ระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี

Looking Great

นางสาวนฤมล โมคภา

โครงงานนี้เป็นส่วนหนึ่งของการศึกษา

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

ภาควิชาคณิตศาสตร์ สถิติ และคอมพิวเตอร์ คณะวิทยาศาสตร์

มหาวิทยาลัยอุบลราชธานี

ปีการศึกษา 2562

ลิขสิทธิ์ของมหาวิทยาลัยอุบลราชธานี

โครงงาน	:	ระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี Looking Great
โดย	:	นางสาวนฤมล โมคภา
อาจารย์ที่ปรึกษา	:	ดร. วิชิต สมบัติ
ระดับการศึกษา	:	วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
ปีการศึกษา	:	2562

ได้รับการพิจารณาให้เป็นส่วนหนึ่งของการศึกษา

ตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์

คณะกรรมการสอบประเมินความรู้โครงงานคอมพิวเตอร์

. . . . . อาจารย์ที่ปรึกษา

(ดร. วิชิต สมบัติ)

. . . . . กรรมการ

(ดร. ไพชยนต์ คงไชย)

. . . . . กรรมการ

(อาจารย์ วาสนา เหง้าเกษ)

. . . . . หัวหน้าภาควิชา

(ดร. ชัชวิน นามมั่น)

วันที่ . . . . . / . . . . . / . . . . .

กิตติกรรมประกาศ

การพัฒนาโครงงานระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี สำเร็จลุล่วงได้ด้วยความกรุณาแลความช่วยเหลือจากหลายท่าน ถึงแม้จะประสบกับปัญหาในการทำโครงงานก็ได้รับคำแนะนำและการช่วยเหลือที่ดีเสมอมา ข้าพเจ้าจึงใคร่ขอขอพระคุณทุกท่าน ที่มีส่วนร่วมในการพัฒนาโครงงานนี้

ขอขอบพระคุณอาจารย์ที่ปรึกษา ดร.วิชิต สมบัติ อาจารย์ที่ปรึกษาโครงงานที่ได้แนะนำทฤษฎีและแนวทางในแก้ปัญหาต่าง ๆ ที่เกิดขึ้นระหว่างการพัฒนาระบบ อีกครั้งยังคอยตรวจสอบความก้าวหน้าของการทำงานเป็นระยะ ๆ รวมทั้งสร้างกำลังใจให้ผู้พัฒนาอยู่เสมอ

ขอขอบพระคุณอาจารย์ประจำสาขาวิทยาการคอมพิวเตอร์ เจ้าหน้าที่ภาควิชาคณิตศาสตร์ สถิติและคอมพิวเตอร์ ที่คอยเอื้อยอำนวยความสะดวกทั้งเรื่องอุปกรณ์และสถานที่ต่อการปฏิบัติงานของผู้พัฒนา

ขอกราบขอบพระคุณบิดา มารดา ที่คอยให้กำลังใจ คอยให้ความรักและความห่วงใยเสมอมาตลอดจนคอยช่วยเหลือทุนทรัพย์ทางด้านการศึกษาและอุปกรณ์ในการพัฒนาโครงงาน

ขอบคุณเพื่อน ๆ สาขาวิทยาการคอมพิวเตอร์ชั้นปีที่ 4 ที่ได้คอยช่วยแก้ไขปัญหาและให้คำปรึกษาในการพัฒนาโครงงานครั้งนี้จนเสร็จสิ้น

นางสาวนฤมล โมคภา

วันที่ 12 เมษายน 2563

โครงงาน	:	ระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี
โดย	:	นางสาวนฤมล โมคภา
อาจารย์ที่ปรึกษา	:	ดร. วิชิต สมบัติ
ระดับการศึกษา	:	วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
ปีการศึกษา	:	2562

บทคัดย่อ

ระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี เป็นเว็บแอปพลิเคชันที่เกี่ยวกับเรื่องของการเลือกซื้อเสื้อผ้า การเลือกเสื้อผ้าให้เหมาะสมกับรูปร่างของผู้ใช้งานแต่ละท่าน เป็นระบบที่เป็นทางเลือกให้แก่ผู้ใช้งานได้ใช้บริการ เพื่อเป็นตัวช่วยในการเลือกเสื้อผ้าให้แก่ผู้ใช้ได้ง่ายยิ่งขึ้น ผู้พัฒนาจึงวัตถุประสงค์พัฒนาระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี เพื่อเป็นทางเลือกในการแนะนำการแต่งกายให้กับผู้ใช้งานที่มีความสนใจในเรื่องการแต่งกาย โดยระบบนี้พัฒนาด้วย Django Web Framework โดยใช้เทคนิค Collaborative Filtering เพื่อประมวลผลในฟังก์ชันแนะนำเสื้อผ้าให้แก่ผู้ใช้งาน เว็บไซต์เปิดใช้งานบน Pythonanywhere ซึ่งเปิดให้ใช้ฐานข้อมูล SQLite3 โดยไม่มีค่าใช้จ่าย

คำสำคัญ: เว็บแอปพลิเคชัน ระบบแนะนำการแต่งกาย

Topic	:	Looking Great
Author	:	NARUMON MOKPHA
Advisor	:	Wichit Sombat, Ph.D.
Degree	:	Bachelor of Science (Computer Science)
Academic Year	:	2019

Abstract

Looking Great system for gentlemen and ladies Is a web application about buying clothes Choosing the right clothes and according to the shape of each user Is a system that is an alternative for users to use the service To help in choosing clothes for users more easily The developer therefore aims to develop a system to introduce dress code for gentlemen and ladies. To be an alternative way to recommend dressing for users who are interested in dressing This system was developed using the Django Web Framework, using collaborative filtering techniques to perform the function of the clothing suggestion function for users. Website enabled on Pythonanywhere Which is open to use the SQLite3 database free of charge.

Keywords: Web application, Dressing system for gentlemen and ladies

สารบัญ

หน้า

กิตติกรรมประกาศ . . . . . ค

บทคัดย่อภาษาไทย . . . . . ง

บทคัดย่อภาษาอังกฤษ . . . . . จ

สารบัญ . . . . . ฉ

สารบัญตาราง . . . . . ญ

สารบัญภาพ . . . . . ฏ

บทที่

1 บทนำ 1

1.1 ที่มาและเหตุผล 1

1.2 วัตถุประสงค์ 1

1.3 ขอบเขตของโครงงาน 2

1.4 ประโยชน์ที่คาดว่าจะได้รับ 3

1.5 เครื่องมือที่ใช้ในการพัฒนา (Development tools) 3

1.5.1 ฮาร์ดแวร์ 3

1.5.2 ซอฟต์แวร์ (Software) 3

1.5.3 แผนการดำเนินการ 5

2 ทฤษฎีที่เกี่ยวข้อง 6

2.1 ความรู้เกี่ยวกับ Visual Studio Code 6

2.2 ความรู้พื้นฐาน Java 6

2.3 ความรู้เกี่ยวกับ Django 6

2.3.1 คุณสมบัติของ Django Framework [1] 6

2.4 ความรู้เกี่ยวกับ SQLite 7

2.5 ความรู้เกี่ยวกับภาษา Python 9

2.6 ความรู้เกี่ยวกับ Collaborative filtering 10

2.6.1 ระบบแนะนำ (Recommender System) 10

2.6.2 Collaborative Filtering Technique 15

2.6.3 การทดสอบประสิทธิภาพของเทคนิคที่ใช้ 16

2.7 เอกสารและงานวิจัยที่เกี่ยวข้อง 16
2.7.1 เว็บแอปพลิเคชัน Wongnai 16
2.7.2 ข้อแตกต่างระหว่างเว็บแอปพลิเคชัน wongnai กับเว็บของโครงงาน 16
3 การวิเคราะห์และออกแบบระบบ 18
3.1 โครงสร้างภาพรวมของระบบ 18
3.2 System Requirements 19
3.2.1 Functional Requirements 19
3.2.2 Non-functional Requirements 20
3.3 User Interface Design 20
3.3.1 ส่วนของผู้ใช้งาน 21
3.3.2 ส่วนของร้านค้า 31
3.3.3 ส่วนของผู้ดูแลระบบ 35
3.4 Use Case Diagram 37
3.5 Class Diagram 48
3.6 Sequence Diagram 53
3.6.1 Sequence Diagram ของการสมัครสมาชิก 54
3.6.2 Sequence Diagram ของการเข้าสู่ระบบ 55
3.6.3 Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน 56
3.6.4 Sequence Diagram ของบันทึกรายการเสื้อผ้าที่ถูกใจ 57
3.6.5 Sequence Diagram ของบันทึกรายการร้านค้าที่ถูกใจ 58
3.6.6 Sequence Diagram ของดูบันทึกรายการเสื้อผ้าที่ถูกใจ 59
3.6.7 Sequence Diagram ของดูบันทึกรายการร้านค้าที่ถูกใจ 60
3.6.8 Sequence Diagram ของดูรายการเสื้อผ้าจากร้านค้า 61
3.6.9 Sequence Diagram ของดูรายการร้านค้า 62
3.6.10 Sequence Diagram ของค้นหาเสื้อผ้า 63
3.6.11 Sequence Diagram ของการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ 64
3.6.12 Sequence Diagram ของการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ 65
3.6.13 Sequence Diagram ของเพิ่มร้านค้า 66
3.6.14 Sequence Diagram ของแก้ไขข้อมูลร้านค้า 67

3.6.15 Sequence Diagram ของเพิ่มเสื้อผ้าในร้านค้า . . . 68
3.6.16 Sequence Diagram ของลบเสื้อผ้าในร้านค้า . . . 69
3.6.17 Sequence Diagram ของแก้ไขข้อมูลเสื้อผ้าในร้านค้า . . . 70
3.6.18 Sequence Diagram ของดูรายการคนที่ถูกใจเสื้อผ้า . . . 71
3.6.19 Sequence Diagram ของดูรายการคนที่ถูกใจร้านค้า . . . 72
3.6.20 Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน . . . 73
3.6.21 Sequence Diagram ของลบผู้ใช้งาน . . . 74
3.6.22 Sequence Diagram ของแก้ไขข้อมูลร้านค้า . . . 75
3.6.23 Sequence Diagram ของลบร้านค้า . . . 76
3.6.24 Sequence Diagram ของฝึกแบบจำลองสำหรับระบบแนะนำ . . . 77
3.7 โครงสร้างฐานข้อมูล SQLite3 . . . 78
4 การพัฒนาระบบ . . . 84
4.1 การพัฒนาในส่วนการสมัครสมาชิกหรือเพิ่มข้อมูล . . . 84
4.2 การพัฒนาในส่วนการเข้าสู่ระบบ . . . 86
4.3 การพัฒนาในส่วนการแสดงข้อมูลทั้งหมด . . . 87
4.4 การพัฒนาในส่วนการแสดงรายละเอียดของข้อมูล . . . 89
4.5 การพัฒนาในส่วนการแก้ไขข้อมูล . . . 90
4.6 การพัฒนาในส่วนการลบข้อมูล . . . 92
4.7 การพัฒนาในส่วนการเพิ่มเป็นสินค้าโปรด . . . 93
4.8 การพัฒนาในส่วนการดูรายการคนที่ถูกใจสินค้า . . . 94
4.9 การพัฒนาในส่วนแนะนำสินค้า . . . 95
4.9.1 การฝึกต้นแบบของระบบผู้แนะนำ . . . 95
4.9.2 การนำต้นแบบมาแนะนำเสื้อผ้า . . . 99
5 การทดสอบระบบ . . . 101
5.1 การทดสอบในส่วนฟังก์ชันก์ของระบบ . . . 101
5.1.1 ผลการทดสอบการสมัครสมาชิกผู้ใช้งาน . . . 101
5.1.2 ผลการทดสอบการเข้าสู่ระบบผู้ใช้งาน . . . 102
5.1.3 ผลการทดสอบหน้าเสื้อผ้าทั้งหมด . . . 102
5.1.4 ผลการทดสอบหน้ารายละเอียดสินค้า . . . 103

5.1.5 ผลการทดสอบหน้าร้านค้าทั้งหมด . . . 103
5.1.6 ผลการทดสอบหน้ารายละเอียดร้านค้า . . . 103
5.1.7 ผลการทดสอบหน้าข้อมูลผู้ใช้งาน . . . 104
5.1.8 ผลการทดสอบหน้าสมัครสมาชิกร้านค้า . . . 104
5.1.9 ผลการทดสอบหน้าข้อมูลร้านค้า . . . 105
5.1.10 ผลการทดสอบหน้าแนะนำเสื้อผ้า . . . 106
5.2 การทดสอบประสิทธิภาพของโมเดล . . . 107
5.2.1 ผลการทดสอบประสิทธิภาพโดยใช้ RMSE และ MAE . . . 107
5.2.2 ผลการทดสอบความถูกต้องของโมเดลโดยวัดจากประเมินของ Rating . . . 108
6 สรุปและข้อเสนอแนะ . . . 109
6.1 สรุปความสามารถของระบบ . . . 109
6.2 ปัญหาและอุปสรรคในการพัฒนา . . . 110
6.3 แนวทางการพัฒนาต่อ . . . 110

บรรณานุกรม . . . 111

ภาคผนวก . . . 113
ภาคผนวก ก การติดตั้งเครื่องมือที่ใช้พัฒนาโปรแกรม . . . 113
ก.1 ขั้นตอนการติดตั้งโปรแกรม Visual Studio Code หรือ VS Code . . . 113
ก.2 การเริ่มต้นใช้ PythonAnywhere . . . 119
ภาคผนวก ข คู่มือการติดตั้งระบบ . . . 125
ภาคผนวก ค คู่มือการใช้งานระบบ . . . 127
ค.1 ส่วนผู้ใช้งาน . . . 127
ค.2 ส่วนร้านค้า . . . 139
ค.3 ส่วนผู้ดูแลระบบ . . . 150

ประวัติผู้พัฒนา . . . 154

สารบัญตาราง

ตารางที่ หน้า

1.1 ขั้นตอนการดำเนินงาน . . . 5
3.1 สัญลักษณ์ของ Use case Diagram . . . 37
3.2 อธิบาย Use Case หน้าที่ของระบบ ในภาพที่ 3.18 . . . 39
3.3 อธิบาย Use Case หน้าที่ของระบบ (ต่อ) ในภาพที่ 3.18 . . . 40
3.4 Use Case สมัครสมาชิก . . . 40
3.5 Use Case เข้าสู่ระบบ . . . 41
3.6 Use Case แก้ไขข้อมูลใช้งาน . . . 41
3.7 Use Case ดูรายการเสื้อผ้าจากร้านค้า . . . 41
3.8 Use Case ดูรายการร้านค้า . . . 42
3.9 Use Case ดูรายการแนะนำเสื้อผ้า . . . 42
3.10 Use Case ดูรายการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ . . . 42
3.11 Use Case ดูรายการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ . . . 43
3.12 Use Case บันทึกรายการเสื้อผ้าที่ถูกใจ . . . 43
3.13 Use Case บันทึกรายการร้านค้าที่ถูกใจ . . . 43
3.14 Use Case ดูบันทึกรายการเสื้อผ้าที่ถูกใจ . . . 44
3.15 Use Case ดูบันทึกรายการร้านค้าที่ถูกใจ . . . 44
3.16 Use Case ค้นหาเสื้อผ้า . . . 44
3.17 Use Case เพิ่มร้านค้า . . . 45
3.18 Use Case แก้ไขข้อมูลร้านค้า . . . 45
3.19 Use Case เพิ่ม ลบ แก้ไขข้อมูลเสื้อผ้าในร้านค้า . . . 45
3.20 Use Case ดูรายการการคนที่ถูกใจเสื้อผ้า . . . 46
3.21 Use Case ดูรายการการคนที่ถูกใจร้านค้า . . . 46
3.22 Use Case เพิ่ม ลบ แก้ไขข้อมูลร้านค้า . . . 46
3.23 Use Case เพิ่ม ลบ แก้ไขข้อมูลผู้ใช้งาน . . . 47
3.24 สัญลักษณ์ของ Class Diagram . . . 48
3.25 อธิบาย Class Diagram ของคลาสพื้นฐานของระบบ . . . 50
3.26 อธิบาย Class Diagram ของคลาสพื้นฐานของระบบ (ต่อ) . . . 51
3.27 อธิบาย Class Diagram ของคลาสพื้นฐานของระบบ (ต่อ) . . . 52
3.28 สัญลักษณ์ของ Sequence Diagram . . . 53
3.29 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Member . . . 79
3.30 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Gender . . . 79
3.31 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Shape . . . 79
3.32 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ SinColor . . . 79
3.33 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Store . . . 80
3.34 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Cloth . . . 81

