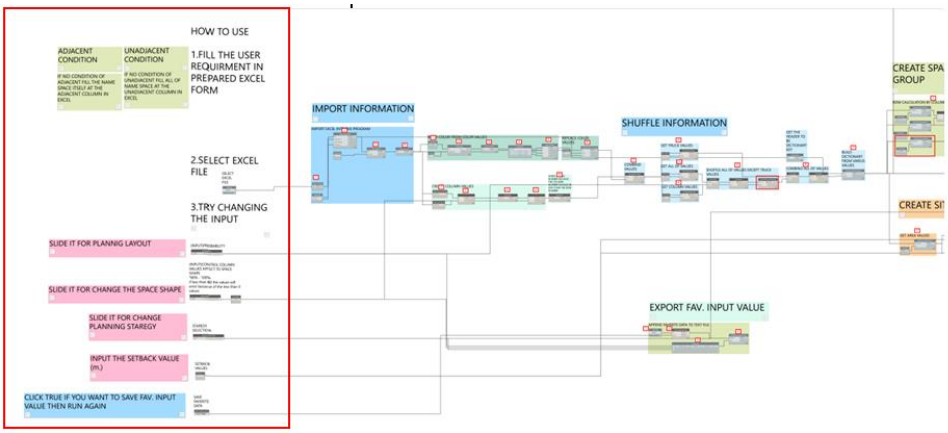
การใช้งานโปรแกรมวางผังคลังสินค้าส่วน Dynamo เมื่อเปิดโปรแกรม Dynamo บนโปรแกรม Revit แล้วเลือกเปิดไฟล์ WAREHOUSE PLANNING PROGRAM.DYN จะเป็นการเรียกโปรแกรมวางผังคลังสินค้าในส่วน Dynamo ขึ้นมา โดยจะหน้าตาโปรแกรมจะแสดง Nodeที่เชื่อมต่อกันทั้งหมด โดยผู้วิจัยได้ทำการจัดกลุ่ม Node ต่างๆโดยเขียนหัวข้อในแต่ละกลุ่ม พร้อมคำอธิบายการทำงานของโปรแกรมในแต่ละส่วนไว้พร้อมทั้งแยกเป็นสีเพื่อการจำแนกกลุ่มที่ง่ายยิ่งขึ้น โดยรายละอียดในแต่ละกลุ่มนั่นได้อธิบายไว้อย่างละเอียดในบทที่ 3 ข้อ 3.2.2 โปรแกรมวางผังคลังสินค้าส่วน Dynamo



ภาพที่ 186 หน้าตาโปรแกรมวางผังคลังสินค้าส่วน Dynamo

4.1.2.1 วิธีการใช้งานโปรแกรมวางผังคลังสินค้าส่วน Dynamo

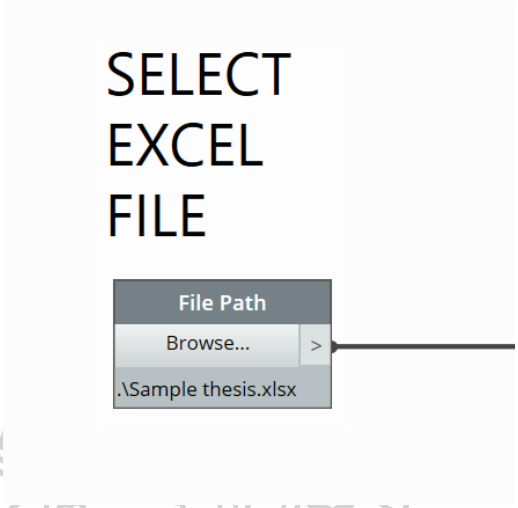
เพื่อให้ง่ายต่อผู้ที่จะนำโปรแกรมวางผังคลังสินค้าส่วน Dynamoไปใช้งาน ผู้วิจัยได้จัดการนำค่า Input ทั้งหมดที่มีผลต่อการเปลี่ยนแปลงผลลัพธ์ของการวางผังคลังสินค้าไว้ที่ซ้ายมือสุดของหน้าจอโปรแกรมพร้อมทั้งระบุคำอธิบายและขั้นตอนการใช้งานดังนี้



ภาพที่ 187 ส่วน Input ซ้ายสุดบนโปรแกรมวางผังคลังสินค้าส่วน Dynamo

(1) เลือก ไฟล์Excel ข้อมูลความต้องการของลูกค้า

เลือกไฟล์Excel ที่ถูกกรอกเรียบร้อยแล้วจากข้อ 4.1.1โปรแกรมวางผังคลังสินค้าส่วน Excel โดยการเลือกที่เครื่องหมาย Browse จากนั้นเลือกไฟล์ข้อมูลความต้องการของลูกค้า

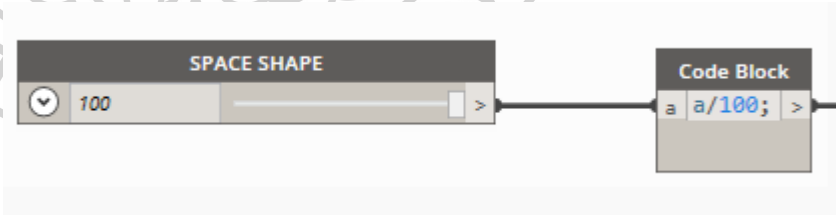


ภาพที่ 188 Node ที่ใช้เลือกไฟล์

(2) ทดลองใส่ค่า Input ที่ต้องการ

ค่า Input ทั้งหมดที่มีผลต่อการเปลี่ยนแปลงผลลัพธ์ของการวางผังคลังสินค้าจะถูกรวมไว้ที่ตำแหน่งนี้ โดยมีทั้งหมด 4ค่าInput

1) Input1 รูปทรงของพื้นที่ (100%-50%)



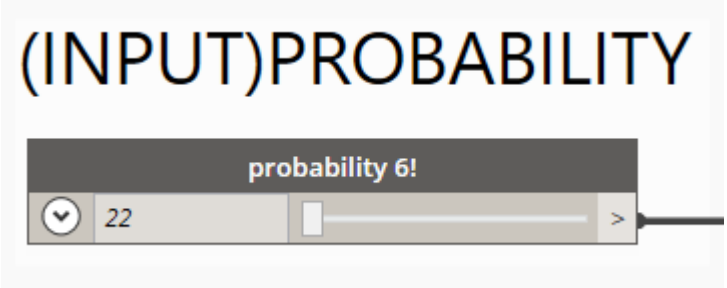
ภาพที่ 189 Node ที่ใช้ปรับค่า Input1 รูปทรงของพื้นที่

ในส่วนนี้เป็นการปรับค่าช่วงเปอร์เซ็นความกว้าง Column (100%-50%) ใช้ในการปรับแต่งและกำหนดรูปทรงของพื้นที่จัดเก็บสินค้าแต่ละชนิด (Room Space) ให้แตกต่างกันโดยจะส่งผลให้เกิดผลลัพธ์การวางผังคลังสินค้าที่แตกต่างและหลากหลาย ดังภาพด้านล่าง

ภาพที่ 190 ผลลัพธ์การวางผังคลังสินค้าที่หลากหลายจากการปรับค่าช่วงเปอร์เซ็นความกว้าง