3.35 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Category 81
3.36 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Color 81
3.37 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Size 82
3.38 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ FavoriteCloth 82
3.39 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ FavoriteStore 83
3.40 อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Recommend 83
5.1 ผลการทดสอบการสมัครสมาชิกผู้ใช้งาน 101
5.2 ผลการทดสอบการเข้าสู่ระบบผู้ใช้งาน 102
5.3 ผลการทดสอบหน้าเสื้อผ้าทั้งหมด 102
5.4 ผลการทดสอบหน้ารายละเอียดสินค้า 103
5.5 ผลการทดสอบหน้าร้านค้าทั้งหมด 103
5.6 ผลการทดสอบหน้ารายละเอียดร้านค้า 103
5.7 ผลการทดสอบหน้าข้อมูลผู้ใช้งาน 104
5.8 ผลการทดสอบหน้าสมัครสมาชิกร้านค้า 104
5.9 ผลการทดสอบหน้าข้อมูลร้านค้า 105
5.10 ผลการทดสอบหน้าแนะนำเสื้อผ้า 106
5.11 ผลลัพธ์ของการประเมินประสิทธิภาพด้วยวิธี K-fold crossvalidtion 107
5.12 ผลลัพธ์การทดสอบประสิทธิภาพของ 3 เทคนิคด้วยช่วงค่าและแปลงเป็นร้อยละ 108

สารบัญภาพ

รูปที่		หน้า
2.1	ระบบ CF / CB แบบรวม	14
2.2	Collaborative filtering แบบ User-Based Filtering และ แบบ Item-Based Filtering	15
2.3	หน้าแรกของเว็บไซต์ Wongnai	17
3.1	System architecture ระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี	19
3.2	หน้าสมัครสมาชิก	21
3.3	หน้าเข้าสู่ระบบ	22
3.4	หน้าจอเสื้อผ้าทั้งหมด	23
3.5	หน้าจอรายละเอียดเสื้อผ้า	24
3.6	หน้าจอร้านค้าทั้งหมด	25
3.7	หน้าจอรายละเอียดร้านค้า	26
3.8	หน้าจอข้อมูลผู้ใช้งาน	27
3.9	หน้าจอรายการสินค้าโปรด	28
3.10	หน้าจอรายการร้านโปรด	29
3.11	หน้าจอรายการแนะนำเสื้อผ้า	30
3.12	หน้าจอเพิ่มร้านค้า	31
3.13	หน้าจอข้อมูลร้านค้า	32
3.14	หน้าจอจัดการข้อมูลสินค้า	33
3.15	หน้าจอเพิ่มสินค้า	34
3.16	หน้าจอเข้าสู่ระบบผู้ดูแลระบบ	35
3.17	หน้าจอฐานข้อมูลทั้งหมดในระบบ	36
3.18	Use Case Diagram ระบบแนะนำการแต่งการสุภาพบุรุษและสุภาพสตรี	38
3.19	Class Diagram ระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี	49
3.20	Sequence Diagram ของการสมัครสมาชิก	54
3.21	Sequence Diagram ของการเข้าสู่ระบบ	55
3.22	Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน	56
3.23	Sequence Diagram ของบันทึกรายการเสื้อผ้าที่ถูกใจ	57
3.24	Sequence Diagram ของบันทึกรายการร้านค้าที่ถูกใจ	58
3.25	Sequence Diagram ของดูบันทึกรายการเสื้อผ้าที่ถูกใจ	59
3.26	Sequence Diagram ของดูบันทึกรายการร้านค้าที่ถูกใจ	60
3.27	Sequence Diagram ของดูรายการเสื้อผ้าจากร้านค้า	61
3.28	Sequence Diagram ของดูรายการร้านค้า	62
3.29	Sequence Diagram ของค้นหาเสื้อผ้า	63
3.30	Sequence Diagram ของการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ	64
3.31	Sequence Diagram ของการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ	65
3.32	Sequence Diagram ของเพิ่มร้านค้า	66
3.33	Sequence Diagram ของแก้ไขข้อมูลร้านค้า	67

3.34 Sequence Diagram ของเพิ่มเสื้อผ้าในร้านค้า . . . 68
3.35 Sequence Diagram ของลบเสื้อผ้าในร้านค้า . . . 69
3.36 Sequence Diagram ของแก้ไขข้อมูลเสื้อผ้าในร้านค้า . . . 70
3.37 Sequence Diagram ของดูรายการคนที่ถูกใจเสื้อผ้า . . . 71
3.38 Sequence Diagram ของดูรายการคนที่ถูกใจร้านค้า . . . 72
3.39 Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน . . . 73
3.40 Sequence Diagram ของลบผู้ใช้งาน . . . 74
3.41 Sequence Diagram ของแก้ไขข้อมูลร้านค้า . . . 75
3.42 Sequence Diagram ของลบร้านค้า . . . 76
3.43 Sequence Diagram ของฝึกแบบจำลองสำหรับระบบแนะนำ . . . 77
3.44 ตารางฐานข้อมูลสมาชิกหรือผู้ใช้งาน . . . 78
3.45 ตารางฐานข้อมูลร้านค้า . . . 80
3.46 ตารางฐานข้อมูลเสื้อผ้า . . . 80
3.47 ตารางฐานข้อมูลสินค้าโปรด . . . 82
3.48 ตารางฐานข้อมูลร้านโปรด . . . 82
3.49 ตารางฐานข้อมูลแนะนำเสื้อผ้า . . . 83
4.1 การทำงานของระบบเมื่อผู้ใช้สมัครสมาชิกหรือเพิ่มข้อมูล . . . 84
4.2 การทำงานของระบบเมื่อผู้ใช้สมัครสมาชิกหรือเพิ่มข้อมูล (ต่อ) . . . 85
4.3 การทำงานของระบบเมื่อผู้ใช้เข้าสู่ระบบ . . . 86
4.4 การทำงานของระบบเมื่อผู้ใช้กดดูรายการเสื้อผ้าทั้งหมด . . . 87
4.5 การทำงานของระบบเมื่อผู้ใช้กดดูรายการเสื้อผ้าทั้งหมด (ต่อ) . . . 88
4.6 การทำงานของระบบเมื่อผู้ใช้กดดูรายละเอียดสินค้า . . . 89
4.7 การทำงานของระบบเมื่อผู้ใช้กดแก้ไขสินค้า . . . 90
4.8 การทำงานของระบบเมื่อผู้ใช้กดแก้ไขสินค้า (ต่อ) . . . 91
4.9 การทำงานของระบบเมื่อผู้ใช้กดลบสินค้า . . . 92
4.10 การทำงานของระบบเมื่อผู้ใช้กดเพิ่มเป็นสินค้าโปรด . . . 93
4.11 การทำงานของระบบเมื่อผู้ใช้กดดูรายการคนที่ถูกใจสินค้า . . . 94
4.12 ภาพประกอบตารางการจำลองข้อมูลการให้ดาว ตัวอย่าง 10 แถว จาก 60 แถว . . . 95
4.13 ภาพประกอบไฟล์จากการ download แปลงข้อมูล favoritecloth จาก xlsx เป็น csv . . . 96
4.14 คำสั่งแปลง 11-column csv เป็น 3-column csv ที่สามารถอ่านด้วย pandas ได้ . . . 96
4.15 คำสั่งอ่าน 3-column csv เป็น dataframe -> dataset transcoder . . . 97
4.16 Algorithm name ที่ใช้ในการประมวณผลของระบบ ได้แก่ KNNBaseline(), KNNBasic(), KNNWithMeans() . . . 97
4.17 การวิเคราะห์ข้อมูลเพื่อนำไปใช้ในฟังก์ชันการแนะเสื้อผ้า . . . 97
4.18 การทำงานของระบบเมื่อผู้ใช้กดดูรายการแนะนำเสื้อผ้า . . . 99
ก.1 หน้าเว็บดาวน์โหลด Visual Studio Code . . . 113
ก.2 ไฟล์โปรแกรมที่โหลดมาได้ . . . 114
ก.3 หน้าต่างแจ้งเตือนก่อนการติดตั้งโปรแกรม . . . 114
ก.4 หน้าต่างข้อตกลงการใช้งาน Visual Studio Code . . . 115

ก.5 หน้าต่างที่จัดเก็บไฟล์ต่างๆ ของ Visual Studio Code . . . 115
ก.6 หน้าต่างการจัดการซอร์ดคัทของ Visual Studio Code . . . 116
ก.7 หน้าต่างเริ่มทำการติดตั้งทำการกดของ Visual Studio Code . . . 116
ก.8 หน้าต่างสุดท้ายการติดตั้งของ Visual Studio Code . . . 117
ก.9 หน้าต่างการติดตั้งของ Visual Studio Code เสร็จสิ้น . . . 118
ก.10 หน้าต่างเมื่อเขาโปรแกรมหลังติดตั้งเสร็จของ Visual Studio Code . . . 118
ก.11 หน้าแรกของ pythonanywhere ก่อนเข้าสู่ระบบ . . . 119
ก.12 หน้าแรกหลังจากทำการเข้าสู่ระบบเสร็จ . . . 120
ก.13 หน้าเริ่มการสร้างเว็บ . . . 120
ก.14 หน้าสร้าง web app . . . 121
ก.15 หน้าเงื่อนไขการ upgrade account . . . 121
ก.16 หน้าการสร้าง web . . . 122
ก.17 หน้าการเลือก version ของ python . . . 122
ก.18 หน้าการตั้งชื่อ project และเลือก directory เก็บ file directory . . . 123
ก.19 หน้าการสร้าง web app สำเร็จ . . . 123
ก.20 หน้าสามารใช้งาน web ได้ของ Django . . . 124
ก.21 หน้า Consoles . . . 124
ก.22 หน้าการเปิด Console ขึ้นมา . . . 124
ข.1 หน้าเว็บ Reposity ของโปรเจ็ค . . . 125
ข.2 คำสั่ง migrate สร้างตาราง . . . 125
ข.3 คำสั่ง migrate สร้างตาราง . . . 126
ข.4 คำสั่ง migrate สร้างตาราง . . . 126
ข.5 คำสั่ง migrate สร้างตาราง . . . 126
ค.1 หน้าจอหลักของระบบ . . . 127
ค.2 หน้าจอสมัครสมาชิก . . . 128
ค.3 หน้าจอเข้าสู่ระบบ . . . 129
ค.4 หน้าจอเสื้อผ้าทั้งหมด . . . 130
ค.5 หน้าจอรายละเอียดสินค้า . . . 131
ค.6 หน้าจอร้านค้าทั้งหมด . . . 132
ค.7 หน้าจอรายละเอียดร้านค้า . . . 133
ค.8 หน้าจอแนะนำเสื้อผ้า . . . 134
ค.9 หน้าจอข้อมูลผู้ใช้งาน . . . 135
ค.10 หน้าจอแก้ไขข้อมูลผู้ใช้งาน . . . 136
ค.11 หน้าจอรายการสินค้าโปรด . . . 137
ค.12 หน้าจอรายการร้านโปรด . . . 138
ค.13 หน้าจอหลักของระบบ . . . 139
ค.14 หน้าจอสมัครสมาชิก . . . 140
ค.15 หน้าจอเข้าสู่ระบบ . . . 141

ค.16 หน้าจอเพิ่มร้านค้า 142
ค.17 หน้าจอข้อมูลร้านค้า 143
ค.18 หน้าจอแก้ไขข้อมูลร้านค้า 144
ค.19 หน้าจอจัดการข้อมูลสินค้า 145
ค.20 หน้าจอเพิ่มสินค้า 146
ค.21 หน้าจอแก้ไขข้อมูลสินค้า 147
ค.22 หน้าจอรายการคนที่ถูกใจเสื้อผ้า 148
ค.23 หน้าจอรายการคนที่ถูกใจร้านค้า 149
ค.24 หน้าจอเข้าสู่ระบบสำหรับผู้ดูแลระบบในฐานข้อมูล 150
ค.25 หน้าจอหน้าตารางฐานข้อมูลทั้งหมด 151
ค.26 หน้าจอตารางข้อมูลผู้ใช้งานทั้งหมด 152
ค.27 หน้าจอแสดงข้อมูลผู้ใช้งาน ซึ่งผู้ดูแลระบบจะสามารถเพิ่ม ลบ และแก้ไขผู้ใช้งานได้ 152
ค.28 หน้าจอแสดงข้อมูลร้านค้าทั้งหมด 153
ค.29 หน้าจอแสดงข้อมูลร้านค้า ซึ่งผู้ดูแลระบบจะสามารถเพิ่ม ลบ และแก้ไขร้านค้าได้ 153

บทที่ 1

บทนำ

1.1 ที่มาและเหตุผล

เนื่องจากในปัจจุบันการแต่งกายถือเป็นเรื่องที่สำคัญ การใช้ชีวิตประจำวันในแต่ละวันที่ต้องไปทำงานหรือต้องไปเรียนทุกคนต่างต้องมีการแต่งกายเพื่อให้ตัวเองดูดีและมีความน่าเชื่อถือ เพราะการแต่งกายสามารถบ่งบอกได้ถึงบุคลิกภาพของผู้ใส่ได้ และการแต่งกายให้เข้ากับรูปร่างยังถือเป็นเรื่องที่ยาก เนื่องจากรูปร่างของแต่ละบุคคลนั้นมีความแตกต่างกัน ดังนั้นการเลือกใส่เสื้อผ้าที่ใส่แล้วทำให้เราดูดีขึ้น การเลือกใส่เสื้อผ้าที่เหมาะสมกับรูปร่างจะส่งผลให้เราดูมีบุคลิกภาพที่ดีขึ้นได้ เพราะสามารถเพิ่มความมั่นใจในการใช้ชีวิตประจำวันได้ ในการเลือกซื้อเสื้อผ้าในแต่ละครั้งบางท่านอาจใช้เวลาในการเลือกนาน เนื่องจากไม่ทราบว่าควรใส่เสื้อผ้าแบบไหนที่ใส่แล้วจะทำให้ดูดีขึ้น เสื้อผ้าที่ใส่แล้วจะไม่ทำให้ดูอวบหรือผอมเพิ่มมากขึ้น จึงทำให้เสียเวลาไปกับการเลือกซื้อเสื้อผ้าไปนาน ดังนั้นการแต่งกายจึงถือเป็นเรื่องที่มีความสำคัญเป็นอย่างมาก และปัจจุบันระบบขายสินค้าออนไลน์ได้รับความนิยมอย่างมากเช่น shopee, lazada เป็นต้น ทำให้ลูกค้าได้รับความสะดวกในการซื้อขายสินค้า แต่สินค้าประเภทเสื้อผ้าเครื่องแต่งกายนั้นกลับมีทำตลาดผ่านการขายผ่านสังคมเครือข่ายเช่น facebook, instagram โดยผู้ที่มีทำการ live เป็นปัจจัยที่สำคัญในการตัดสินใจเลือกซื้อของลูกค้าเนื่องจากการแนะนำ ให้ความรู้เรื่องการแต่งกายที่เหมาะสม และการแนะนำชุดตัวอย่างให้ลูกค้าได้พิจารณาเลือกตาม อย่างไรก็ดี ลูกค้าที่เลือกชุดแต่งกายอาจจะมีรูปร่าง ลักษณะ บุคลิกที่แตกต่างกันออกไป และอาจจะแตกต่างจากผู้ทำการ live ขายสินค้า ทำให้การสินค้าที่เสนอ ไม่ตรงตามความต้องการของผู้ใช้งาน

ในปัจจุบันการใช้อินเตอร์เน็ตเป็นปัจจัยหนึ่งในชีวิตของมนุษย์ไปแล้ว และมีการพัฒนาอย่างต่อเนื่องทั้งด้านฮาร์ดแวร์ (Hardware) และซอฟต์แวร์ (Software) เพื่อตอบสนองความตองการของผู้ใช้งานทั้งในด้านบวกและด้านลบ จึงทำให้ผู้พัฒนามีความคิดที่จะสร้างแอปพลิเคชันที่เป็นประโยชน์ให้แก่ผู้ใช้งาน

ดังนั้นผู้พัฒนาจึงได้สร้างเป็นเว็บที่จะมาช่วยให้แนะนำการแต่งกายให้กับสุภาพบุรุษและสุภาพสตรีทุกท่านที่มีปัญหาในเรื่องของการเลือกใส่เสื้อผ้า จะช่วยท่านในเรื่องของการเลือกใส่เสื้อผ้าแบบไหนที่จะเข้ากับรูปร่างและใส่แล้วเหมาะสมกับหุ่นของท่าน ระบบแนะนำการแต่งกายสุภาพบรุษและสุภาพสตรีนี้จะช่วยทำให้การแต่งกายเป็นเรื่องที่ง่ายและสนุกมากยิ่งขึ้น โดยใช้หลักการของ Collaborative Filtering เพื่อแนะนำเสื้อผ้าที่เหมาะสมจากหลายร้านค้าให้กับผู้ใช้งานโดยพิจารณาจากกลุ่มผู้ใช้ที่มีคุณลักษณะและประวัติการซื้อที่ใกล้เคียงกัน ข้อมูลที่จัดเก็บสามารถนำมาวิเคราะห์ข้อมูลการซื้อขายและการจัดกลุ่มลูกค้าเพื่อให้ร้านค้าที่เป็นสมาชิกสามารถนำไปวิเคราะห์เพื่อทำการส่งเสริมการขายของร้านต่อไป จุดเด่นคือสมาชิกสามารถเลือกเสื้อผ้าจากหลายร้านค้าในระบบเพื่อสร้างเป็นชุดที่ชื่นชอบ สามารถนำไปแสดงเป็นข้อความสาธารณะบนสังคมเครือข่าย โดยสมาชิกคนอื่นสามารถติดตามและให้เสนอข้อคิดเห็นได้

1.2 วัตถุประสงค์

1. เพื่อพัฒนาเว็บแนะนำการเลือกเสื้อผ้าโดยใช้หลักการเรียนรู้ของเครื่อง
2. เพื่อพัฒนาระบบเว็บเพื่อนำเสนอความรู้และให้คำแนะนำในการแต่งกายให้แก่สุภาพบุรุษและสุภาพสตรี