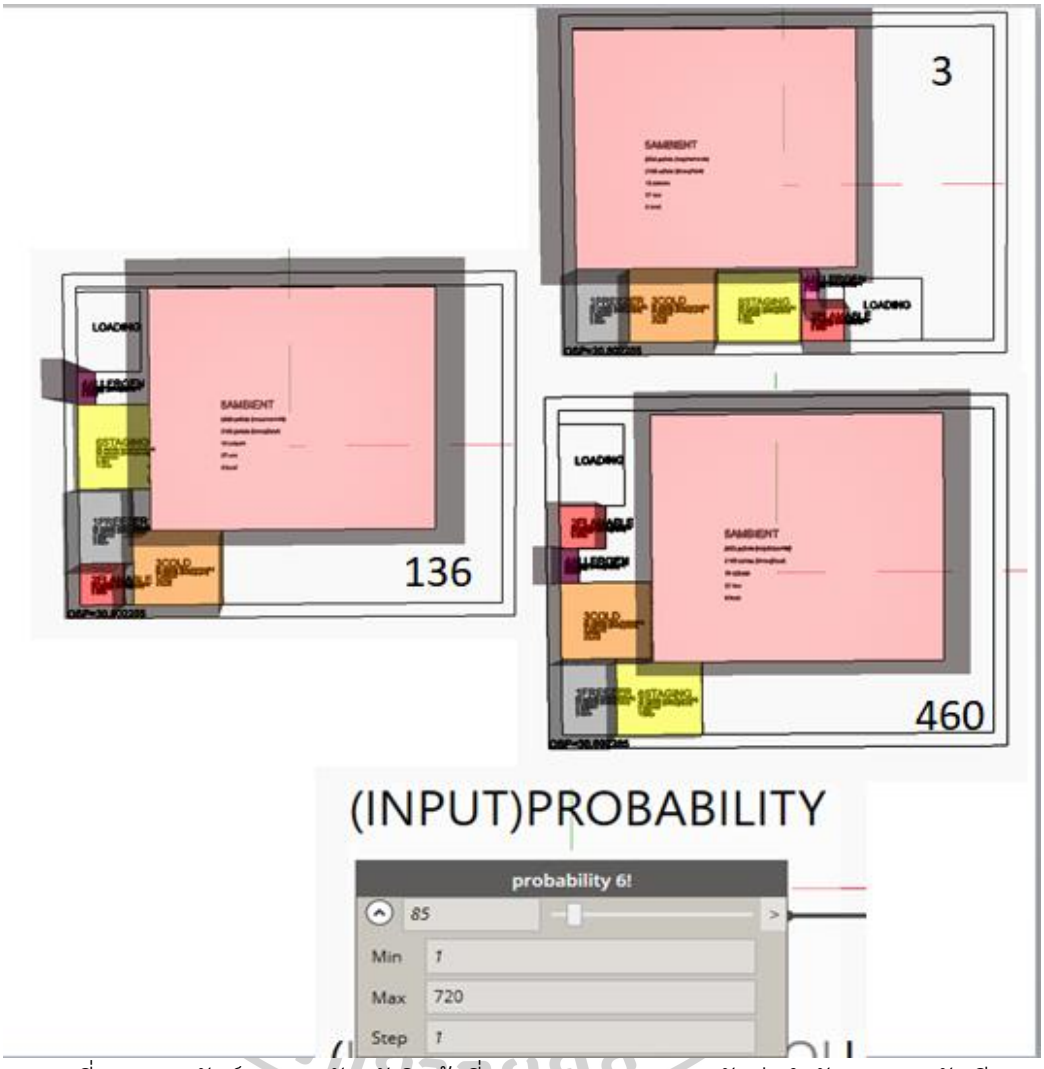
2) Input2 ช่วง Probability (วิธีที่ 1- วิธีที่ n!)

กำหนดวิธีการของลำดับของการจัดเรียง (Permutation)ให้แตกต่างกันในแต่ละผลลัพธ์การวางผังคลังสินค้า



ภาพที่ 191 Node ที่ใช้ปรับค่า Input2 ช่วง Probability

โดยค่าของลำดับของการจัดเรียงจะส่งผลให้เกิดผลลัพธ์การวางผังคลังสินค้าที่แตกต่างและหลากหลาย ดังภาพด้านล่าง

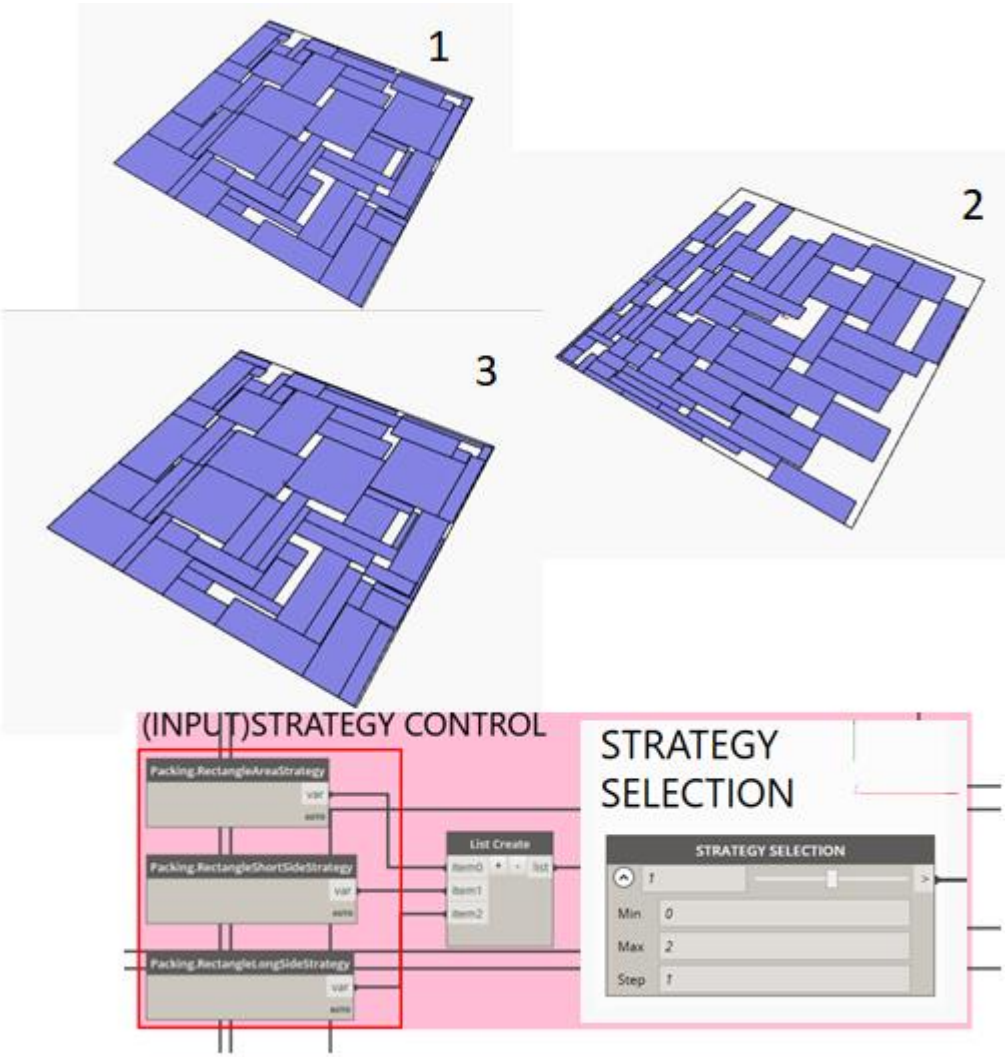


ภาพที่ 192 ผลลัพธ์การวางผังคลังสินค้าที่หลากหลายจากการปรับค่าลำดับของการจัดเรียง

3) Input3 ช่วงstrategy (รูปแบบที่1 - รูปแบบที่3)

ทดลองเลือกกระบวนการจัดเรียงพื้นที่ (Strategy) 3รูปแบบ คือ รูปแบบที่1 (Short Side Strategy), รูปแบบที่2 (Long Side Strategy), รูปแบบที่3 (Area strategy)โดยแต่ละรูปแบบจะส่งผลให้เกิดผลลัพธ์การวางผังคลังสินค้าที่แตกต่างและหลากหลาย ดังภาพด้านล่าง

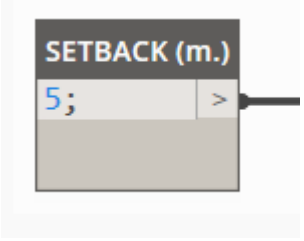
ภาพที่ 193 Node ที่ใช้ปรับค่า Input3 ช่วง Strategy