1.3 ขอบเขตของโครงงาน

1.3.1 ผู้ใช้งาน

- สามารถสมัครสมาชิกได้
- สามารถเข้าสู่เข้าสู่ระบบได้
- สามารถแก้ไขข้อมูลผู้ใช้งานได้
- สามารถดูรายการเสื้อผ้าจากร้านค้าได้
- สามารถดูรายการร้านค้าได้
- สามารถดูรายการแนะนำเสื้อผ้าได้
- สามารถบันทึกรายการเสื้อผ้าที่ถูกใจได้
- สามารถบันทึกรายการร้านค้าที่ถูกใจได้
- สามารถดูบันทึกรายการเสื้อผ้าที่ถูกใจได้
- สามารถดูบันทึกรายการร้านค้าที่ถูกใจได้
- สามารถดูค้นหาเสื้อผ้าได้

1.3.2 ร้านค้า

- สามารถสมัครสมาชิกได้
- สามารถเข้าสู่ระบบได้
- สามารถแก้ไขข้อมูลร้านค้าได้
- สามารถเพิ่ม ลบ แก้ไข ข้อมูลเสื้อผ้าในร้านค้าได้
- สามารถดูรายการคนที่ถูกใจเสื้อผ้าได้
- สามารถดูรายการคนที่ถูกใจร้านค้าได้

1.3.3 ผู้ดูแลระบบ

- สามารถเข้าสู่ระบบได้
- สามารถเพิ่ม ลบ แก้ไข ข้อมูลร้านค้าได้
- สามารถเพิ่ม ลบ แก้ไข ข้อมูลผู้ใช้งานได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. มีระบบแนะนำการแต่งกายสำหรับสุภาพบุรุษและสุภาพสตรี
2. มีระบบเว็บแนะนำในเรื่องของการแต่งกายให้เหมาะกับรูปร่าง เพื่อช่วยให้ลูกค้าสามารถเลือกดูสินค้าทุกเวลาที่สะดวก
3. มีระบบสามารถแนะนำสินค้าตามรสนิยมของผู้ใช้ได้

1.5 เครื่องมือที่ใช้ในการพัฒนา (Development tools)

1.5.1 ฮาร์ดแวร์

1. เครื่องคอมพิวเตอร์ส่วนบุคคล (Personal computer)
    - หน่วยประมวลผลกลาง CPU Intel ® CoreTM i3-50005U CPU @ 2.0GHz 3MB L3Cache
    - หน่วยความจำหลัก ขนาด 4 GB
    - หน่วยความจำสำรอง Solid State Drive (SSD) ขนาดความจุ 240 GB
    - หน่วยความจำแสดงผล Intel ® Graphics 5500
    - ระบบปฏิบัติการ: Windows 10 Education 64 bit

1.5.2 ซอฟต์แวร์ (Software)

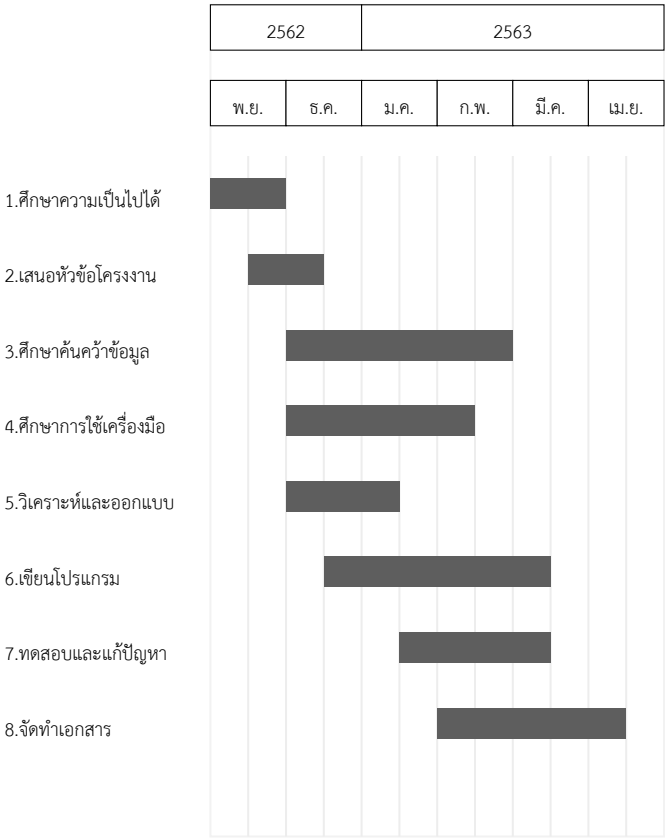
1. Visual Studio Code เป็น text editor เครื่องมือสำหรับพัฒนาเว็บแอปพลิเคชัน

2. PythonAnywhere คือ เว็บที่ให้บริการเครื่องมือที่ใช้ในการพัฒนาโปรแกรม (IDE) และ เป็น web hosting service โดยใช้ภาษา Python
3. Django คือ โปรแกรม OpenSource สำหรับพัฒนา web application ซึ่งเขียนในภาษา Python
4. SQLite เป็นโปรแกรมฐานข้อมูลที่มีขนาดเล็ก เก็บฐานข้อมูลเป็นไฟล์โดยไม่จำเป็นต้องมีเซิร์ฟเวอร์
5. Collaborative filtering คือระบบ RS ที่ใช้ข้อมูลการกระทำของเราและผู้ใช้งานท่านอื่น เพื่อใช้แนะนำการกระทำที่เราน่าจะทำหรือน่าสนใจ

1.5.3 แผนการดำเนินการ

ในการสร้างระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรีนี้ ผู้พัฒนาได้แบ่งขั้นตอน การดำเนินงานไว้ด้วยกัน 8 ขั้นตอน ดังตารางที่ 1.1

ตารางที่ 1.1: ขั้นตอนการดำเนินงาน



บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงรายละเอียดเกี่ยวกับทฤษฎีและงานวิจัยที่เกี่ยวของกับการพัฒนาระบบ แนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี โดยแบ่งเนื้อหาออกเป็น 2 ส่วน ได้แก่ ส่วนที่หนึ่ง เป็นเนื้อหาพื้นฐานเกี่ยวกับทฤษฎีการเขียนโปรแกรมและเทคโนโลยีที่นำมาใช้ในการพัฒนาในหัวข้อที่ 2.1-2.6 ได้แก่ ความรู้พื้นฐานเกี่ยวกับ Visual studio code, PythonAnywhere, Django, SQLite, Collaborative Filtering และภาษา Python และในส่วนที่สองเป็นเนื้อหาเกี่ยวกับเว็บที่เกี่ยวของกับโครงงานนี้ในหัวข้อที่ 2.7

2.1 ความรู้เกี่ยวกับ Visual Studio Code

วิชวล สตูดิโอโค้ด (Visual Studio Code) [2] เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไขและปรับแต่งโค้ด จากค่ายไมโครซอฟท์ (Microsoft) มีการพัฒนาออกมาในรูปแบบของ OpenSource จึงสามารถนำมาใช้งานได้แบบฟรีที่ต้องการความเป็นมืออาชีพ ซึ่ง Visual Studio Code นั้น เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานข้ามแพลตฟอร์ม (platform) รองรับการใช้งานทั้งบน Windows, macOS และ Linux สนับสนุนทั้งภาษา JavaScript, TypeScript และ Node.js สามารถเชื่อมต่อกับ Git ได้ นำมาใช้งานได้ง่ายไม่ซับซ้อนมีเครื่องมือส่วนขยายต่าง ๆ ให้เลือกใช้อย่างมาก การเปิดใช้งานภาษาอื่น ๆ ทั้ง ภาษา C++, C, Java, Python, PHP หรือ Go

2.2 ความรู้พื้นฐาน Java

PythonAnywhere [3] เป็นเว็บไซต์ที่ให้บริการ Online IDE และ Web Hosting Service โดยใช้ภาษา Python เป็นหลัก มี Python Web Framework ต่างๆ เช่น Django, Flask, Bottle สามารถสมัครสมาชิกได้ฟรี หรือสามารถจ่ายเงินเป็นรายเดือนเพื่อเพิ่มความสามารถต่างๆ ได้ โดยสมาชิกแบบฟรีสามารถ Deploy Web App ได้ 1 Web App (ไม่สามารถตั้งชื่อโดเมน Web App เองได้ โดยทางเว็บไซต์จะให้ใช้ ID เป็นชื่อโดเมนของ Web App)

2.3 ความรู้เกี่ยวกับ Django

แจงโก้ (Django) [4] เป็น framework ที่ใช้ในการสร้าง Web Application ในฝั่งของ Back End ที่พัฒนาด้วยภาษา Python โดยในตัว Framework จะมีส่วนประกอบทุกอย่างที่จำเป็นตั้งแต่การเชื่อมต่อฐานข้อมูล ไปจนถึงการ Render ข้อมูลออกมาให้ฝั่ง Front End แสดงผลข้อมูลเหล่านั้นได้ ซึ่ง Framework ในรูปแบบนี้ในภาษาอื่นๆ เช่น Ruby on rails สำหรับภาษา Ruby, Play Framework สำหรับภาษา Java หรือ Scala, Groovy on Grails สำหรับภาษา Groovy, Laravel สำหรับภาษา PHP หรือ Express สำหรับภาษา Javascript ของ Node.js เป็นต้น

2.3.1 คุณสมบัติของ Django Framework [1]

1. Object-Relational Mapper คือ การกำหนด Data Model ในภาษา Python เพื่อการทำงานด้านข้อมูล และสนับสนุน Dynamic Database-Access API

2. Automatic Admin Interface คือ ส่วนของการสร้าง Interface อัตโนมัติสำหรับการ Add, Edit, Delete และ Search ด้วย Django Framework
3. Elegant URL design คือ การทำให้ URL มีความสวยงาม สั้น กระชับ และสื่อความหมายของหน้านั้น ๆ ได้อย่างชัดเจน เหมาะสมกับการทำ SEO ในปัจจุบัน
4. Template System คือ Django นั้นมีการออกแบบ Template Language เพื่อการเขียนแยกส่วนระหว่าง Design และ Business Logic
5. Cache System คือ ส่วนของการบันทึก หรือจัดการข้อมูลที่มีการดาวน์โหลดไปแล้ว เพื่อเพิ่มประสิทธิภาพการทำงานของเว็บไซต์ด้านความเร็ว และด้านอื่น ๆ
6. nternationalization คือ Django สนับสนุน Application ที่มีความหลากหลายด้านภาษาในการแสดงผล

2.4 ความรู้เกี่ยวกับ SQLite

เอสคิวแอลไลท์ (SQLite) [5] เป็นห้องสมุดที่น่าทึ่งที่ฝังตัวอยู่ภายในแอปพลิเคชันที่ใช้งาน ในฐานะที่เป็นฐานข้อมูลแบบไฟล์ในตัว SQLite มีชุดเครื่องมือที่ยอดเยี่ยมในการจัดการข้อมูลทุกประเภทโดยมีข้อ จำกัด และความสะดวกน้อยกว่ามากเมื่อเปรียบเทียบกับฐานข้อมูลเชิงสัมพันธ์แบบโฮสต์กระบวนการ (เซิร์ฟเวอร์)

เมื่อแอปพลิเคชันใช้ SQLite กาพรวมจะทำงานกับการเรียกใช้งานและการโทรโดยตรงกับไฟล์ที่เก็บข้อมูล (เช่นฐานข้อมูล SQLite) แทนการสื่อสารผ่านอินเทอร์เฟซประเภท (พอร์ตพอร์ตซ็อกเก็ต) สิ่งนี้ทำให้ SQLite รวดเร็วและมีประสิทธิภาพเป็นอย่างยิ่งและยังมีเทคโนโลยีพื้นฐาน

ประเภทข้อมูลที่สนับสนุนของ SQLite

- โมฆะ:

ค่า NULL

- จำนวนเต็ม:

เลขจำนวนเต็มที่ลงนามซึ่งเก็บใน 1, 2, 3, 4, 6 หรือ 8 ไบต์ขึ้นอยู่กับขนาดของค่า

- จริง:

ค่าจุดลอยตัวเก็บไว้เป็นหมายเลขทศนิยม IEEE 8 ไบต์

- ข้อความ:

สตริงข้อความที่จัดเก็บโดยใช้การเข้ารหัสฐานข้อมูล (UTF-8, UTF-16BE หรือ UTF-16LE)

- หยด:

หยดของข้อมูลที่จัดเก็บเหมือนกับข้อมูลที่ป้อนเข้า

หมายเหตุ: หากต้องการเรียนรู้เพิ่มเติมเกี่ยวกับประเภทข้อมูลของ SQLite และความสัมพันธ์ประเภท SQLite ให้ตรวจสอบเอกสารอย่างเป็นทางการเกี่ยวกับเรื่องนี้

ข้อดีของ SQLite

- ไฟล์ที่ใช้:

ฐานข้อมูลทั้งหมดประกอบด้วยไฟล์เดียวบนดิสก์ซึ่งทำให้พกพาได้อย่างมาก

- มาตรฐานการตระหนักถึง:

แม้ว่ามันอาจดูเหมือนการใช้งานฐานข้อมูล“ แบบง่าย” แต่ SQLite ก็ใช้ SQL มันมีคุณสมบัติบางอย่างที่ถูกละเว้น (เข้าร่วมขวาหรือเพื่อแต่ละงบ) แต่บางส่วนเพิ่มเติมจะถูกอบเข้า

- เหมาะสำหรับการพัฒนาและการทดสอบ:

ในระหว่างขั้นตอนการพัฒนาแอปพลิเคชั่นส่วนใหญ่สำหรับคนส่วนใหญ่มีแนวโน้มที่จะต้องการโซลูชันที่สามารถปรับขนาดสำหรับการทำงานพร้อมกัน SQLite ซึ่งมีฐานคุณลักษณะที่หลากหลายสามารถให้มากกว่าสิ่งที่จำเป็นสำหรับการพัฒนาด้วยความเรียบง่ายในการทำงานกับไฟล์เดียวและไลบรารีบนฐาน C ที่เชื่อมโยง

ข้อเสียของ SQLite

- ไม่มีการจัดการผู้ใช้:

ฐานข้อมูลขั้นสูงมาพร้อมกับการสนับสนุนสำหรับผู้ใช้เช่นการเชื่อมต่อที่มีการจัดการที่มีสิทธิ์ตั้งค่าการเข้าถึงฐานข้อมูลและตาราง เมื่อพิจารณาถึงวัตถุประสงค์และลักษณะของ SQLite (ไม่มีระดับที่สูงขึ้นของการทำงานพร้อมกันของไคลเอนต์หลายรายการ) คุณลักษณะนี้ไม่มีอยู่

- ขาดความเป็นไปได้ที่จะจัดด้วยสำหรับประสิทธิภาพเพิ่มเติม

อีกครั้งโดยการออกแบบ SQLite เป็นไปไม่ได้ที่จะได้รับประสิทธิภาพที่เพิ่มขึ้นอย่างมาก ไลบรารี่นั้นง่ายต่อการปรับแต่งและใช้งานง่าย เนื่องจากมันไม่ซับซ้อนจึงเป็นไปไม่ได้ในทางเทคนิคที่จะทำให้มันมีประสิทธิภาพมากกว่ามันแล้วน่าประหลาดใจคือ

เมื่อต้องการใช้ SQLite

- แอปพลิเคชันแบบฝัง:

แอปพลิเคชันทั้งหมดที่ต้องการความสะดวกในการพกพาซึ่งไม่จำเป็นต้องมีการขยายเช่น แอพพลิเคชั่นในท้องถิ่นแอปพลิเคชันมือถือหรือเกม

- การแทนที่การเข้าถึงดิสก์:

ในหลายกรณีแอปพลิเคชันที่จำเป็นต้องอ่าน / เขียนไฟล์ไปยังดิสก์โดยตรงจะได้ประโยชน์จากการสลับไปใช้ SQLite สำหรับการทำงานเพิ่มเติมและความเรียบง่ายที่มาจากการใช้ภาษาที่มีโครงสร้างของแบบสอบถาม (SQL)

- การทดสอบ:

มันเกินความเป็นจริงสำหรับแอปพลิเคชันขนาดใหญ่เพื่อใช้กระบวนการเพิ่มเติมสำหรับการทดสอบตรรกะทางธุรกิจ (เช่นวัตถุประสงค์หลักของแอปพลิเคชัน:

ฟังก์ชันการทำงาน)

เมื่อไม่ต้องการใช้ SQLite

- แอปพลิเคชันผู้ใช้หลายคน:

หากคุณกำลังทำงานกับแอปพลิเคชันที่ลูกค้าหลายรายต้องการเข้าถึงและใช้ฐานข้อมูลเดียวกัน RDBM ที่มีคุณสมบัติครบถ้วน (เช่น MySQL) น่าจะเลือก SQLite ได้ดีกว่า

- แอปพลิเคชันที่ต้องการปริมาณการเขียนสูง:

หนึ่งในข้อ จำกัด ของ SQLite คือการดำเนินการเขียน DBMS นี้อนุญาตให้ใช้งานการเขียน * เพียงครั้งเดียวในเวลาที่กำหนดดังนั้นจึงอนุญาตให้มีปริมาณงานที่จำกัด

2.5 ความรู้เกี่ยวกับภาษา Python

ภาษาโปรแกรม Python [6] คือภาษาโปรแกรมคอมพิวเตอร์ระดับสูง โดยถูกออกแบบมาให้เป็นภาษาสคริปต์ที่อ่านง่าย โดยตัดความซับซ้อนของโครงสร้างและไวยกรณ์ของภาษาออกไป ในส่วนของการแปลงชุดคำสั่งที่เราเขียนให้เป็นภาษาเครื่อง Python มีการทำงานแบบ Interpreter คือเป็นการแปลชุดคำสั่งทีละบรรทัด เพื่อป้อนเข้าสู่หน่วยประมวลผลให้คอมพิวเตอร์ทำงานตามที่เราต้องการ นอกจากนั้นภาษาโปรแกรม Python ยังสามารถนำไปใช้ในการเขียนโปรแกรมได้หลากหลายประเภท โดยไม่ได้จำกัดอยู่ที่งานเฉพาะทางใดทางหนึ่ง (General-purpose language) จึงทำให้มีการนำไปใช้กันแพร่หลายในหลายองค์กรใหญ่ระดับโลก เช่น Google, YouTube, Instagram, Dropbox และ NASA เป็นต้น

ประวัติของภาษาโปรแกรม Python ได้เริ่มต้นขึ้นในเดือนธันวาคมปี 1989 โดยนาย Guido van Rossum โปรแกรมเมอร์ชาวดัตช์ ในตอนนั้นทำงานอยู่ที่สถาบันวิจัยแห่งชาติ Centrum Wiskunde Informatica (CWI) ซึ่งเป็นสถาบันวิจัยทางด้านคณิตศาสตร์และวิทยาการคอมพิวเตอร์ในเมืองอัมสเตอร์ดัม ประเทศเนเธอร์แลนด์ ในเวลานั้น Guido ต้องพัฒนาโปรแกรมสำหรับผู้ดูแลระบบ เพื่อใช้ในโครงการ Amoeba ซึ่งเป็นโครงการเกี่ยวกับระบบปฏิบัติการแบบกระจาย (Distributed operating system) อย่างไรก็ตามเขารู้สึกว่าภาษาโปรแกรม ABC, C และ Bourne shell มีข้อจำกัดมากมาย ทั้งเรื่องใช้เวลาในการพัฒนานานมากและไม่สามารถตอบโจทย์หลายประการ ดังนั้น Guido จึงได้ตัดสินใจเริ่มพัฒนาภาษาโปรแกรมระดับสูงขึ้นมาใหม่เพื่อใช้งานเองเป็นงานอดิเรก โดยนำเอาสิ่งที่ชอบในภาษา ABC มาพัฒนาลงไปในภาษาโปรแกรม Python รวมถึงได้พัฒนาส่วนอื่น ๆ เพิ่มเติมเข้าไป และในเวลาต่อมาจึงได้เผยแพร่ Python 1.0 เวอร์ชันแรกในปี 1994 หากเทียบกับภาษา Java ที่ได้ทำการเผยแพร่เวอร์ชันแรกในปี 1996 จะเห็นได้ว่าภาษา Python มีอายุมากกว่าภาษา Java ถึง 2 ปีที่มาของชื่อภาษาโปรแกรม Python นั้นไม่ได้มีที่มาเกี่ยวข้องกับงูเหมือนกับชื่อของมันแต่อย่างใด แต่ในช่วงที่ตัดสินใจเลือกชื่อนั้น ชื่อแรกที่เข้ามาในความคิดของ Guido ก็คือ มอนตี้ ไพธอน: ละครสัตว์เหินหาว (Monty Python's Flying Circus) ซึ่งเป็นชื่อรายการโทรทัศน์ทางช่อง BBC แนวตลกชื่อดังจากฝั่งอังกฤษที่เขาชื่นชอบมาก ๆ โดยเขาให้เหตุผลว่า "Python" เป็นชื่อที่สั้น จำได้ง่าย ฉีกแนวนิดๆ และดูลึกลับ ในตอนนั้นโดยทั่วไปมักจะนิยมเอาชื่อของบุคคลที่มีชื่อเสียงมาใช้เป็นชื่อภาษาโปรแกรมคอมพิวเตอร์ เช่น Ada, Pascal และ Eiffel ถึงแม้ว่าทีมนักแสดงในรายการจะไม่ได้มีชื่อเสียงทางด้านวิทยาศาสตร์และเทคโนโลยี แต่ก็เป็นที่ชื่นชอบในกลุ่มชาว Geek อย่างมาก รวมถึงกลุ่มคนที่ทำงานใน CWI ก็มักจะนิยมเอาชื่อรายการทีวีโชว์มาตั้งชื่อในงานของตัวเองอีกด้วย นี่คือเหตุผลที่มาที่ไปของชื่อภาษา Python นอกจากนั้น Guido

ยังใช้ชื่อของนักแสดงตลกชาวอังกฤษชื่อดังและเป็นหนึ่งในสมาชิกผู้ก่อตั้งทีม Monty Python ที่ชื่อ Eric Idle มาใช้เป็นชื่อ IDE หรือเครื่องมือที่ใช้ในการพัฒนาโปรแกรมว่า “IDLE” อีกด้วย

2.6 ความรู้เกี่ยวกับ Collaborative filtering

2.6.1 ระบบแนะนำ (Recommender System)

ระบบผู้แนะนำ (Recommender System) หมายถึง การเสนอแนวทางเลือกให้แก่บุคคลใดบุคคลหนึ่ง ยกตัวอย่างสิ่งที่สามารถแนะนำ เช่น เสื้อผ้า ร้านอาหาร เพลง หนัง เป็นต้น ซึ่งในที่นี้จะเป็นระบบการแนะนำเสื้อผ้าให้แก่ผู้ใช้งาน โดยจะใช้หลักการของ Collaborative Filtering จะเลือกแนะนำสิ่งที่ผู้ใช้คนอื่นๆ ในระบบที่มีลักษณะความชอบคล้ายคลึงกับผู้ใช้ปัจจุบันมาก และผู้ใช้เหล่านี้ได้ให้คะแนนความชอบต่อไอเทม (item) นั้นไว้ค่อนข้างสูง ในขณะที่ผู้ใช้ปัจจุบันยังไม่เคยใช้ไอเทมนั้นมาก่อน

ระบบผู้แนะนำสามารถแบ่งออกได้เป็น 5 ประเภทตามลำดับความซับซ้อนของเทคนิคที่ใช้ได้แก่ [7]

1. ไอเท็มยอดนิยม - กลยุทธ์ที่ง่ายที่สุด (Most Popular Items - The Simplest Strategy)

   วิธีที่ง่ายที่สุดคือให้ลูกค้าสิ่งที่ได้รับความนิยมมากที่สุดไม่ว่าจะเป็นภาพยนตร์หนังสือหรือบทความเกี่ยวกับเสื้อผ้า โดยไม่ต้องทำอะไรมากกว่าดูในบันทึกการขายของคุณคุณสามารถทำได้ ไม่ต้องใช้วิทยาศาสตร์ข้อมูล ข้อมูลไม่ได้เป็นแบบส่วนบุคคล แต่อาจมีประโยชน์หากคุณรู้น้อยมากเกี่ยวกับผู้เข้าชม ต้องมีคุณสมบัติเนื้อหาพื้นฐานเพื่อสร้างหมวดหมู่ย่อยที่สามารถจับคู่การเรียกดูปัจจุบันของผู้เข้าชม ตัวอย่างเช่นหากคุณเสนอสินค้าที่หลากหลายเช่นทุกอย่างตั้งแต่เครื่องมือไปจนถึงเสื้อผ้าหรือภาพยนตร์หนังสือหรือข่าวที่ดึงดูดความสนใจ จะต้องพยายามจับคู่รายการอย่างน้อยในหมวดหมู่เดียวกัน แม้จะมีข้อจำกัดเหล่านี้ Home Depot ใช้ผู้ขายที่ดีที่สุดอย่างสม่ำเสมอและ GAP ใช้ผลิตภัณฑ์ล่าสุด และมาถึงเร็วนี้เป็นประจำเพื่อเพิ่มรายได้ เป็นไปได้มากที่จะพิจารณาผู้แนะนำที่นิยมมากที่สุดว่าเป็นวิธีเสริม
2. การวิเคราะห์ความสัมพันธ์หรือการวิเคราะห์ตลาด (Association or Market Basket Analysis)

   การวิเคราะห์ความสัมพันธ์และการวิเคราะห์ตลาดมองเฉพาะเนื้อหา การวิเคราะห์ทางสถิติประเภทนี้ขึ้นอยู่กับการคำนวณที่ง่ายที่สุดเท่านั้นในการค้นหารายการที่ใช้ร่วมกันบ่อยครั้ง การวิเคราะห์ความสัมพันธ์และการวิเคราะห์ตลาดมีความเหมือนกันทางคณิตศาสตร์ เมื่อลูกค้ามักจะได้รับรายการหรือบริการหนึ่งครั้ง (เช่นบริการธนาคาร) เรียกสมาคมนี้ว่า เมื่อลูกค้าอาจซื้อหลายอย่างพร้อมกันเรียก Market Basket ดังนั้นการวิเคราะห์ความสัมพันธ์จะดำเนินการในระดับลูกค้า (สิ่งที่อยู่ในบัญชีของลูกค้า) ในขณะที่การวิเคราะห์ตลาดจะดำเนินการในระดับการทำธุรกรรม (สิ่งที่อยู่ในรายการของลูกค้า) มี 3 ขั้นตอนหลัก:

   1 ประเมินความสัมพันธ์ระหว่างแต่ละผลิตภัณฑ์ของสินค้าและผลิตภัณฑ์อื่นที่ได้เสนอโดยใช้อัลกอริทึมของคณิตศาสตร์

   2 ระบุการจับคู่ที่มีความสัมพันธ์ที่ดี (โดยทั่วไปจะเป็นคะแนนความสัมพันธ์ 2 หรือสูงกว่า) ตัวอย่างเช่นลูกค้าที่มีบัตรเครดิตอาจพบว่ามีแนวโน้มที่จะมีสินเชื่อรถยนต์มากกว่าสองเท่าหรือสามเท่ากว่าลูกค้าสินเชื่อรถยนต์ที่สุ่มเลือก

   3 สร้างข้อเสนอส่วนบุคคลสำหรับลูกค้าที่มีผลิตภัณฑ์หนึ่งของคู่ที่เชื่อมโยงอย่างยิ่ง แต่ไม่ใช่อีกข้อ

ข้อดีสำหรับเทคนิคนี้

- ง่ายและรวดเร็วมาก
- จะทำงานกับฐานลูกค้าขนาดเล็กและกระจัดกระจาย ตัวอย่างเช่นหากเป็นผู้ให้บริการขนาดเล็กเช่นธนาคารระดับภูมิภาคที่มีลูกค้าน้อยกว่า 5,000 รายและอาจมีเพียง 30 ผลิตภัณฑ์หรือผู้ให้บริการอีคอมเมิร์ซบูติกขนาดเล็กในทำนองเดียวกัน
- ความรู้เกี่ยวกับลูกค้า นอกเหนือจากผลิตภัณฑ์หรือบริการที่มีอยู่ในปัจจุบันไม่จำเป็นต้องมี การเตรียมข้อมูล เนื่องจากการเพิ่มความรู้เกี่ยวกับลูกค้าอาจใช้เวลานานและมีค่าใช้จ่ายสูงและไม่จำเป็นที่นี่การวิเคราะห์ความสัมพันธ์ของกลุ่มลูกค้าและตลาดจึงเป็นวิธีที่คุ้มค่าที่สุดสำหรับการสร้างข้อเสนอส่วนบุคคล

การวิเคราะห์ลุ่มลูกค้าและการวิเคราะห์ตลาดเป็นหัวใจสำคัญของคำแนะนำอีคอมเมิร์ซภายใต้หัวข้อ“ ลูกค้าที่ซื้อสิ่งนี้และถือว่าเป็นรายการเหล่านี้” หรือ“ รายการที่ซื้อด้วยกัน” ซึ่งเป็นวัตถุดิบหลักของ Amazon เนื่องจากต้องการข้อมูลลูกค้าน้อยมากจึงไม่ได้ผลเช่นกันเพราะการเลือกนั้นกว้างมากเช่นภาพยนตร์หนังสือหรือเพลงและมีคนรู้น้อยเกี่ยวกับสิ่งที่ลูกค้าชอบ อย่างไรก็ตามสามารถกรองโดยใช้กิจกรรมการเรียกดูปัจจุบันของผู้เข้าชม

3. การกรองเนื้อหา (Content Filtering :CF)

การกรองตามเนื้อหาเป็นสิ่งดีเมื่อ 10 ปีที่แล้ว ยังคงพบในการใช้งานที่หลากหลายและมีแอปพลิเคชันที่ใช้ได้ เนื่องจากชื่อแสดงถึง CF จะมองหาความคล้ายคลึงกันระหว่างรายการที่ลูกค้าบริโภคหรือเรียกดูในอดีตเพื่อนำเสนอทางเลือกในอนาคต CFs เป็นตัวแยกประเภทเฉพาะของผู้ใช้ที่เรียนรู้ที่จะจัดประเภททางเลือกในเชิงบวกหรือเชิงลบโดยยึดตามความชอบของผู้ใช้ (โปรไฟล์ผู้ใช้)

อัลกอริทึม:

ระบบจะสร้างโปรไฟล์ตามเนื้อหาของผู้ใช้โดยใช้แอตทริบิวต์ที่ไม่ต่อเนื่อง ประวัติความเป็นมาของการบริโภคหรือการเรียกดูของผู้ใช้ถูกใช้เพื่อสร้างเวกเตอร์ถ่วงน้ำหนักของคุณสมบัติของรายการ น้ำหนักจะเรียนรู้หรือกำหนดให้แตกต่างกันไปตามความสำคัญของคุณลักษณะสำหรับผู้ใช้เฉพาะ น้ำหนักนั้นถูกใช้เพื่อเปรียบเทียบกับน้ำหนักเวกเตอร์ของรายการต่าง ๆ ที่อาจแนะนำ เทคนิคการคำนวณอาจแตกต่างจากค่าเฉลี่ยถ่วงน้ำหนักอย่างง่ายไปจนถึงตัวจำแนกเบส์การวิเคราะห์กลุ่มต้นไม้ตัดสินใจหรือวิธีการที่ซับซ้อนมากขึ้นรวมถึงตาข่ายประสาทเทียม คุณจะต้องตรวจสอบวิธีแก้ปัญหาแบบแพคเกจเพื่อประเมินวิธีการคำนวณและประสิทธิผล

คุณสมบัติของรายการ:

ข้อกำหนดที่ชัดเจนคือสามารถอธิบายเนื้อหาจำนวนมากเพื่อใช้ในการจำแนกประเภท สิ่งเหล่านี้สามารถเป็นบูลีนได้ (ภาพยนตร์เป็นภาพเคลื่อนไหวผู้แต่งหนังสือเล่มนี้คือ Clive Cussler วัสดุของเสื้อเชิ้ตเป็นผ้าฝ้าย พวกเขายังสามารถต่อเนื่องเช่นการจัดอันดับภาพยนตร์ที่ได้รับจากแหล่งการจัดอันดับ“การจัดอันดับเฉลี่ย” ของลูกค้ารายอื่นที่บริโภครายการหรือร้อยละหรือจำนวนนาทีในภาพยนตร์ที่พิจารณาว่าเป็น “การกระทำ” ความสามารถในการรับและรักษาแอตทริบิวต์ของเนื้อหาเป็นทั้งเกณฑ์หลักและข้อจำกัดที่สำคัญของ CF คุณลักษณะบางอย่างอาจหาได้ง่าย แต่คุณสมบัติอื่นอาจไม่ใช่ (เช่นคุณลักษณะที่อัปเดตอย่างต่อเนื่องของอุปกรณ์อิเล็กทรอนิกส์ใหม่หรือคุณลักษณะของภาพยนตร์) ในสภาพแวดล้อมเช่นภาพยนตร์เพลงและข่าวสินค้าคงคลังอาจมีการเปลี่ยนแปลงอย่างรวดเร็วและมีขนาดใหญ่จนการรับและบำรุงรักษาคุณลักษณะนั้นยากหรือมีค่าใช้จ่ายสูงเกินไป ในบางสภาพแวดล้อมที่มีปริมาณการซื้อขายสูงอาจมีข้อมูลภายนอกอยู่ ตัวอย่างเช่น Pandora Radio ที่ใช้ CF สามารถใช้คุณลักษณะ 400 รายการทั้งเพลงและศิลปินที่จัดทำ

โดยโครงการจีโนมดนตรีเพื่อค้นหาความคล้ายคลึงกัน มะเขือเทศเน่าเว็บไซต์แนะนำภาพยนตร์เป็นอีกตัวอย่างหนึ่งของการนำ CF มาใช้

เพิ่มประสิทธิภาพ:

หากตัวจำแนกมีอะไรที่จะทำงานกับอื่น นอกเหนือจากการซื้อแบบไบนารีหรือไม่ซื้อคำแนะนำผลลัพธ์จะขาดความแม่นยำ วิธีแก้ทั่วไปที่มีค่าในบางสภาพแวดล้อม แต่ไม่รวมอยู่ในสภาพอื่น