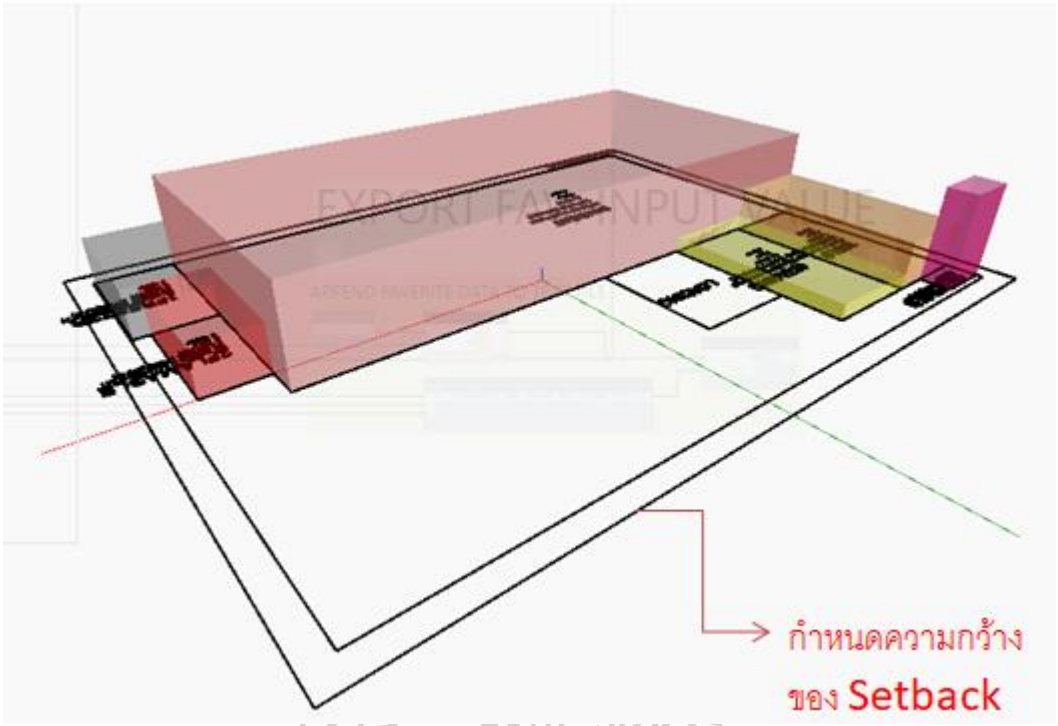
ภาพที่ 194 ผลลัพธ์การวางผังคลังสินค้าที่หลากหลายจากการปรับค่า Strategy

4) Input4 ค่าSetback Values

ผู้ใช้งานโปรแกรมสามารถระบุค่าระยะร่นที่Nodeนี้โดยการพิมตัวเลขลงไปซึ่งค่าจะเป็นค่าของระยะร่นทุกด้านที่จะเกิดขึ้นกับพื้นที่ดินโดยจะมีหน่วยเป็นเมตร



ภาพที่ 195 Node ที่ใช้ปรับค่า Input4 ค่าSetback Values

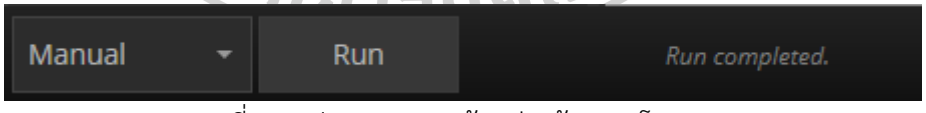


ภาพที่ 196 ผลลัพธ์การวางผังคลังสินค้าที่หลากหลายจากการปรับค่า Setback

(3) เริ่มการประมวลผล

หลังจากการกำหนดค่า Input ครบแล้วการเริ่มประมวลผลใน Dynamo สามารถเลือกได้2 รูปแบบ

รูปแบบที่ 1 คือแบบManual คือผู้ใช้งานโปรแกรมจะต้องกดที่ปุ่ม Run ทางด้านล่างซ้ายของโปรแกรมทุกครั้งที่ต้องการประมวลผลลัพธ์ใหม่

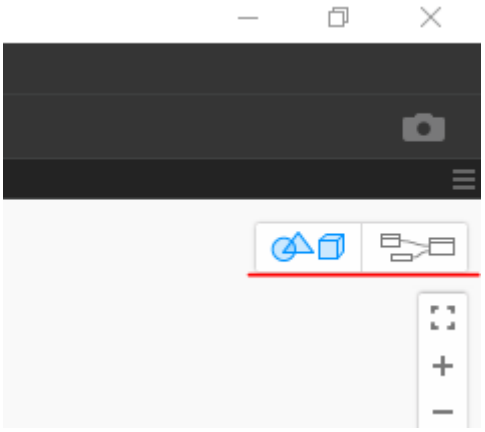


ภาพที่ 197 ปุ่ม Run ทางด้านล่างซ้ายของโปรแกรม

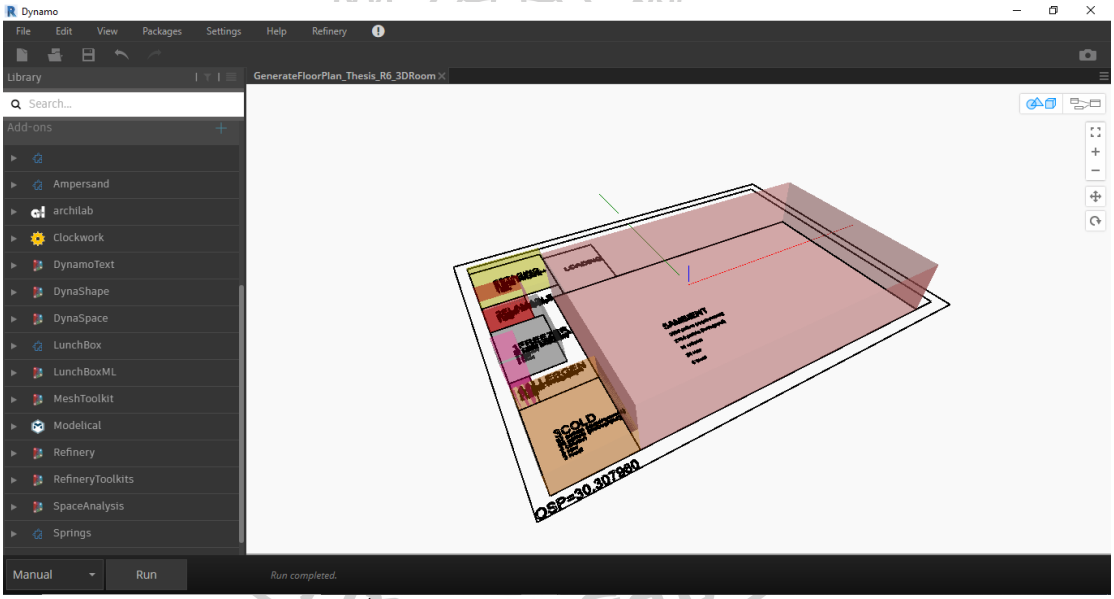
รูปแบบที่ 2 คือแบบ Automatic คือทุกครั้งที่ผู้ใช้งานเปลี่ยนแปลงค่า Input โปรแกรมจะประมวลผลลัพธ์ใหม่อัตโนมัติ แต่ข้อควรระวังคือหากคอมพิวเตอร์ที่ใช้มีประสิทธิภาพการประมวลผลที่ไม่สูงพออาจจะทำให้การประมวลผลของโปรแกรมใช้เวลานาน

มุมมองการแสดงผลจะปรับได้สองรูปแบบ ซึ่งอยู่บริเวณมุมบนขวา

1) สัญลักษณ์รูปเรขาคณิตจะเลือกเพื่อแสดงมุมมองการแสดงผลส่วนโมเดลสามมิติซึ่งในโปรแกรมนี้จะแสดงผลของผังคลังสินค้า

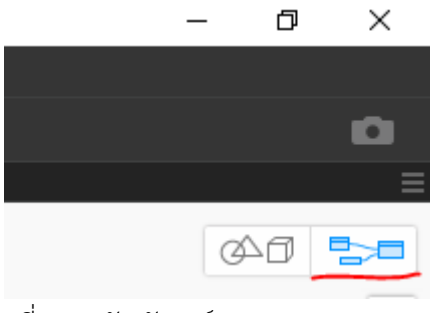


ภาพที่ 198 สัญลักษณ์มุมมองแสดงผลส่วนโมเดลสามมิติ

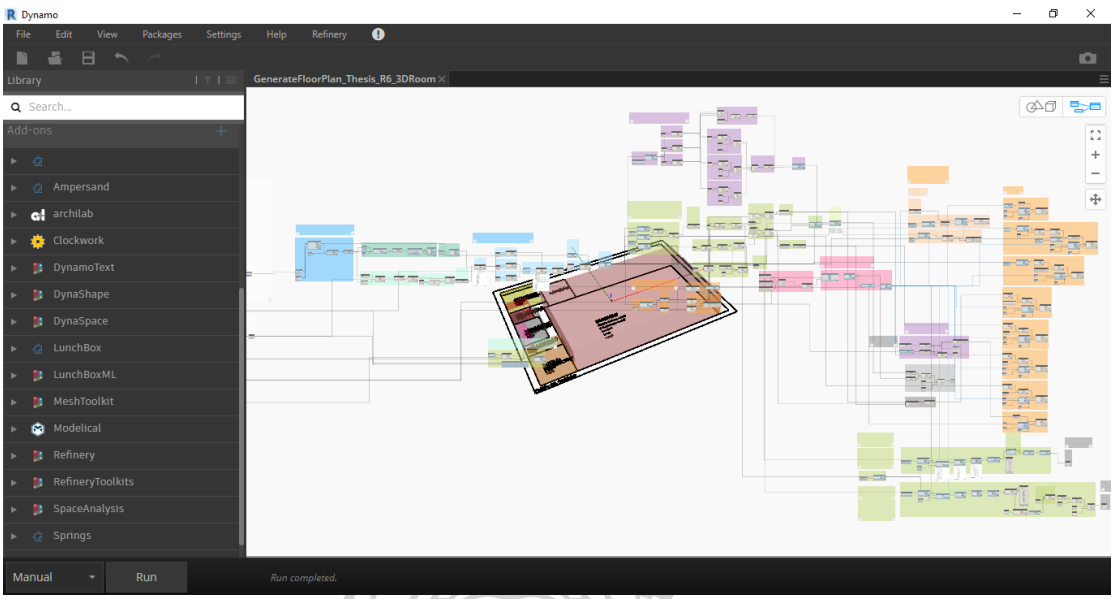


ภาพที่ 199 มุมมองแสดงผลส่วนโมเดลสามมิติ

2) สัญลักษณ์รูปสี่เหลี่ยมมีเส้นโยงหากันจะเลือกเพื่อแสดงมุมมองการแสดงผล Node หรือคำสั่งที่เป็นรายละเอียดของโปรแกรม นี้



ภาพที่ 200 สัญลักษณ์มุมมองแสดงผล Node



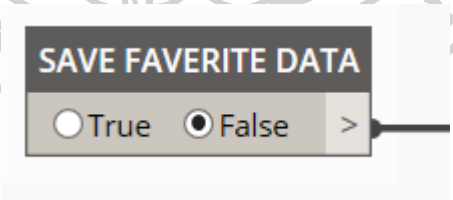
ภาพที่ 201 มุมมองแสดงผล Node

(4) เลือกบันทึกค่า Input ที่ได้ผลลัพธ์ตรงตามความต้องการ

ขั้นตอนสุดท้ายหากผู้ใช้งานต้องการบันทึกค่า Input ของผลลัพธ์การวางผังคลังสินค้าที่ผู้ใช้งานพึงพอใจ สามารถเลือกได้จาก Nodeนี้ โดยเป็นลักษณะของการเลือก2 ค่าดังนี้

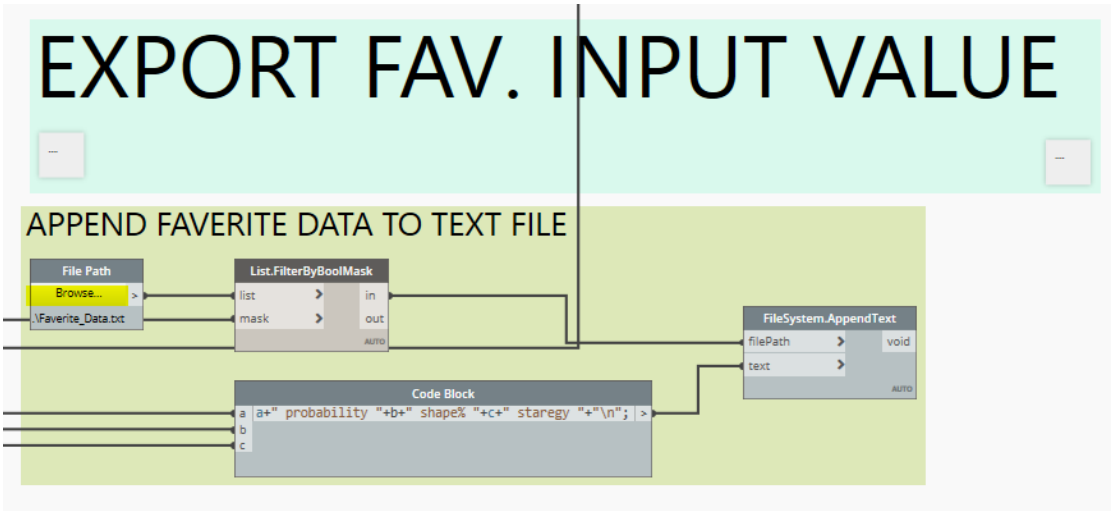
เลือก True ถ้าต้องการบันทึกค่า

เลือก False ถ้าไม่ต้องการบันทึกค่า



ภาพที่ 202 ปุ่มเลือกในการบันทึกค่า

โดยค่าของ Input ที่บันทึกจะออกไปลักษณะ Text file โดยผู้ใช้งานจำเป็นต้องสร้าง Text file ขึ้นมาก่อนจากนั้นเลือก Browse ไปยัง Text file ที่สร้างไว้

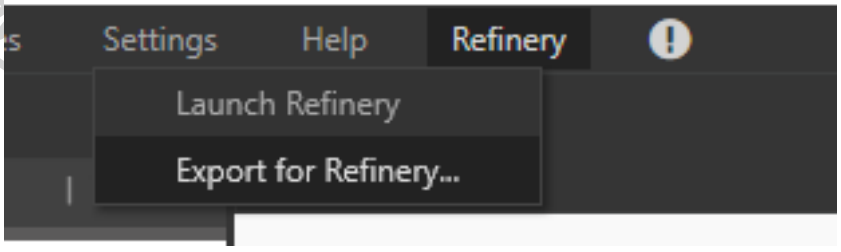


ภาพที่ 203 โปรแกรมในส่วนบันทึกค่า Input ที่ได้ผลลัพธ์ตรงตามความต้องการ

4.1.3 โปรแกรมวางผังคลังสินค้าส่วน Refinery

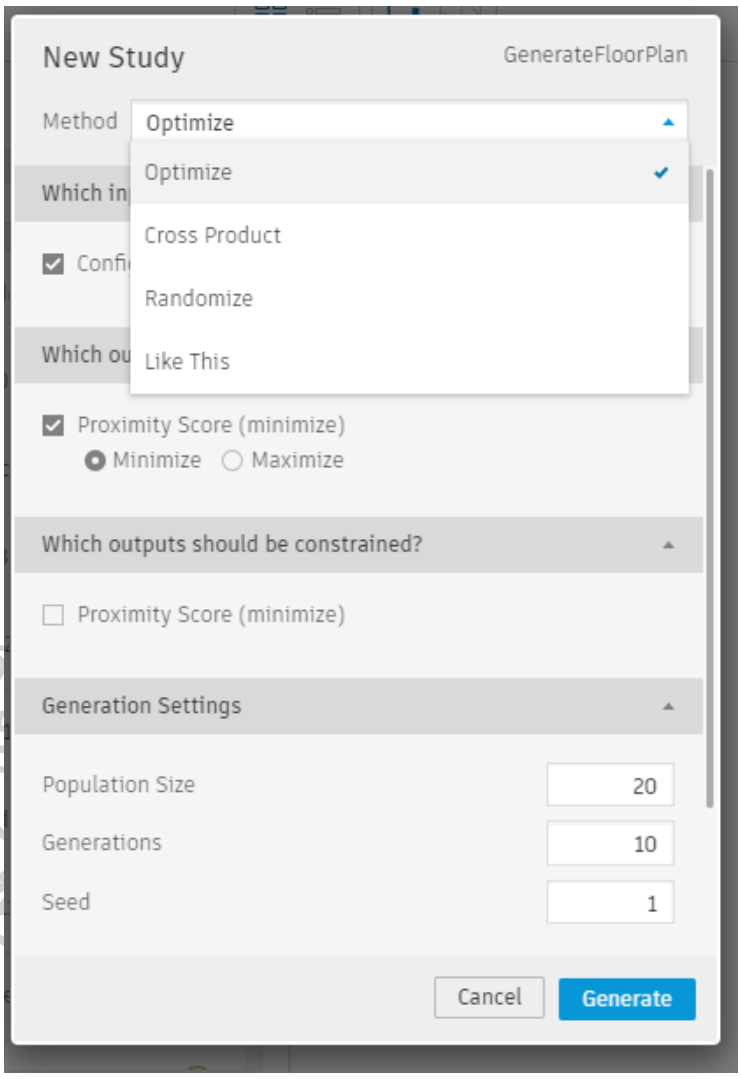
ขั้นตอนการส่งต่อข้อมูลจาก Dynamo ไปประมวลผลต่อเพื่อสร้างรูปแบบการวางผังที่หลากหลายมีขั้นตอนดังนี้