- โพสต์คะแนนการแนะนำ: ผู้ใช้จะได้รับตัวเลือกในการใช้ปุ่มชอบหรือไม่ชอบสำหรับคำแนะนำที่นำเสนอแม้ว่าพวกเขาจะไม่ได้ซื้อหรือเลือก
- โพสต์ซื้อคะแนน: ผู้ใช้ให้คะแนน (ปกติ 1 ถึง 5) เกี่ยวกับความพึงพอใจของพวกเขากับรายการ
- โปรไฟล์อาสาสมัครการบริโภคล่วงหน้า: สำหรับแอปพลิเคชันจำนวนมากที่ขอให้ผู้ใช้ล่วงหน้าเพื่อให้ข้อมูลโปรไฟล์บางอย่างเกี่ยวกับการกำหนดลักษณะช่วยในทางเลือกการกรองล่วงหน้าอย่างมาก
- การทำโปรไฟล์ผู้ใช้: โปรไฟล์ ผู้ใช้สามารถปรับปรุงด้วยการแบ่งส่วนหรือข้อมูลประชากร

จุดแข็งและจุดอ่อน:

1. ระบบผู้แนะนำที่ใช้จริงส่วนใหญ่แก้ปัญหา Cold Start ของวิธีจัดการกับผู้เยี่ยมชมที่มีประวัติเพียงเล็กน้อยหรือไม่มีเลยเนื่องจากคำแนะนำสามารถขับเคลื่อนโดยการรวมประวัติการสืบค้น (สั้น ๆ) พร้อมทั้งชอบหรือไม่ชอบหรือการทำโปรไฟล์โดยสมัครใจ
2. การปรับปรุงโปรไฟล์อาจไม่สามารถทำได้หรืออยู่ในระดับที่เท่ากันสำหรับผู้ใช้ทุกคนที่ปล่อยให้ปัญหาการกระจัดกระจายในการคำนวณ คุณอาจไม่มีแท็กสำหรับทุกหมวดหมู่สำหรับทุกรายการ
3. การใช้งาน CF โดยทั่วไปสามารถเรียนรู้จากแหล่งเดียว (เว็บไซต์ของคุณ) ซึ่งจำกัดความสามารถในการให้คำแนะนำที่ไม่ได้เป็นส่วนหนึ่งของ 'เมล็ดพันธุ์' ดั้งเดิม ระบบบางระบบเช่นตัวรวบรวมข่าวได้รับการออกแบบเพื่อรวมเนื้อหาจากหลายแหล่งเพื่อปรับปรุงข้อจำกัดนี้
4. หากไม่มีการขยายโปรไฟล์หรือข้อเสนอแนะของผู้ใช้ระบบ CF จะใช้การคำนวณอื่น นอกเหนือจากการบริโภคหรือไม่ได้ใช้

4. การคัดกรอง (Collaborative Filtering: CB)

ตัวกรองความร่วมมือมุ่งเน้นไปที่ผู้ใช้และผู้ใช้อื่นที่พบว่ามีลักษณะทางคณิตศาสตร์คล้ายกับผู้ใช้ ในทางทฤษฎีไม่จำเป็นต้องมีคุณลักษณะเฉพาะสำหรับเนื้อหาที่ CB สามารถอนุมานได้ หลังจากนั้นเราจะเห็นว่าการเพิ่มคุณสมบัติเนื้อหาสามารถเพิ่มประสิทธิภาพ แต่ไม่จำเป็นต้องใช้ทางเทคนิค หลักฐานพื้นฐานคือหากผู้ใช้สองคนมีความชอบและชอบที่ไม่เหมือนกันในอดีตที่ผ่านมาพวกเขาจะยังคงมีความคล้ายคลึงกันที่แข็งแกร่งในอนาคต CB จะจับคู่คนที่ชอบหนังแนวโรแมนติกกับภาพยนตร์ที่มีเนื้อหาโรแมนติกที่แข็งแกร่งโดยไม่ต้องมีการกำหนด “ความรัก” เมื่อสร้างความคล้ายคลึงกันแล้วรายการที่บริโภคโดยผู้ใช้รายหนึ่งสามารถแนะนำให้กับผู้ใช้รายอื่นที่คล้ายคลึงกันได้ การมีอยู่ของการจัดอันดับภายหลังการเลือกการบ่งชี้ว่าชอบหรือไม่ชอบทันทีและ / หรือโปรไฟล์ผู้ใช้ที่มีอยู่ก่อนมีความจำเป็นเพื่อให้เทคนิคนี้ทำงานได้ CB พยายามที่จะทำนายการจัดอันดับผู้ใช้สำหรับรายการที่มองไม่เห็น ความถูกต้องของการทำนายสามารถกำหนดได้โดยการเปรียบเทียบการจัดอันดับที่คาดการณ์กับการจัดอันดับจริงเมื่อได้รับรายการที่แนะนำ

อัลกอริทึม:

ระบบ CB ส่วนใหญ่ใช้การแยกตัวประกอบเวกเตอร์และเริ่มต้นด้วยการสร้างเวกเตอร์คุณลักษณะที่อธิบายผู้ใช้ (ผลิตภัณฑ์และคุณสมบัติที่ระบุว่าน่าสนใจขนาดและความถี่ของการซื้อก่อนหน้า ฯลฯ) ในระบบ CB ขั้นสูงเพิ่มเติม (รวม CB / CF ระบบ) คุณสมบัติเวกเตอร์นอกจากนี้ยังสร้างขึ้นสำหรับ

ผลิตภัณฑ์ (ผู้เขียน ประเภท คุณสมบัติ ฯลฯ) การคำนวณความคล้ายคลึงกันของโคไซน์นั้นเกิดขึ้นกับเวกเตอร์คุณลักษณะเพื่อระบุลูกค้าและผลิตภัณฑ์ที่คล้ายคลึงกัน ข้อเสนอแนะสามารถทำขึ้นอยู่กับความคล้ายคลึงกันของลูกค้าหรือความคล้ายคลึงกันของผลิตภัณฑ์กับผลิตภัณฑ์อื่นที่ลูกค้าได้ซื้อหรือเรียกดูการแยกตัวประกอบตามลูกค้าเป็นพีชคณิตเชิงเส้นตรงไปตรงมาและสามารถรีเฟรชในหน่วยความจำแบบเรียลไทม์หลังจากการจัดอันดับแต่ละครั้งหรือการเลือกหรือไม่เลือกรายการที่แนะนำ ประสิทธิภาพจะเพิ่มขึ้นตามหลักวิชาการระหว่างการโต้ตอบของผู้ใช้กับไซต์ของคุณเมื่อพวกเขาจัดอันดับและเลือกรายการใหม่ โปรดทราบว่าสำหรับฐานผู้ใช้ขนาดใหญ่มากที่มีสินค้าคงคลังขนาดใหญ่มากพลังการคำนวณอาจมีความสำคัญ แต่อยู่ในช่วงของบริการคลาวด์ MPP การแยกตัวประกอบตามผลิตภัณฑ์มีแนวโน้มมากขึ้นจากการรวมกลุ่มหรือสถิติเช่นสหสัมพันธ์ของเพียร์สันและดำเนินการในแบทช์ออฟไลน์

คุณสมบัติของผู้ใช้:

ระบบ CB พึ่งพาข้อมูลสองประเภทเพื่อให้ทำงานได้ดี รายการแรกคือรายการที่เราขอให้ผู้ใช้ระบุ ได้แก่

- ให้คะแนนรายการในระดับห้าจุดหรือใช้ปุ่มชอบหรือไม่ชอบทันที
- ขอให้ผู้ใช้ค้นหารายการและปฏิบัติตามข้อกำหนดที่ใช้
- การสร้างโปรไฟล์ของไลค์และไม่ชอบตามปัจจัยต่าง ๆ เช่นประเภทผู้แต่งหรือปัจจัยที่คล้ายกัน
- นำเสนอสองทางเลือกและถามผู้ใช้ว่าดีกว่า (การวิเคราะห์ร่วมกัน)

เราจะรวบรวมข้อมูลเกี่ยวกับพฤติกรรมออนไลน์ที่แท้จริงของผู้ใช้ซึ่งรวมถึง:

- การสังเกตรายการที่ดูรวมถึงเวลาที่ใช้ดู
- การซื้อจริง
- หากได้รับอนุญาตให้รวบรวมข้อมูลจากคอมพิวเตอร์ของผู้ใช้เช่นรายการที่รับฟังหรือดูจริง
- การใช้แหล่งข้อมูลภายนอกเช่นไซต์เครือข่ายสังคมออนไลน์เพื่อเพิ่มไลค์และไม่ชอบสำหรับผู้ใช้รายนี้

จุดแข็งและจุดอ่อน:

1. CBs ไม่พึ่งพาแอตทริบิวต์เนื้อหาที่กำหนดไว้ล่วงหน้าและสามารถให้คำแนะนำที่ถูกต้องสำหรับเนื้อหาประเภทต่างๆโดยไม่จำเป็นต้องมีความเข้าใจในรายการเอง แอ็ตทริบิวต์สังเคราะห์เหล่านี้มีค่าสามารถส่งออกไปยังระบบกรองเนื้อหาได้อย่างไรก็ตามอาจยากที่จะทำให้มนุษย์ตีความได้
2. CBs ต้องการข้อมูลจำนวนมากเพื่อวิเคราะห์ ในกรณีที่ความหลากหลายของรายการมีขนาดใหญ่มากอาจมีรายการที่ไม่ได้ติดแท็กจำนวนมากที่สร้างปัญหา Sparsity ในการคำนวณ อาจเป็นการยากที่จะค้นหาผู้ใช้รายอื่นที่ให้คะแนนรายการเดียวกัน แม้แต่ไอเท็มที่ได้รับความนิยมมากที่สุดอาจมีเรตติ้งไม่กี่อย่างเท่านั้น
3. รายการที่ใหม่และไม่เคยได้รับการจัดอันดับไม่สามารถแนะนำได้เนื่องจากคำแนะนำนั้นขึ้นอยู่กับการให้คะแนนก่อน
4. CBs มักจะแนะนำรายการที่เป็นที่นิยมและอาจอยู่ภายใต้การดำเนินการสำหรับผู้ใช้ที่มีรสนิยมเฉพาะ

5. การแยกตัวประกอบเมทริกซ์อาจเหมาะสมเกินไปหากมีพารามิเตอร์มากเกินไป
6. รายการยอดนิยมส่วนใหญ่ (เช่นภาพยนตร์) สามารถดำเนินการแยกต่างหากจากสินค้าคงคลังแบบเต็ม (ถ้ามีขนาดใหญ่มาก) เพื่อให้สามารถอัปเดตรายการยอดนิยมแบบเรียลไทม์แม้ว่าสินค้าคงคลังทั้งหมดมีขนาดใหญ่เกินไปที่จะประมวลผลหลังจากการเลือกของผู้ใช้แต่ละคน

5. ไฮบริด (Hybrid)

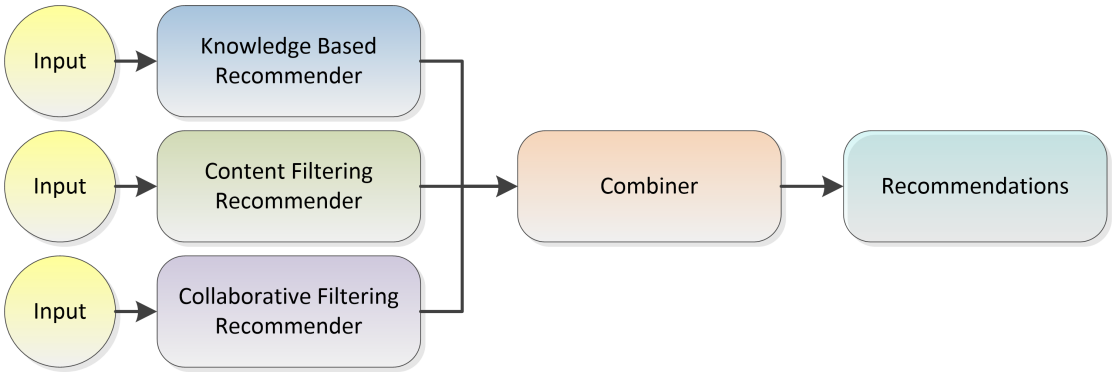
มีการตีความสองอย่างของผู้แนะนำไฮบริดและคุณควรคิดว่าพวกเขาเป็นทั้งสองด้านกับเหรียญเดียวกัน

- ความรู้
- ระบบ CF / CB แบบรวม

1. ความรู้

การเปลี่ยนแปลงนี้ค่อนข้างง่ายที่จะเข้าใจเนื่องจากมันเกี่ยวข้องกับการเพิ่มกฎโดยผู้เชี่ยวชาญในประเด็นมนุษย์ ผู้จัดการฝ่ายการตลาดผลิตภัณฑ์ที่ดีสามารถกำหนดผลิตภัณฑ์ที่ทำและไม่ไปด้วยกันได้บ่อยครั้งและผลิตภัณฑ์ใดบ้างที่อาจเป็นส่วนเสริมและเสริม

2. ระบบ CF / CB แบบรวม เมื่อรู้เกี่ยวกับผู้แนะนำ CF และ CB อาจจะเป็นประโยชน์ต่อทั้งสองฝ่าย โปรดทราบว่าระบบ CB บางเวอร์ชันใช้ทั้งแอททริบิวต์ลูกค้าและไอเท็มแล้วแม้ว่าแอททริบิวต์ไอเท็มในกรณีนี้โดยทั่วไปจะเป็นตัวแปรที่คำนวณจากการสังเคราะห์ซึ่งอาจไม่มีตรรกะที่เข้าใจได้ง่าย ในกรณีนี้ระบบ CF / CB ที่รวมกันนั้นแตกต่างกันไปขึ้นอยู่กับหลาย ๆ วิธีที่ระบบเหล่านี้สามารถรวมกันได้



รูปที่ 2.1: ระบบ CF / CB แบบรวม

ที่มา : https://www.datasciencecentral.com/profiles/blogs/5-types-of-recommenders

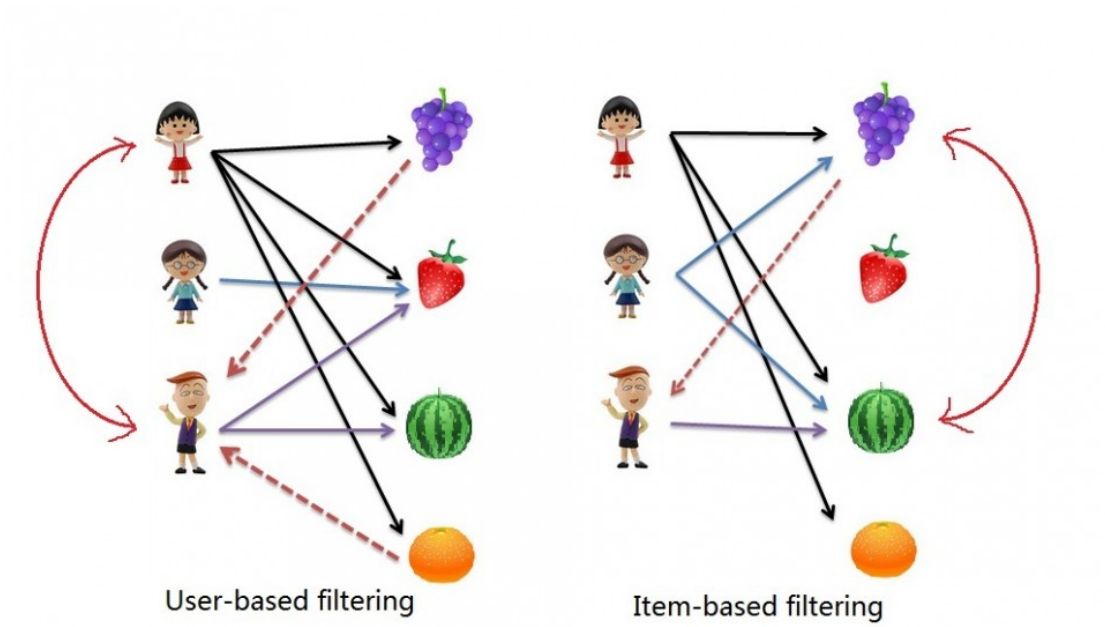
ไม่มีแนวปฏิบัติสากลที่ดีที่สุดเฉพาะสำหรับไฮบริดผู้แนะนำซึ่งจะต้องใช้ความเข้าใจของคุณในสถานการณ์พิเศษของธุรกิจของคุณ กลยุทธ์บางอย่างอาจรวมถึง:

- น้ำหนัก: กลยุทธ์การลงคะแนนระหว่างผู้แนะนำคู่แข่ง
- การเลือกฐานกฎ: ตามกฎที่คุณกำหนดในบางกรณีผู้แนะนำรายหนึ่งจะโดดเด่นกว่าอีกฝ่าย
- การรวมกัน: คำแนะนำจากทั้งสองจะถูกนำเสนอเข้าด้วยกัน
- การรวมคุณสมบัติ: การใช้ข้อมูลเมตาจากผู้แนะนำทั้งสองเพื่อขับยังคงเป็นผู้แนะนำที่สาม