1) ขั้นตอน 1 ทำได้โดยการเลือกคำสั่งบนแถบคำสั่งด้านบน เลือก Refinery > Export for Refinery จะขึ้นข้อความ Complete เป็นอันเรียบร้อย จากนั่นเลือก Refinery > Launch Refinery เพื่อเป็นการเปิดโปรแกรม Refinery ขึ้นมา

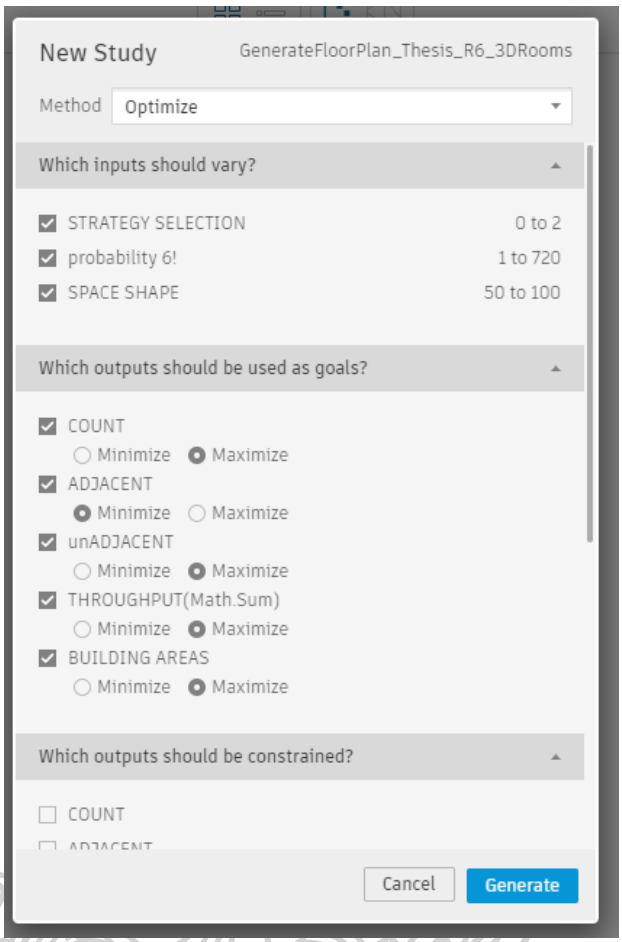


2) ขั้นตอน 2 ในหน้าต่างโปรแกรม Refinery เลือก New Study > WAREHOUSE PLANNING PROGRAM จากนั่นจะขึ้นหน้าต่างให้ปรับค่าช่วง Input และ Output ที่ผู้ใช้งานต้อง โดยสามารถเลือกวิธีการของการประมวลผลได้ 4 วิธีคือ Optimize, Cross Product, Randomize, LikeThis ในกรณีของ โปรแกรมวางผังคลังสินค้านี้เหมาะกับ วิธี Optimize ซึ่งเป็นการกำหนดเป้าหมายของผลลัพธ์ที่ต้องการให้เกิด ดังนั่นหากผลลัพธ์

ของการวางผังรูปแบบใดที่ไม่ตรงกับเป้าหมายจะไม่ถูกคัดออก เนื่องจากห้ามมีพื้นที่ที่ไม่ควรใกล้กันเกิดขึ้นในผลลัพธ์จึงควรใช้วิธีนี้ รายละเอียดการเลือกกำหนดเป้าหมายสามารถดูได้จาก ข้อที่ 3.1.3 ใช้ Refinery ในการสร้างและกำหนดผลลัพธ์การวางผังคลังสินค้าที่หลากหลาย



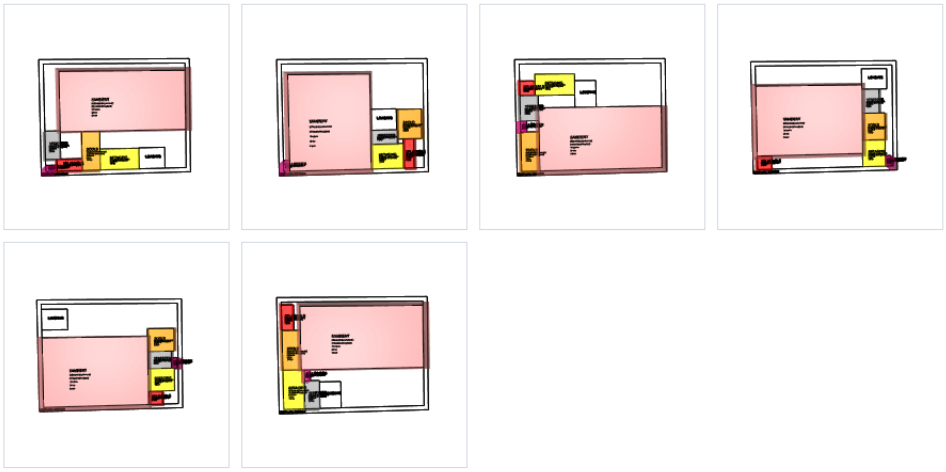
ภาพที่ 204 การประมวลผล 4 วิธีคือ Optimize, Cross Product, Randomize, Like This



ภาพที่ 205 หน้าต่างเมื่อเลือกการประมวลผลวิธี Optimize สามารถกำหนดผลลัพธ์ที่ได้

3) ขั้นตอน 3 เมื่อกำหนดเป้าหมายของผลลัพธ์เรียบร้อย เลือก Generate โปรแกรมจะประมวลผลและสร้างผลลัพธ์โดยด้านบนของโปรแกรมจะเป็นส่วนที่แสดงค่าของลลัพธ์ของรูปแบบผังคลังสินค้านั่นๆ ว่าเกิดจาก Input และ output อะไร ส่วนล่างของโปรแกรมจะแสดงผลลัพธ์สามมิติของ ผังคลังสินค้าในรูปแบบที่หลากหลายเพื่อใช้ในพิจารณาเลือกแบบที่เหมาะสมต่อไป

ภาพที่ 206 ส่วนบนของโปรแกรมแสดงค่า Input และ output ของผลลัพธ์ที่เกิดขึ้น



ภาพที่ 207 ส่วนล่างของโปรแกรมจะแสดงผลลัพธ์สามมิติของผังคลังสินค้าในรูปแบบที่หลากหลาย

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัยโปรแกรมวางผังคลังสินค้าจากการนำไปใช้งานจริง

จากการนำโปรแกรมการวางผังคลังสินค้าไปทดลองใช้จริงในบริษัทที่ผู้ทำวิจัยทำงานอยู่ ซึ่งเกี่ยวกับการเป็นที่ปรึกษาและออกแบบงานสถาปัตยกรรมโรงงานอุตสาหกรรม โดยโครงการที่ใช้ทดลองนั่นเป็นโครงการคลังสินค้าเก็บวัตถุดิบเพื่อผลิตเครื่องดื่มชูกำลังยี่ห้อหนึ่งในประเทศไทย โดยทดลองโดยเปลี่ยนขั้นตอนการทำงานรูปแบบเดิมจากที่อธิบายไว้ในบทที่ 2 ข้อ 2.1 ขั้นตอนการวางผังคลังสินค้าในปัจจุบัน มาเป็นการใช้โปรแกรมวางผังคลังสินค้า โดยมีผู้ใช้งานแบ่งตามขั้นตอนของการใช้งานโปรแกรม ดังนี้

ขั้นตอนที่ 1 ใช้โปรแกรมวางผังคลังสินค้าส่วน Excel รับข้อมูลจากลูกค้าโดยสถาปนิกเจ้าของโครงการ แทนการจดบันทึกแบบเดิมทำให้ได้ข้อมูลที่ตรงต่อการนำไปออกแบบวางผังคลังสินค้า โดยมีข้อมูลดังต่อไปนี้

1) ขอบเขตที่ดิน (Area) = 85 x 65 เมตร
2) ประเภทรถที่ใช้งาน (Car Type) และ จำนวน (Number Of Car Type) = รถ 6 ล้อ 3 คัน
3) พื้นที่ที่ต้องการ (Space) = 6 พื้นที่ ประกอบไปด้วย Freezer, Flammable, Cold, Allergen, Ambient, Staging
4) ปริมาณที่ต้องการจัดเก็บสินค้า (Capacity) = เรียงตามพื้นที่ 54, 30, 87, 12, 2088, 32
5) การซ้อนชั้นของสินค้า (Level) เรียงตามพื้นที่ 3, 3, 3, 6, 6, 1

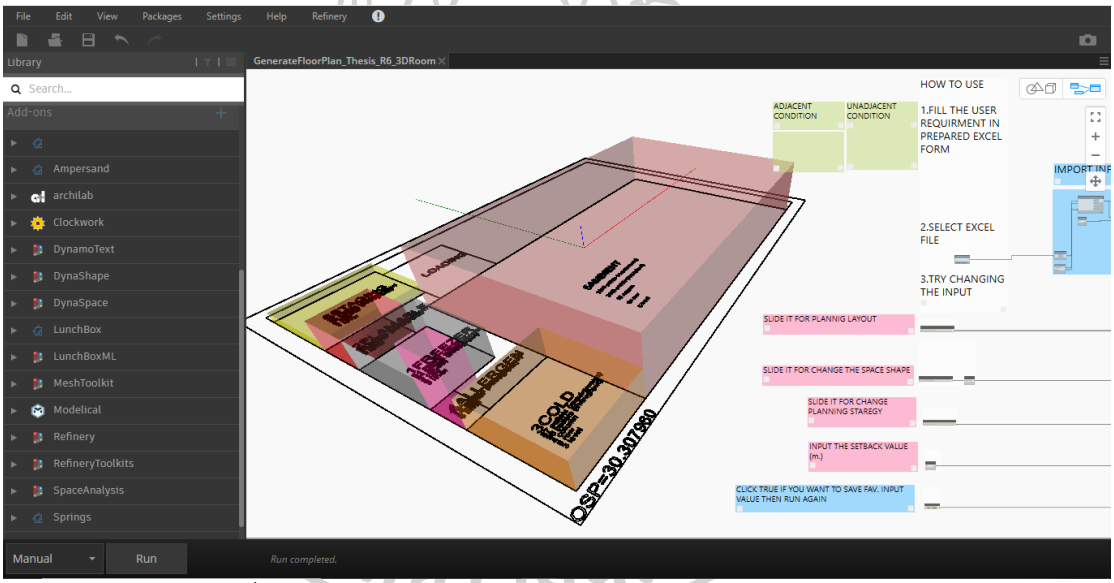
เงื่อนไข

6) เงื่อนไขการควรใกล้กันของพื้นที่ (Adjacent) = พื้นที่ Freezer และ Cold ควรอยู่ใกล้กัน , พื้นที่ Ambient และ Staging ควรอยู่ใกล้กัน
7) เงื่อนไขการไม่ควรใกล้กันของพื้นที่ (Nonadjacent) = พื้นที่ Flammable ห้ามติดกับ Allergen

	A	B	C	D	E	F	G	H	I	J	K
1	AREA	CAR TYPE	NUMBER OF CAR TYPE	SPACE ID	SPACE NAME	CAPACITY	RACK LEVEL	/	COLOUR	ADJACENT SPACES	UNADJACENT SPACES
2	85	PICKUP	0	0	FREEZER	54	3		128,128,128	COLD	
3	65	TRUCK	3	1	FLAMABLE	30	3		255,0,0		ALLERGEN
4		TRAILER	0	2	COLD	87	3		255,128,0	FREEZER	
5				3	ALLERGEN	12	6		255,0,128		FLAMABLE
6				4	AMBIENT	2088	6		255,128,128	STAGING	
7				5	STAGING	32	1		255,255,0	AMBIENT	

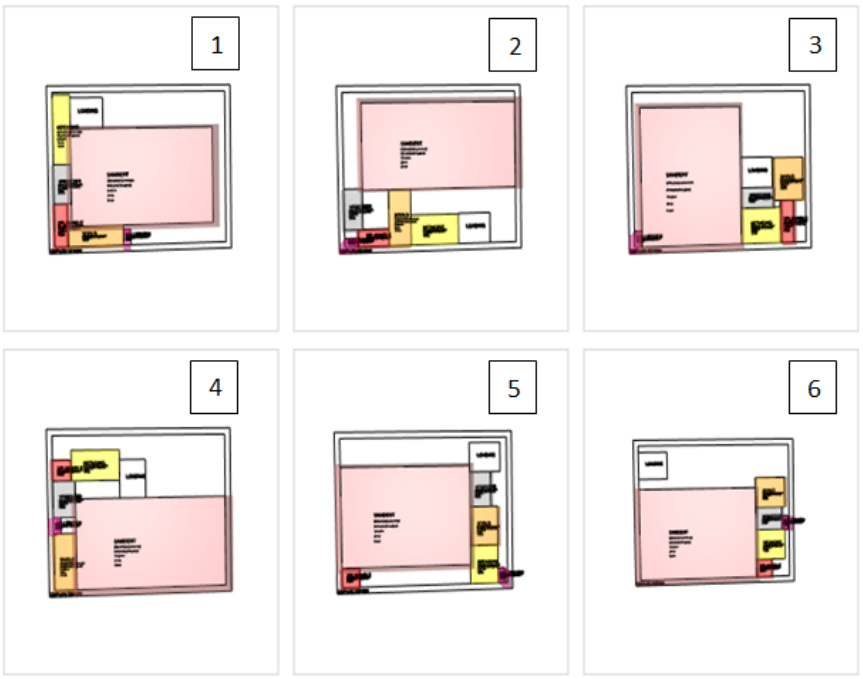
ภาพที่ 208 ผลการใช้โปรแกรมวางผังคลังสินค้าส่วน Excel รับข้อมูลจากลูกค้า

ขั้นตอนที่ 2 ใช้โปรแกรมวางผังคลังสินค้าส่วน Dynamo นำเข้าข้อมูลเพื่อประมวลผล โดยโปรแกรมโดยสถาปนิกเจ้าของโครงการ โดยสถาปนิกได้ทดลองเปลี่ยนแปลงค่า Input เกิดขึ้นเพื่อดูผลลัพธ์ที่เปลี่ยนไปของผังคลังสินค้า ตามวิธีการในข้อ 4.1.2.1 วิธีการใช้งานโปรแกรมวางผังคลังสินค้าส่วน Dynamo



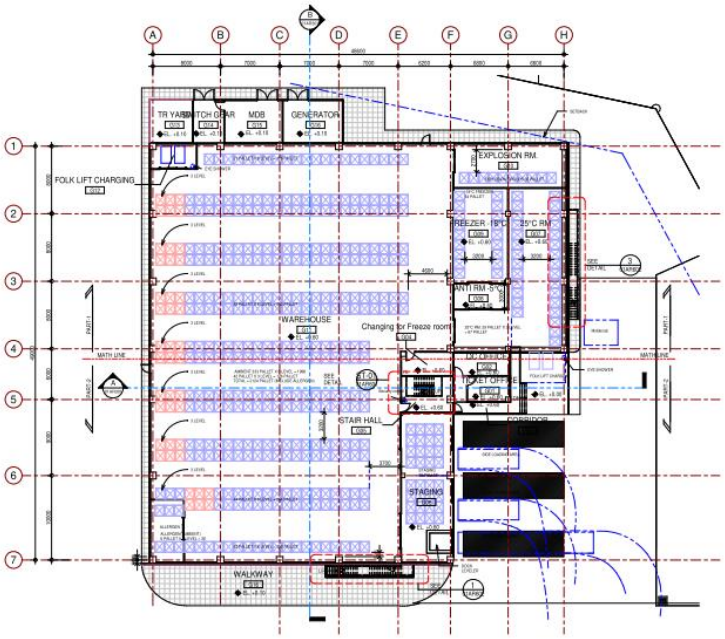
ภาพที่ 209 ผลลัพธ์การใช้โปรแกรมวางผังคลังสินค้าส่วน Dynamo