ในทางปฏิบัติแล้วการเพิ่มองค์ประกอบฐานความรู้ไปยังเทคนิคอื่น ๆ คุณอาจเลือกที่มีกฎฐานความรู้เป็นตัวกรองแรกสำหรับคำแนะนำที่เป็นไปได้หรือเป็นตัวกรองสุดท้าย Netflix ใช้ผู้แนะนำ CB / CF ไฮบริด มันมีคำแนะนำทั้งสองตามนิสัยของลูกค้าที่คล้ายกัน (Collaborative Filtering) เช่นเดียวกับคำแนะนำจากภาพยนตร์ที่ได้รับการจัดอันดับสูงซึ่งมีลักษณะคล้ายกันโดยแอตทริบิวต์เนื้อหา (การกรองเนื้อหา)

2.6.2 Collaborative Filtering Technique

Collaborative filtering [8] คือระบบผู้แนะนำประเภทหนึ่งที่ใช้ข้อมูลการกระทำของเราและเพื่อน เพื่อใช้แนะนำการกระทำที่เราน่าจะทำหรือน่าสนใจ แต่ระบบนี้มีปัญหา Cold Start Problem หรือคือการที่ต้องมีข้อมูลเยอะพอสมควรเพื่อเป็นฐานข้อมูลในการแนะนำ และนอกจากนั้นระบบแบบนี้จะแนะนำการกระทำที่เป็นกระแสในช่วงนั้น โดยไม่แนะนำการกระทำนอกกระแส ก็เป็นข้อเสียเปรียบของระบบผู้แนะนำแบบนี้ก็ว่าได้ ระบบผู้แนะนำนี้ยกตัวอย่าง เช่น Amazon, Facebook, Twitter, LinkedIn, Spotify, Google News และ Last.fm Collaborative filtering ช่วยค้นหารูปแบบ Patterns หรือกฎ Rules ในการแนะนำสินค้า ซึ่งแบ่งได้ง่ายๆ เป็น 2 แบบ ได้แก่ User-Based Filtering ซึ่งเป็นการแนะนำโดยหาลูกค้าที่พฤติกรรมเหมือนกัน และ Item-Based Filtering ซึ่งเป็นการแนะนำโดยหาสินค้าที่ถูกซื้อด้วยลูกค้ากลุ่มเดียวกัน สามารถแสดงให้เห็นข้อแตกต่างได้ดังรูปที่ 2.1



รูปที่ 2.2: Collaborative filtering แบบ User-Based Filtering และ แบบ Item-Based Filtering

ที่มา : http://www.salemmarafi.com/code/collaborative-filtering-r/

K-Nearest Neighbour หรือ KNN เป็นเทคนิคที่นิยมมาใช้ใน CB ซึ่งจะช่วยจัดกลุ่มผู้ใช้หรือกลุ่มสินค้าที่ควรแนะนำได้ โดยเทคนิคนี้จะสามารถบอกรายการแนะนำเพื่อตอบคำถามต่อไปนี้

- สินค้าแนะนำ k รายการให้กับสมาชิก u
- สมาชิก k คน ที่มีรสนิยมใกล้เคียงกันกับ u

2.6.3 การทดสอบประสิทธิภาพของเทคนิคที่ใช้

เทคนิคที่ใช้สำหรับระบบผู้แนะนำนั้นโดยพื้นฐานแล้วจะสร้างต้นแบบจากการเรียนรู้ข้อมูลการให้คะแนน rating ของสมาชิกทุกคนในระบบ จากนั้นจะสามารถใช้ต้นแบบเหล่านี้เพื่อประเมินว่าสมาชิก u จะให้ rating สินค้า i เท่าใด ซึ่งโดยปกติแล้ว rating จะมีค่าระหว่าง 1 ถึง 5 ประสิทธิภาพของต้นแบบจึงสามารถวัดได้จากค่าประเมินว่าแตกต่างจากค่า rating จริงเท่าใด ในเชิงวิจัยการวัดประสิทธิภาพของต้นแบบจากชุดข้อมูล data ทั้งหมด n รายการที่แต่ละรายการประกอบไปด้วย สมาชิก(u), สินค้า(i), rating(r) สามารถวัดได้จากค่าแตกต่างทางสถิติรวมทุกรายการที่เรียกว่า Root-Mean Square Error: RMSE หรือไม่เอาผลต่างไปยกกำลังสองแล้วถอดรากที่สองแต่ใช้ค่าสัมบูรณ์แทนเรียกว่า Mean Absolute Error (MAE)

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(r_j - e_j)^2}$$

$$MAE = \frac{1}{n}\sum_{j=1}^{n}\left|r_j - e_j\right|$$

เมื่อ:

- rj หมายถึง รายการที่ j ใน data ซึ่งประกอบไปด้วย หมายเลขสมาชิก (u), หมายเลขสินค้า (i), rating (r)
- ej หมายถึง ค่าประเมินที่ต้นแบบคำนวณได้จากการระบุ หมายเลขสมาชิก (u) และหมายเลขสินค้า (i)

ชุดข้อมูล data สำหรับสร้างต้นแบบนั้นจะแบ่งออกเป็น K กลุ่มเท่าๆ กันเพื่อใช้ K-1 กลุ่มในการสร้างต้นแบบ (trainset) และเหลืออีกกลุ่มสำหรับทดสอบความถูกต้อง (testset) โดยทำวนซ้ำสลับแต่ละกลุ่มเป็นกลุ่มทดสอบทั้งหมด K ครั้งเพื่อให้แต่ละกลุ่มได้มีโอกาสใช้เป็นกลุ่มทดสอบ การทดสอบประสิทธิภาพแบบนี้เรียกว่า K-fold cross-validation

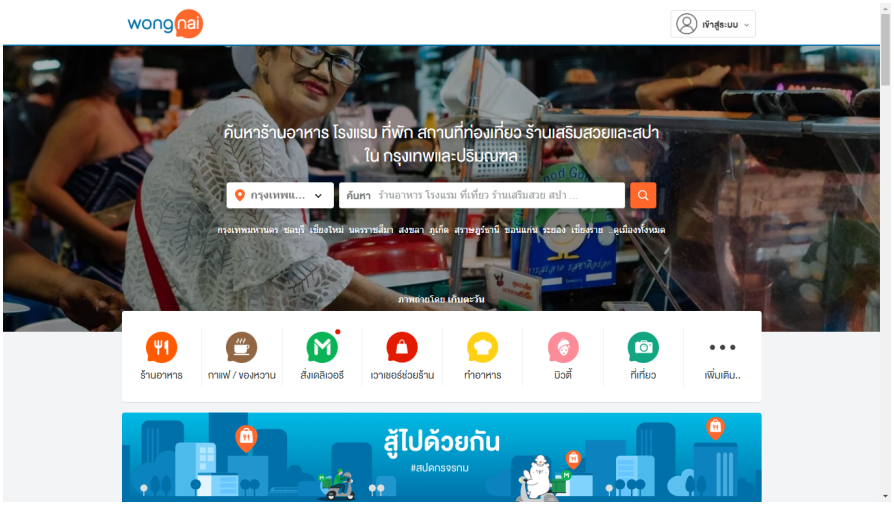
2.7 เอกสารและงานวิจัยที่เกี่ยวข้อง

2.7.1 เว็บแอปพลิเคชัน Wongnai

Wongnai [9] เป็นเว็บไซต์และแอปพลิเคชันที่ให้บริการเกี่ยวกับการค้นหาร้านอาหาร สถาน ที่ท่องเที่ยว ร้านเสริมสวย มีฟังก์ชันการทำงานพื้นฐานอันได้แก่ การค้นหา ดูข้อมูล โพสต์รีวิว บันทึกร้านที่ชอบ เป็นต้น

2.7.2 ข้อแตกต่างระหว่างเว็บแอปพลิเคชัน wongnai กับเว็บของโครงงาน

เว็บแอปพลิเคชันของ Wognai ยังไม่มีฟังก์ชันการแนะนำการแต่งกายให้ผู้ใช้ โดยในเว็บแอปพลิเคชั่นของ Wognai ยังขาดในส่วนของการแนะนำเสื้อผ้า ซึ่งเป็นอีกหนึ่งฟังก์ชันที่หน้าสนใจเพราะในปัจจุบัน คนนิยมเข้าซื้อของออนไล์มากยิ่งขึ้น หากเพิ่มในส่วนของการขายเสื้อผ้าหรือการแนะนำรายการเสื้อผ้าให้แก่ผู้ใช้งานได้อีกจะยิ่งทำให้ระบบมีการให้บริการแก่ลูกค้าที่ครอบคลุมมากยิ่งขึ้น ผู้พัฒนาจึงได้ทำฟังก์ชันการแนะนำการแต่งกายที่เหมาะสมตามรสนิยมของผู้ใช้เพิ่มลงในเว็บของโครงงาน



รูปที่ 2.3: หน้าแรกของเว็บไซต์ Wongnai

ที่มา : https://www.wongnai.com/

บทที่ 3

การวิเคราะห์และออกแบบระบบ

ในบทนี้จะกล่าวถึงการวิเคราะห์และออกแบบระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี ซึ่งการวิเคราะห์และออกแบบระบบที่ใช้ในการพัฒนาจะแบ่งออกเป็น 6 ขั้นตอนดังนี้

3.1 โครงสร้างภาพรวมของระบบ (System Architecture) เป็นการออกแบบภาพรวมและเทคโนโลยีของระบบ

3.2 System Requirements คือ ความต้องการหรือสิ่งที่ระบบควรจะทำ หรือหน้าที่หลักของ ระบบที่จะต้องทำ

3.3 User Interface Design เป็นการออกแบบส่วนต่อประสานกับผู้ใช้

3.4 Use Case Diagram เป็นแผนภาพที่ใช้แสดงให้ทราบว่าระบบทำงานหรือมีหน้าที่ใดบ้าง

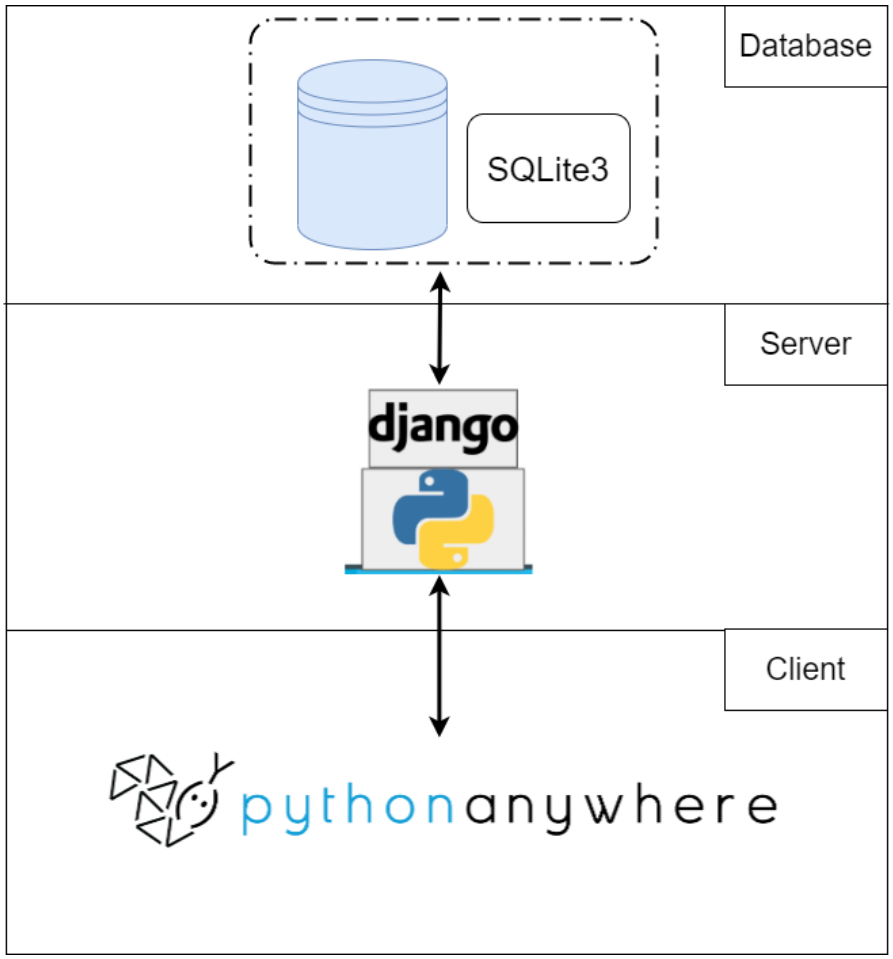
3.5 Class Diagram เป็นแผนภาพที่ใช้แสดง Class และความสัมพันธ์ระหว่าง Class

3.6 Sequence Diagram เป็นแผนภาพที่ใช้แสดงให้เห็นถึงการตอบโต้ข้อมูลระหว่างคลาส เรียงตามลำดับของเวลาที่เกิดเหตุการณ์จากน้อยไปมาก

3.1 โครงสร้างภาพรวมของระบบ

ความหมายของ System Architecture [10] กรอบโครงสร้างของระบบ ITS ที่อธิบายความสัมพันธ์ขององค์ประกอบต่างๆถึงขั้นการเชื่อมต่อกันของระบบย่อยต่างๆ รวมไปถึงระบบ ITS ต่างๆโดยจัดกลุ่มองค์ประกอบไว้หลายๆลักษณะเพื่อให้ผู้เกี่ยวข้อง (Stakeholder) จากพื้นฐาน (สาขาอาชีพ) ต่างๆกันสามารถทำความเข้าใจได้ง่าย เช่น การจัดแบ่งองค์ประกอบตามลักษณะการทำงานของระบบ (functional components) หรือจัดแบ่งองค์ตามลักษณะการเป็นเจ้าของภาระกิจหรือตามลักษณะของที่จะเข้าร่วมในภารกิจนั้นๆ (marketing or institutional components)

การออกแบบ System architecture แสดงภาพรวมและเทคโนโลยีของระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี ซึ่งมีรายละเอียดดังรูปที่ 3.1



รูปที่ 3.1: System architecture ระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี

จากรูปที่ 3.1 สามารถอธิบายโครงสร้างและเทคโนโลยีของระบบโดยแบ่งเป็น 3 ส่วนหลัก ดังนี้

1. Database ระบบใช้ระบบจัดการฐานข้อมูล SQLite3
2. Server กระบวนการทำงานในส่วนของเซิฟเวอร์ (server) คือใช้ Django เป็น framework ที่ใช้ในการสร้าง web application
3. Client เว็บแอปพลิเคชันแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี มีวัตถุประสงค์เพื่อรองรับการทำงานของผู้ใช้บนเว็บบราวเซอร์พัฒนาด้วย pythonanywhere

3.2 System Requirements

3.2.1 Functional Requirements

ระบบของเว็บแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี แบ่งความสามารถของระบบตามประเภทของผู้ใช้งานดังนี้

3.2.1.1 ผู้ใช้งาน

- สามารถสมัครสมาชิกได้

- สามารถเข้าสู่เข้าสู่ระบบได้
- สามารถแก้ไขข้อมูลผู้ใช้งานได้
- สามารถดูรายการเสื้อผ้าจากร้านค้าได้
- สามารถดูรายการร้านค้าได้
- สามารถดูรายการแนะนำเสื้อผ้าได้
- สามารถบันทึกรายการเสื้อผ้าที่ถูกใจได้
- สามารถบันทึกรายการร้านค้าที่ถูกใจได้
- สามารถดูบันทึกรายการเสื้อผ้าที่ถูกใจได้
- สามารถดูบันทึกรายการร้านค้าที่ถูกใจได้
- สามารถดูค้นหาเสื้อผ้าได้

3.2.1.2 ร้านค้า

- สามารถสมัครสมาชิกได้
- สามารถเข้าสู่ระบบได้
- สามารถแก้ไขข้อมูลร้านค้าได้
- สามารถเพิ่ม ลบ แก้ไข ข้อมูลเสื้อผ้าในร้านค้าได้
- สามารถดูรายการคนที่ถูกใจเสื้อผ้าได้
- สามารถดูรายการคนที่ถูกใจร้านค้าได้

3.2.1.3 ผู้ดูแลระบบ

- สามารถเข้าสู่ระบบได้
- สามารถเพิ่ม ลบ แก้ไข ข้อมูลร้านค้าได้
- สามารถเพิ่ม ลบ แก้ไข ข้อมูลผู้ใช้งานได้

3.2.2 Non-functional Requirements

- ต้องเชื่อมต่ออินเตอร์เน็ตเพื่อใช้งานเว็บ
- สนับสนุนการแสดงผลในหน้าจอคอมพิวเตอร์เท่านั้น

3.3 User Interface Design

ในการออกแบบแบ่งการออกเป็น 3 ส่วนคือ ส่วนของผู้ใช้งาน ส่วนของร้านค้า และส่วนของผู้ดูแลระบบ