ขั้นตอนที่ 3 ใช้โปรแกรมวางผังคลังสินค้าส่วน Refinery กำหนดเป้าหมายของผลลัพธ์การวางผังคลังสินค้า เพื่อให้โปรแกรมสร้างผังคลังสินค้าที่หลากหลายโดยโปรแกรมโดยสถาปนิกเจ้าของโครงการ และพิจารณาและเลือกผลลัพธ์สุดท้ายของผังคลังสินค้าโดยทีมออกแบบประกอบไปด้วย หัวหน้าแผนกสถาปัตยกรรม สถาปนิกเจ้าของโครงการ สถาปนิกที่ปรึกษาอาวุโส และ ทีมเจ้าของโครงการ ประกอบไปด้วย เจ้าของโครงการ ผู้ใช้งานโครงการโดยตรง และ ฝ่ายความปลอดภัยอาคาร รูปด้านล่างแสดงผลลัพธ์ของการประมวลผลที่เกิดขึ้น ซึ่งผลสรุปมีการเลือกผลลัพธ์หมายเลข 3 เพื่อนำมาพัฒนาแบบต่อในขั้นตอนของแบบก่อสร้างต่อไป



ภาพที่ 210 ผลลัพธ์การใช้โปรแกรมวางผังคลังสินค้าส่วน Refinery

จากผลลัพธ์หมายเลข 3 ถูกนำพัฒนาต่อโดยการปรับเปลี่ยนฝั่งของลานโหลดสินค้า (พื้นที่สีขาว) จากด้านบนมาเป็นด้านล่าง เพื่อให้เหมาะสมกับเส้นทางเดินรถจริงและนำไปสู่การพัฒนาแบบต่อในรายละเอียดต่อไปตามภาพด้านล่าง



ภาพที่ 211 ผลลัพธ์ของแบบที่ถูกพัฒนามาจากโปรแกรมวางผังคลังสินค้า

จากการใช้งานโปรแกรมวางผังคลังสินค้ากับโครงจริงพบว่าสามารถแก้ปัญหาของการวางผังในปัจจุบันที่เป็นที่มาของการทำการวิจัยได้ โดยอ้างอิงจากบทที่ 2 ข้อที่ 2.2 คือปัญหาในเรื่องของความซับซ้อนของวัตถุที่จัดเก็บซึ่งส่งผลให้การวางผังคลังสินค้าซับซ้อนตามไปด้วย และ ปัญหาเรื่องเวลาการทำงานที่ไม่สอดคล้องกับปริมาณและความซับซ้อนของงาน

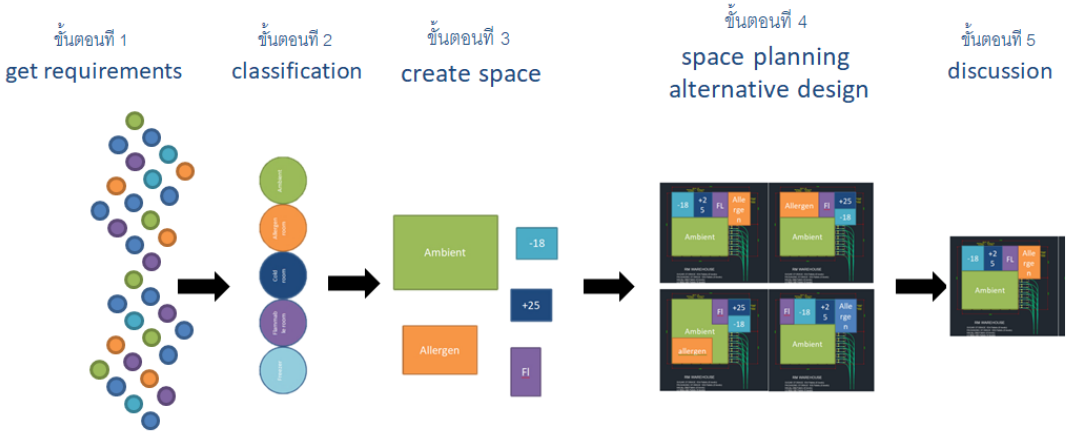
5.1.1 สรุปผลโปรแกรมวางผังคลังสินค้ากับปัญหาความซับซ้อนของการวางผังคลังสินค้า

การใช้โปรแกรมวางผังคลังสินค้าในการทำงานก่อให้เกิดขั้นตอนการทำงานแบบใหม่ที่ลดความซับซ้อนของการวางผังคลังสินค้าลงจากการทำงานแบบปัจจุบันโดยสถาปนิก

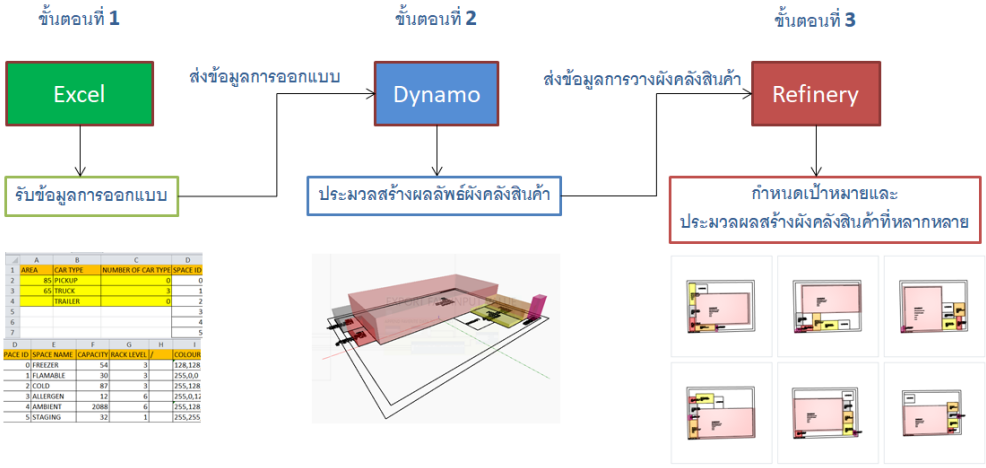
5.1.1.1 สรุปผลขั้นตอนการทำงานด้วยโปรแกรมวางผังคลังสินค้า

หากเปรียบเทียบกับขั้นตอนการทำงานแบบปัจจุบันที่มีขั้นตอนการทำงานถึง 5 ขั้นตอนที่ใช้เวลาและความซับซ้อนสูง การทำงานโดยใช้โปรแกรมวางผังคลังสินค้าจะช่วยลดขั้นตอนการทำงานเหลือเพียง 3 ขั้นตอน ดังแสดงให้เห็นในแผนภาพเปรียบเทียบด้านล่าง

5 ขั้นตอนการทำงานแบบปัจจุบัน



3 ขั้นตอนการทำงานโดยใช้โปรแกรมวางผังคลังสินค้า



ภาพที่ 212 เปรียบเทียบขั้นตอนการวางผังแบบเดิมกับการวางผังคลังสินค้าด้วยโปรแกรม

5.1.1.2 สรุปผลความถูกต้องของผลลัพธ์จากโปรแกรมวางผังคลังสินค้า

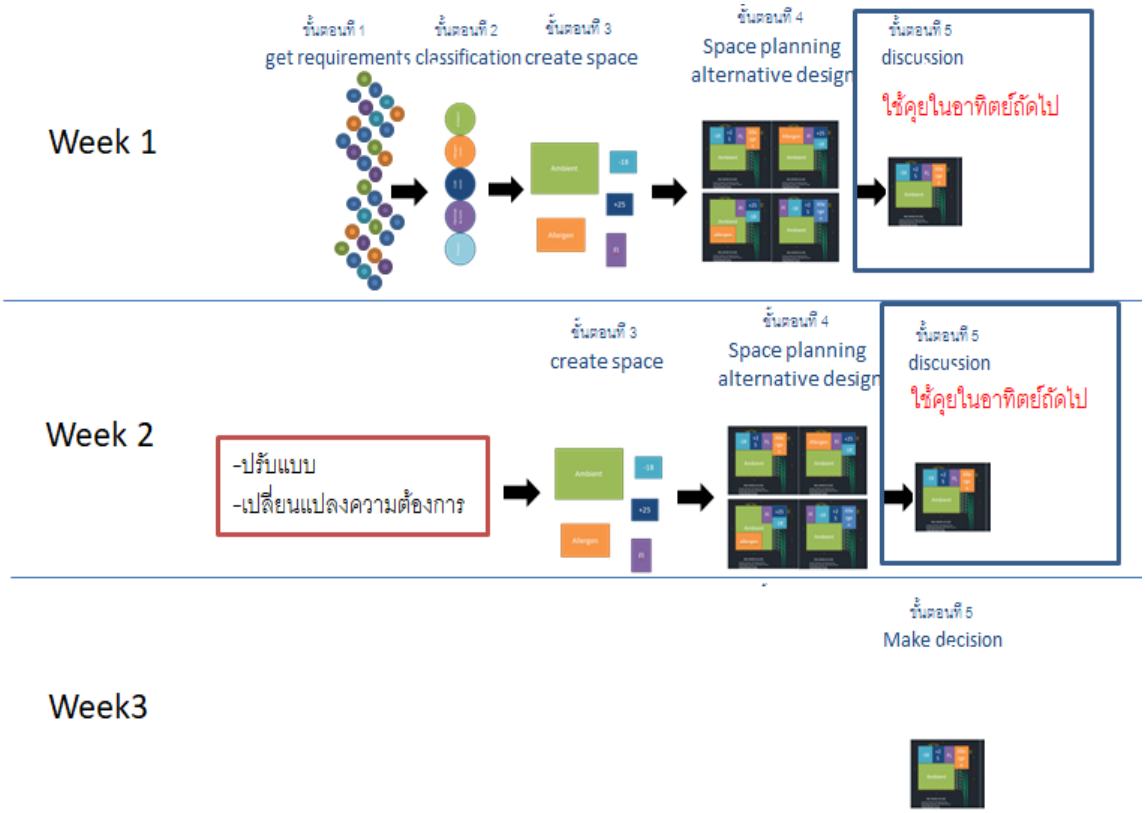
จากการทดลองใช้โปรแกรมวางผังคลังสินค้ากับขั้นตอนการทำงานแบบใหม่นั่นพบว่า ยังเพิ่มประสิทธิภาพของการทำงานทั้งในแง่ของปริมาณผลลัพธ์ของผังคลังสินค้าที่ได้และคุณภาพคือความถูกต้องของงานอีกด้วย ดังเห็นได้จากผลลัพธ์ของโครงการคลังสินค้าเครื่องดื่มชูกำลัง ที่โปรแกรมทำการประมวลผลจากข้อมูลความต้องการของลูกค้า ภายใต้เงื่อนไขในการจัดเก็บสินค้าคือระหว่างพื้นที่ห้อง Allergen จะไม่อยู่ติดกับห้อง Flamable อันเนื่องมาจากเหตุผลเรื่องความปลอดภัยและการจัดเก็บสารเคมี และ ห้อง Cold room ควรอยู่ใกล้กับ ห้อง Freezer เพื่อประหยัดพลังงานและลดการสูญเสียความเย็นเมื่อเปิดใช้ห้อง โดยผลลัพธ์ของผังคลังสินค้าที่หลากหลายโดยโปรแกรมสามารถแสดงผลได้ถูกต้องตามเงื่อนไขที่ผู้ใช้งานกำหนด ดังภาพด้านล่างสังเกตุว่าไม่มีพื้นที่สีแดง (Flamable) ติดกับพื้นที่สีม่วง (Allergen) เลย

ภาพที่ 213 ผลลัพธ์ของผังคลังสินค้าที่หลากหลายและถูกต้องตามเงื่อนไขที่ผู้ใช้งานกำหนด

5.1.2 สรุปผลโปรแกรมวางผังคลังสินค้ากับเวลา

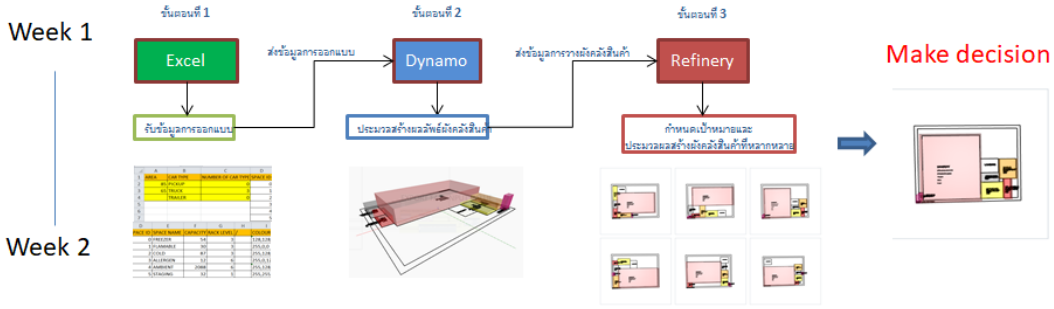
การใช้โปรแกรมวางผังคลังสินค้าในการทำงานก่อให้เกิดขั้นตอนการทำงานแบบใหม่ที่ยืดหยุ่นและสอดคล้องกับเวลามากกว่าการทำงานแบบปัจจุบันโดยสถาปนิก

โดยหากเปรียบเทียบกับขั้นตอนการทำงานแบบปัจจุบันที่มีขั้นตอนการทำงานถึง 5 ขั้นตอนที่ใช้เวลาและความซับซ้อนสูง โดยการที่จะได้ผลลัพธ์ในการคุยกันในครั้งแรกของสัปดาห์ที่ 1 นั่นเป็นไปไม่ได้เลย จึงจำเป็นต้องเวลาถึง 3 สัปดาห์ เพื่อการสรุปแบบ



ภาพที่ 214 แผนภาพแสดงเวลากับขั้นตอนการทำงานวางผังคลังสินค้าแบบเดิม

แต่กลับกันหากวัตถุที่จัดเก็บไม่มีความซับซ้อนมากกล่าวคือลูกค้าหรือผู้ใช้งานสามารถจัดกลุ่มพื้นที่การใช้งานได้เลยในสัปดาห์ที่ 1 ของการประชุม สถาปนิกสามารถใช้โปรแกรมวางผังคลังสินค้าประมวลผลเบื้องต้นของการวางผังคลังสินค้าให้กับลูกค้าได้เลยในการประชุมครั้งนั้น ซึ่งเป็นข้อได้เปรียบที่การทำงานแบบปัจจุบันทำไม่ได้ จากการทดลองใช้ในการทำงานจริงพบว่าสามารถสรุปเลือกแบบได้ภายใน 2 สัปดาห์ โดยผลลัพธ์ที่หลากหลายและถูกต้องตามเงื่อนไขอีกด้วย



ภาพที่ 215 แผนภาพแสดงเวลากับขั้นตอนการทำงานวางผังคลังสินค้าด้วยโปรแกรมวางผังคลังสินค้า

มหาวิทยาลัยศิลปากร

รายการอ้างอิง

(GHS), G. H. S. (2020). Chemical Hazard Classification. Access Date June 30,2019 Retrieved from https://drs.illinois.edu/Page/SafetyLibrary/ChemicalHazardClassification

Das, S. (2016). Automated Service Core Generator in Autodesk Dynamo. *DESIGN TOOLS,* 2, 217-225.

kargo. (2017). Which Is The Best Truck Size For You. Access Date November 28,2019. Retrieved from https://kargo.tech/artikel/which-is-the-best-truck-size-for-you/

Loscam. (2020). Loscam Wooden Pallet – Asia. Access Date December 12,2019. Retrieved from https://www.loscam.com/en/product-detail.php?id=2

macrack. (2017). How To Calculate Warehouse Rack Space. Access Date March 30,2020. Retrieved from https://www.macrack.com.au/how-to-calculate-warehouse-rack-space/

MATTERLAB. (2020). Refinery Toolkit. Access Date March 1,2020. Retrieved from https://dynamobim.org/refinery-toolkit/

Mecalux, I. (2018). How to Calculate Warehouse Capacity. Access Date June 28,2020. Retrieved from https://www.interlakemecalux.com/blog/calculate-warehouse-capacity

Synergysoft. (2017). generative-design. Access Date June 28,2020. Retrieved from http://synergysoft.co.th/special-article/25-mfg-special-articles/321-generative-design

ประวัติผู้เขียน

ชื่อ-สกุล	นายวัชราวุธ เจนเรือ
วัน เดือน ปี เกิด	28 สิงหาคม 2536
สถานที่เกิด	จังหวัดภูเก็ต
วุฒิการศึกษา	พ.ศ. 2555 สถาปัตยกรรมศาสตรบัณฑิต (สถ.บ.) สถาปัตยกรรมไทย (เกียรตินิยมอันดับ 2) มหาวิทยาลัยศิลปากร
ที่อยู่ปัจจุบัน	399/45 ถนนเยาวราช ตำบลตลาดใหญ่ อำเภอเมือง จังหวัดภูเก็ต 83000