3.3.1 ส่วนของผู้ใช้งาน

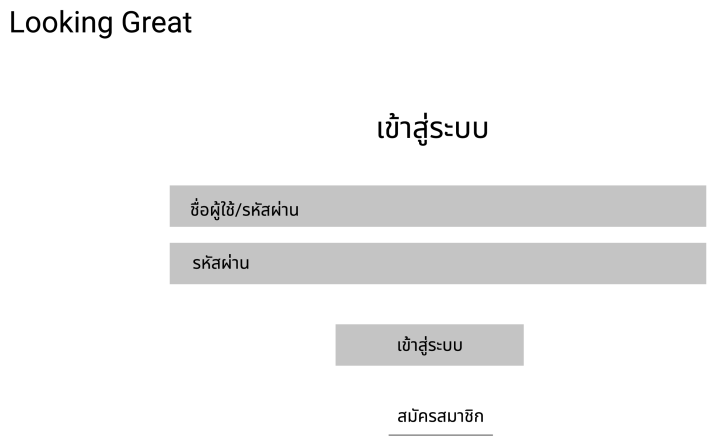
โดยประกอบด้วยส่วนการออกแบบหน้าจอทำงาน ดังนี้

1. การออกแบบหน้าสมัครสมาชิก

รูปที่ 3.2: หน้าสมัครสมาชิก

จากภาพที่ 3.2 การออกแบบหน้าสมัครสมาชิก หากผู้ใช้ยังไม่มีบัญชีในระบบผู้ใช้งานสามารถทำการสมัครสมาชิกเพื่อเข้าใช้งานในระบบได้จากหน้าสมัครสมาชิก โดยผู้ใช้จำเป็นต้องกรอกข้อมูลให้ครบก่อนกดปุ่มสมาชิก

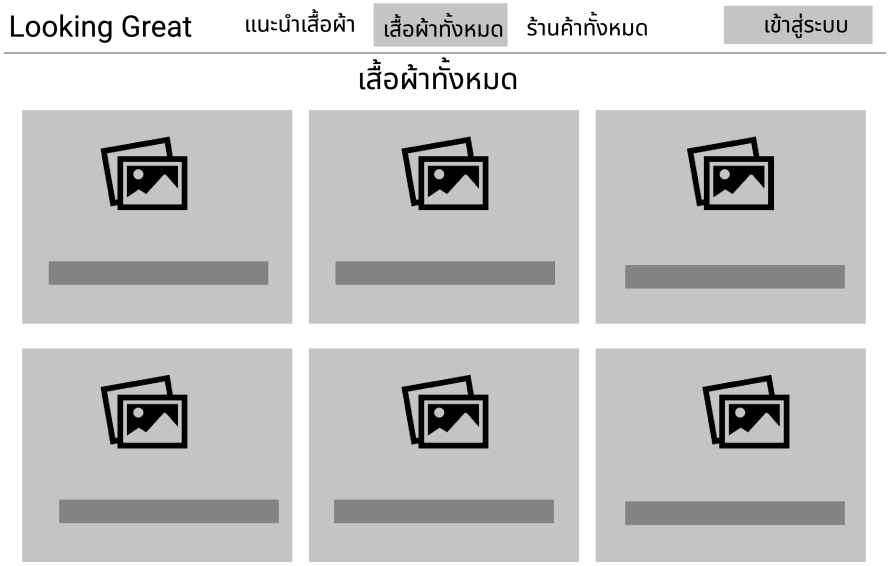
2. การออกแบบหน้าจอเข้าสู่ระบบ



รูปที่ 3.3: หน้าเข้าสู่ระบบ

จากภาพที่ 3.3 แสดงหน้าจอเข้าสู่ระบบเมื่อผู้ใช้ยังไม่ได้ทำการเข้าสู่ระบบ โดยจำเป็นต้องกรอกชื่อผู้ใช้และรหัสผ่านเพื่อใช้ในการเข้าสู่ระบบ

3. การออกแบบหน้าจอเสื้อผ้าทั้งหมด



รูปที่ 3.4: หน้าจอเสื้อผ้าทั้งหมด

จากภาพที่ 3.4 แสดงหน้าจอเสื้อผ้าทั้งหมด ซึ่งการออกแบบในหน้านี้จะแยกเป็นประเภทของเสื้อผ้ามีทั้งหมด 6 ประเภท ได้แก่ เสื้อ กางเกง กระโปรง เดรส สูท และอื่นๆ หากผู้ใช้ต้องการดูรายละเอียดของเสื้อผ้าก็สามารถคลิกดูที่ปุ่มดูรายละเอียดเสื้อผ้าได้ และจะมีในส่วนของการค้นหาเสื้อผ้าถ้าผู้ใช้ต้องการค้นหาเสื้อผ้าก็สามารถค้นหาได้

4. การออกแบบหน้ารายละเอียดเสื้อผ้า

รูปที่ 3.5: หน้าจอรายละเอียดเสื้อผ้า

จากภาพที่ 3.5 แสดงหน้าจอรายละเอียดเสื้อผ้า โดยข้อมูลที่แสดงในหน้านี้ประกอบไปด้วย รูปสินค้า ชื่อสินค้า ชื่อร้านค้า ราคา ขนาดสินค้า สีสินค้า จำนวนสินค้า และปุ่มเพิ่มเป็นสินค้าโปรด โดยในส่วนของปุ่มสินค้าโปรดจะแสดงก็ต่อเมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบเท่านั้น

5. การออกแบบหน้าจอร้านค้าทั้งหมด

รูปที่ 3.6: หน้าจอร้านค้าทั้งหมด

จากภาพที่ 3.6 โดยจะแสดงรายการร้านค้าทั้งหมดที่มีในระบบ จะแสดงรูปภาพร้านค้า ชื่อร้านค้า และหากผู้ใช้ต้องการดูรายละเอียดของร้านค้าก็สามารถคลิกดูที่ปุ่มดูรายละเอียดร้านค้าได้

6. การออกแบบหน้าจอรายละเอียดร้านค้า

รูปที่ 3.7: หน้าจอรายละเอียดร้านค้า

จากภาพที่ 3.7 แสดงหน้าจอรายละเอียดร้านค้า โดยข้อมูลที่แสดงในหน้านี้ประกอบไปด้วย รูปร้านค้า ชื่อร้าค้า ที่อยู่ร้าน เบอร์โทรติดต่อ อีเมล์ (E-mail), Facebook สินค้าในร้าน และปุ่มเพิ่มเป็นร้านโปรดโดย ในส่วนของปุ่มร้านโปรดจะแสดงก็ต่อเมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบเท่านั้น

7. การออกแบบหน้าจอข้อมูลผู้ใช้งาน

รูปที่ 3.8: หน้าจอข้อมูลผู้ใช้งาน

จากภาพที่ 3.8 แสดงหน้าจอข้อมูลผู้ใช้งานโดยหน้านี้จะแสดงข้อมูลของผู้ใช้งาน ได้แก่ ชื่อผู้ใช้ วัน เดือน ปีเกิด เพศ รูปร่าง สีผิว และรูปโปร์ไฟล์ นอกจากนี้หากผู้ใช้งานต้องการแก้ไขข้อมูลก็สามารถกดที่ปุ่มแก้ไขข้อมูลได้ และสามาถดูรายการเสื้อผ้าและร้านค้าโปรดได้

8. การออกแบบหน้าแสดงรายการสินค้าโปรด

รูปที่ 3.9: หน้าจอรายการสินค้าโปรด

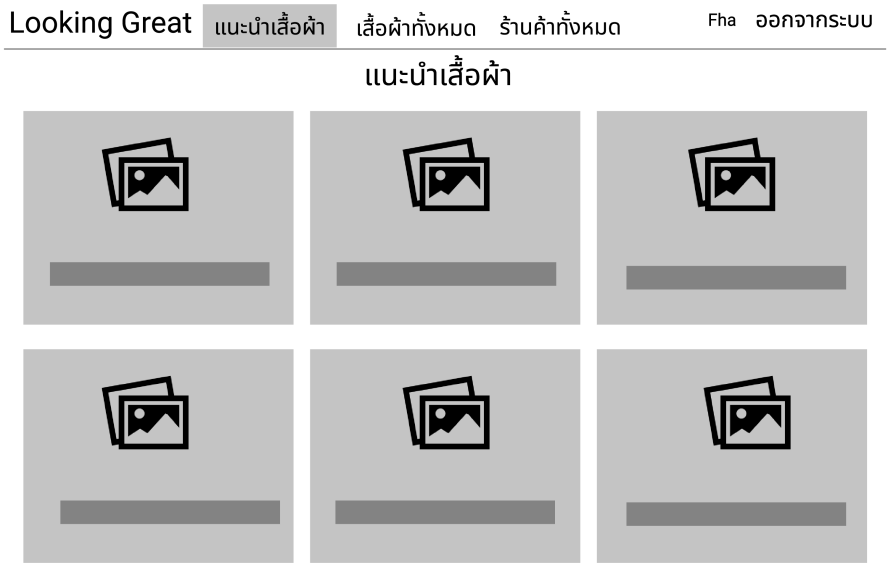
จากภาพที่ 3.9 แสดงหน้าจอรายละเอียดสินค้าโปรด โดยจะแสดงรายการเสื้อผ้าที่ผู้ใช้งานได้ทำการเพิ่มเป็นสินค้าโปรดแล้ว นอกจากนี้สามาถดูรายละเอียดเสื้อผ้าและสามารถลบรายการเสื้อผ้าออกจากรายการโปรดได้

9. การออกแบบหน้ารายการร้านโปรด

รูปที่ 3.10: หน้าจอรายการร้านโปรด

จากภาพที่ 3.10 แสดงหน้าจอข้อมูลผู้ใช้งานโดยหน้านี้จะแสดงข้อมูลของผู้ใช้งาน ได้แก่ ชื่อผู้ใช้ วัน เดือน ปีเกิด เพศ รูปร่าง สีผิว และรูปโปร์ไฟล์ นอกจากนี้หากผู้ใช้งานต้องการแก้ไขข้อมูลก็สามารถกดที่ปุ่มแก้ไขข้อมูลได้ และสามาถดูรายการเสื้อผ้าและร้านค้าโปรดได้

10. การออกแบบหน้ารายการแนะนำเสื้อผ้า



รูปที่ 3.11: หน้าจอรายการแนะนำเสื้อผ้า

จากภาพที่ 3.11 แสดงหน้าจอรายการแนะนำเสื้อผ้า โดยจะแสดงรายการเสื้อผ้าที่มีคนชื่นชอบมากที่สุดมา 9 ชิ้น ในกรณีที่ยังไม่มีการเพิ่มรายการเสื้อผ้าที่ชอบ และในกรณีที่มีรายการเสื้อผ้าที่ถูกใจแล้ว ระบบจะแนะนำรายการเสื้อผ้า 9 ชิ้น ตามรายการเสื้อผ้าที่ผู้ใช้ชื่นชอบ

3.3.2 ส่วนของร้านค้า

โดยประกอบด้วยส่วนการออกแบบหน้าจอทำงาน ดังนี้

1. การออกแบบหน้าจอเพิ่มร้านค้า

รูปที่ 3.12: หน้าจอเพิ่มร้านค้า

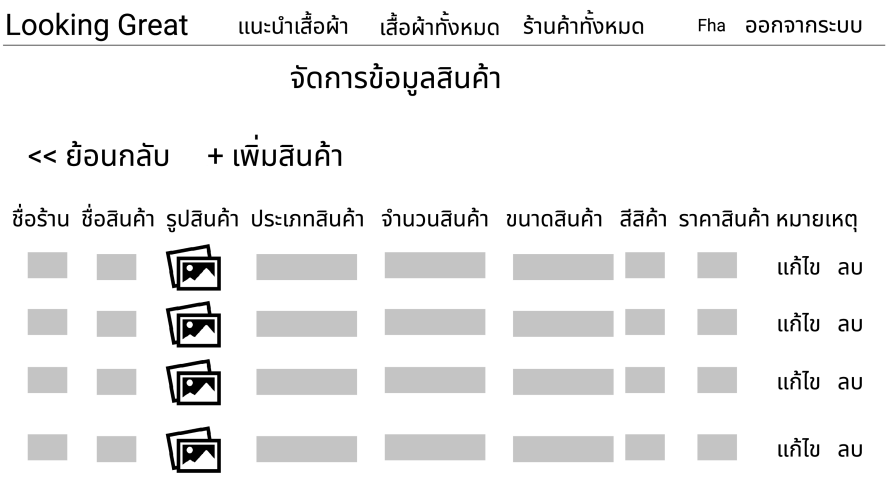
จากภาพที่ 3.12 แสดงหน้าจอเพิ่มร้านค้า ในการเพิ่มร้านค้าผู้ใช้งานจะสามารถเพิ่มร้านได้แค่ 1 ร้าน โดยผู้ใช้จำเป็นต้องกรอกข้อมูลให้ครบก่อนกดปุ่มสมาชิก

2. การออกแบบหน้าจอข้อมูลร้านค้า

รูปที่ 3.13: หน้าจอข้อมูลร้านค้า

จากภาพที่ 3.13 แสดงหน้าจอข้อมูลร้านค้าโดยหน้านี้จะแสดงข้อมูลร้านค้าได้แก่ รูปร้านค้า ชื่อร้าค้า ที่อยู่ร้าน เบอร์โทรติดต่อ อีเมล์ (E-mail) Facebook และรูปโปร์ไฟล์ร้านค้า นอกจากนี้ร้านค้าต้องการแก้ไขข้อมูลร้านก็สามารถกดที่ปุ่มแก้ไขข้อมูลร้านค้าได้ หากร้านค้าต้องการดูรายการสินค้าในร้าค้าสามาถคลิกดูได้ที่จัดการสินค้าเพื่อทำการเพิ่มสินค้า แก้ไข และลบได้ และร้านค้ายังสามาถดูรายการคนที่ถูกใจเสื้อผ้าและร้านค้าได้

3. การออกแบบหน้าจอจัดการข้อมูลสินค้า



รูปที่ 3.14: หน้าจอจัดการข้อมูลสินค้า

จากภาพที่ 3.14 แสดงหน้าจอจัดการข้อมูลสินค้าในส่วนของร้านค้า โดยหน้านี้จะแสดงข้อมูลรายการเสื้อผ้าทั้งหมดที่มีในร้าน ซึ่งร้านค้าสามารถทำการเพิ่มสินค้าได้ สามารถแก้ไขข้อมูลเสื้อผ้า และสามารถลบรายการเสื้อผ้าได้

4. การออกแบบหน้าจอเพิ่มสินค้า

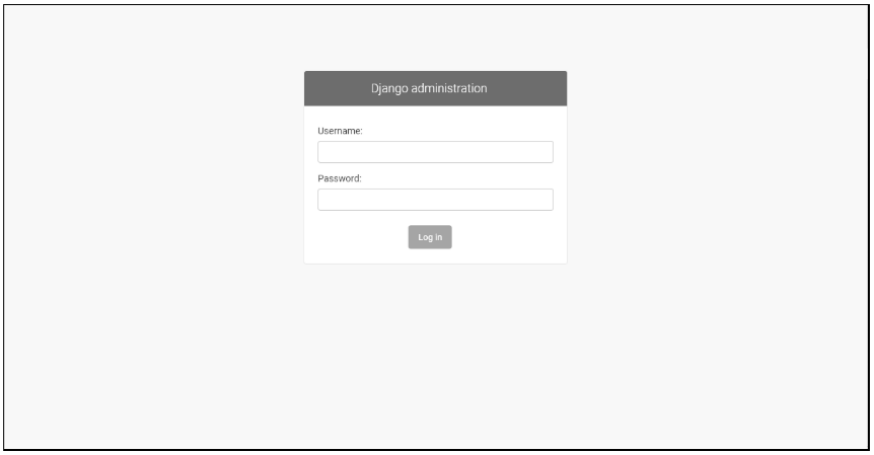
รูปที่ 3.15: หน้าจอเพิ่มสินค้า

จากภาพที่ 3.15 แสดงหน้าจอเพิ่มสินค้า โดยมีช่องให้กรอกชื่อสินค้า ประเภทสินค้า จำนวนสินค้า ขนาดสินค้า สีสินค้า ราคาสินค้า และรูปสินค้า

3.3.3 ส่วนของผู้ดูแลระบบ

โดยประกอบด้วยส่วนการออกแบบหน้าจอทำงาน ดังนี้

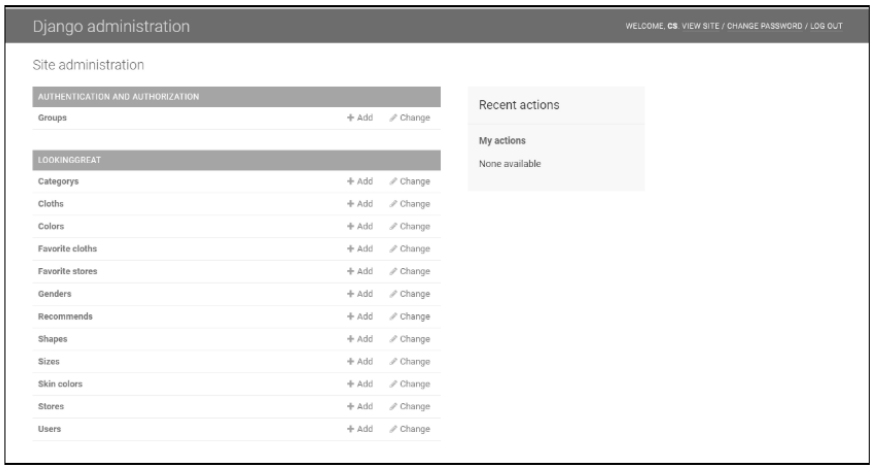
1. การออกแบบหน้าจอเข้าสู่ระบบ



รูปที่ 3.16: หน้าจอเข้าสู่ระบบผู้ดูแลระบบ

จากภาพที่ 3.16 แสดงหน้าจอเข้าสู่ระบบ โดยต้องกรอกชื่อผู้ใช้และรหัสผ่านก่อนกดปุ่ม Login

2. การออกแบบหน้าจอฐานข้อมูลทั้งหมดในระบบ



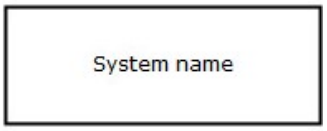
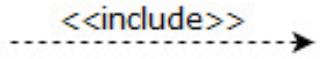
รูปที่ 3.17: หน้าจอฐานข้อมูลทั้งหมดในระบบ

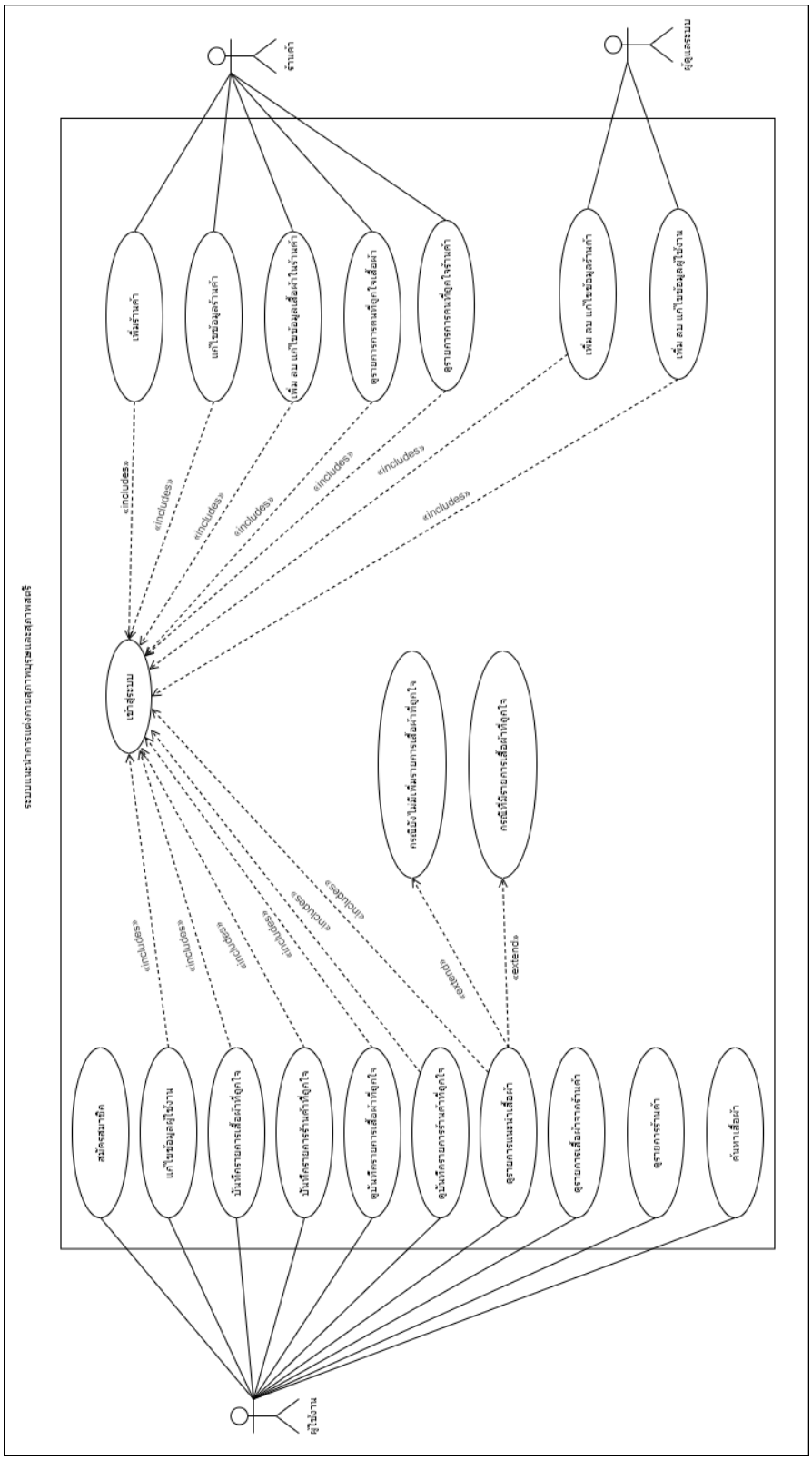
จากภาพที่ 3.17 แสดงหน้าจอฐานข้อมูลทั้งหมดในระบบ ยกตัวอย่างเช่น ข้อมูลผู้ใช้งาน ข้อมูลร้านค้า ข้อมูลเสื้อผ้า เป็นต้น

3.4 Use Case Diagram

Use Case Diagram เป็นแผนผังเพื่อแสดงฟังก์ชันแสดงการทำงานของระบบโดยรวม แสดงส่วนประกอบในระบบและกิจกรรมที่เกิดขึ้นในระบบ สัญลักษณ์ที่ใช้ในการเขียน Use Case Diagram แสดงในตารางที่ 3.1

ตารางที่ 3.1: สัญลักษณ์ของ Use case Diagram

สัญลักษณ์	การใช้งาน
Use case	Use case คือส่วนย่อยของระบบงาน แทนด้วยวงรีและชื่อของ Use case ภายในวงรี
	Actor คือบุคคลหรือระบบงานอื่นที่ใช้งานระบบหรือได้รับประโยชน์จากระบบซึ่งอยู่ภายนอกระบบ แทนด้วยรูปคนและมีชื่อบทบาทการใช้งานระบบ
	เส้นตรงที่แสดงถึงการใช้งาน Use case ของผู้กระทำ
	กรอบสี่เหลี่ยมแสดงถึงขอบเขตของระบบโดยแสดงชื่อระบบภายในหรือด้านบนกรอกสี่เหลี่ยม Use case อยู่ภายในกรอบสี่เหลี่ยม และ actor อยู่ภายนอกกรอบสี่เหลี่ยม
	ความสัมพันธ์แบบ «includes» แสดงว่า Use case หนึ่งดำเนินการตามขั้นตอนของ Use case อื่น โดยแทนด้วยสัณลักษณ์ลูกศรเส้นประ ซึ่ง Use case ที่หางลูกศรเรียกใช้งาน Use case ที่หัวลูกศรทุกครั้งที่มีการทำงาน
	ความสัมพันธ์แบบ «extend» แสดงว่า Use case หนึ่งดำเนินการตามขั้นตอนของ Use case อื่น โดยแทนด้วยสัญลักษณ์ลูกศรเส้นประ ซึ่ง Use case ที่หัวลูกศรเรียกใช้งาน Use case ที่หางลูกศร แต่การใช้งานไม่จำเป็นต้องเกิดขึ้นทุกครั้งขึ้นอยู่กับเงื่อนไขระหว่างการทำงาน



รูปที่ 3.18: Use Case Diagram ระบบแนะนำการแต่งการสุภาพบุรุษและสุภาพสตรี

ตารางที่ 3.2: อธิบาย Use Case หน้าที่ของระบบ ในภาพที่ 3.18

Use Case	คำอธิบาย
สมัครสมาชิก	ผู้ใช้งานและร้านค้าสามารถสมัครสมาชิกเพื่อเข้าใช้งานในระบบได้
เข้าสู่ระบบ	ผู้ใช้งานที่เป็นสมาชิกสามารถใช้ username และ password ของตัวเอง เพื่อเข้าสู่ระบบได้
แก้ไขข้อมูลผู้ใช้งาน	เมื่อผู้ใช้งานเข้าสู่ระบบเรียบร้อย จะสามารถแก้ไขข้อมูลผู้ใช้งานได้
ดูรายการเสื้อผ้าจากร้านค้า	ผู้ใช้งานสามารถเลือกดูรายการเสื้อผ้าจากร้านค้าได้ โดยไม่จำเป็นต้องเข้าสู่ระบบ
ดูรายการร้านค้า	ผู้ใช้งานสามารถเลือกดูรายการร้านค้าได้ โดยไม่จำเป็นต้องเข้าสู่ระบบ
ดูรายการแนะนำเสื้อผ้า	ผู้ใช้งานสามารถเลือกดูเสื้อผ้าที่แนะนำจากร้านค้าได้ โดยจำเป็นต้องเข้าสู่ระบบก่อน โดยรายการเสื้อผ้าที่แนะนำจะดึงข้อมูลมาจากการรายการสินค้าโปรดของผู้ใช้งาน โดยจะแบ่งออกเป็น 2 กรณี คือ กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ และกรณีที่มีรายการเสื้อผ้าที่ถูกใจ
ดูรายการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ	การแนะนำเสื้อผ้าในกรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ จะแนะนำสินค้าตามรายการสินค้าที่มีคนถูกใจมากที่สุดหรือให้คะแนนสินค้ามากที่สุด โดยจะแนะนำมาทั้งหมด 9 รายการ
ดูรายการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ	การแนะนำเสื้อผ้าในกรณีที่มีรายการเสื้อผ้าที่ถูกใจ จะแนะนำสินค้าตามรายการที่ผู้ใช้งานถูกใจเสื้อผ้าหรือสินค้านั้น โดยจะแนะนำมาทั้งหมด 9 รายการ
บันทึกรายการเสื้อผ้าที่ถูกใจ	เมื่อผู้ใช้งานเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถบันทึกรายการเสื้อผ้าที่ถูกใจได้
บันทึกรายการร้านค้าที่ถูกใจ	เมื่อผู้ใช้งานเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถบันทึกรายการร้านค้าที่ถูกใจได้

ตารางที่ 3.3: อธิบาย Use Case หน้าที่ของระบบ (ต่อ) ในภาพที่ 3.18

Use Case	คำอธิบาย
ดูบันทึกรายการเสื้อผ้าที่ถูกใจ	เมื่อผู้ใช้งานเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถดูบันทึกรายการเสื้อผ้าที่ถูกใจได้
ดูบันทึกรายการร้านค้าที่ถูกใจ	เมื่อผู้ใช้งานเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถดูบันทึกรายการร้านค้าที่ถูกใจได้
ค้นหาเสื้อผ้า	ผู้ใช้งานสามารถค้นหารายชื่อเสื้อผ้าที่ต้อการได้ โดยไม่จำเป็นต้องเข้าสู่ระบบ
เพิ่มร้านค้า	เมื่อผู้ใช้ทำการสมัครสมาชิกและเข้าสู่ระบบแล้ว สามารถทำการเพิ่มร้านค้าได้
แก้ไขข้อมูลร้านค้า	เมื่อร้านค้าเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถแก้ข้อมูลร้านค้าได้
เพิ่ม ลบ แก้ไขข้อมูลเสื้อผ้าในร้านค้า	เมื่อร้านค้าเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถเพิ่มเสื้อผ้า ลบ และแก้ไขข้อมูลเสื้อมูลเสื้อผ้าในร้านค้าได้
ดูรายการคนที่ถูกใจเสื้อผ้า	เมื่อร้านค้าเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถดูรายการคนที่ถูกใจเสื้อผ้าได้
ดูรายการคนที่ถูกใจร้านค้า	เมื่อร้านค้าเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถดูรายการคนที่ถูกใจร้านค้าได้
เพิ่ม ลบ แก้ไขข้อมูลร้านค้า	เมื่อผู้ดูแลระบบเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถเพิ่ม ลบ แก้ไขข้อมูลร้านค้าได้
เพิ่ม ลบ แก้ไขข้อมูลผู้ใช้งาน	เมื่อผู้ดูแลระบบเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถเพิ่ม ลบ แก้ไขข้อมูลผู้ใช้งานได้

ตารางที่ 3.4: Use Case สมัครสมาชิก

Use Case Title : สมัครสมาชิก	Use case Id : 1
Primary Actor : ผู้ใช้งาน และร้านค้า	
Stakeholder Actor : -	
Main Flow : ผู้ใช้งานและร้านค้าสามารถสมัครสมาชิกเพื่อเข้าใช้งานระบบได้ จะต้องกรอกข้อมูลให้ครบถ้วนทุกช่อง	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตจะไม่สามารถสมัครสมาชิกได้	
Exceptional Flow ที่ 2 : หากผู้ใช้กรอกข้อมูลไม่ครบทุกช่องตามที่กำหนดจะไม่สามารถกดสมัครสมาชิกได้	
Exceptional Flow ที่ 3 : หากผู้ใช้กรอกชื่อเป็นภาษาไทยระบบจะไม่สามารถทำการสมัครสมาชิกได้ และชื่อผู้ใช้ที่กรอกเข้ามาจะต้องไม่ซ้ำในระบบถึงจะสามารถสมัครสมาชิกได้	

ตารางที่ 3.5: Use Case เข้าสู่ระบบ

Use Case Title : เข้าสู่ระบบ	Use case Id : 2
Primary Actor : ผู้ใช้งาน ร้านค้า และผู้ดูแลระบบ	
Stakeholder Actor : -	
Main Flow : ในส่วนนี้ผู้ใช้งานและผู้ดูแลระบบ สามารถใช้ username และ password ที่มีเพื่อลงชื่อเข้าใช้งานระบบได้	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตจะไม่สามารถสมัครสมาชิกได้	
Exceptional Flow ที่ 2 : หากผู้ใช้กรอก username และ password ไม่ถูกต้องจะไม่สามารถเข้าสู่ระบบได้	

ตารางที่ 3.6: Use Case แก้ไขข้อมูลใช้งาน

Use Case Title : แก้ไขข้อมูลใช้งาน	Use case Id : 3
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : ในส่วนของการแก้ไขข้อมูลผู้ใช้งาน เมื่อเข้าสู่ระบบเรียบร้อยแล้วจะสามารถทำการแก้ไขข้อมูลผู้ใช้งานได้ โดยสามารถแก้ไขข้อมูลได้ทั้งหมด	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถบันทึกข้อมูลผู้ใช้งาน	

ตารางที่ 3.7: Use Case ดูรายการเสื้อผ้าจากร้านค้า

Use Case Title : ดูรายการเสื้อผ้าจากร้านค้า	Use case Id : 4
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : ในส่วนของดูเสื้อผ้าจากร้านค้านี้ผู้ใช้งานสามารถกดดูรายการเสื้อผ้าในระบบได้ทั้งหมด โดยไม่จำเป็นต้องเข้าสู่ระบบ หรือเมื่อเข้าสู่ระบบแล้วก็สามารถดูเสื้อผ้าจากร้านค้าได้เช่นกัน	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตจะไม่สามารถดูรายการเสื้อผ้าจากร้านค้าได้	

ตารางที่ 3.8: Use Case ดูรายการร้านค้า

Use Case Title : ดูรายการร้านค้า	Use case Id : 5
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow :ในส่วนของดูร้านค้านี้ผู้ใช้งานสามารถกดดูรายการร้านค้าในระบบได้ทั้งหมด โดยไม่จำเป็นต้องเข้าสู่ระบบ หรือเมื่อเข้าสู่ระบบแล้วก็สามารถดูร้านค้าได้เช่นกัน	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตจะไม่สามารถดูรายการร้านค้าได้	

ตารางที่ 3.9: Use Case ดูรายการแนะนำเสื้อผ้า

Use Case Title : ดูรายการแนะนำเสื้อผ้า	Use case Id : 6
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบเรียบร้อย จะสามารถดูรายการแนะนำเสื้อผ้าได้ ผู้ใช้งานสามารถเลือกดูเสื้อผ้าที่แนะนำจากร้านค้าได้ โดยจำเป็นต้องเข้าสู่ระบบก่อน โดยรายการเสื้อผ้าที่แนะนำจะดึงข้อมูลมาจากการรายการสินค้าโปรดของผู้ใช้งาน โดยจะแบ่งออกเป็น 2 กรณี คือ กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ และกรณีที่มีรายการเสื้อผ้าที่ถูกใจ	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถดูรายการแนะนำเสื้อผ้าได้	

ตารางที่ 3.10: Use Case ดูรายการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ

Use Case Title : ดูรายการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ	Use case Id : 7
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบเรียบร้อย จะสามารถดูรายการแนะนำเสื้อผ้าได้ ซึ่งข้อมูลส่วนนี้จะมาจากรายการสินค้าโปรด การแนะนำเสื้อผ้าในกรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ จะแนะนำสินค้าตามรายการสินค้าที่มีคนถูกใจมากที่สุดหรือให้คะแนนสินค้ามากที่สุด โดยจะแนะนำมาทั้งหมด 9 รายการ	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถดูรายการแนะนำเสื้อผ้าได้	

ตารางที่ 3.11: Use Case ดูรายการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ

Use Case Title : ดูรายการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ	Use case Id : 8
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบเรียบร้อย จะสามารถดูรายการแนะนำเสื้อผ้าได้ ซึ่งข้อมูลส่วนนี้จะมาจากรายการสินค้าโปรด การแนะนำเสื้อผ้าในกรณีที่มีรายการเสื้อผ้าที่ถูกใจ จะแนะนำสินค้าตามรายการที่ผู้ใช้งานถูกใจเสื้อผ้าหรือสินค้านั้น โดยจะแนะนำมาทั้งหมด 9 รายการ	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถดูรายการแนะนำเสื้อผ้าได้	

ตารางที่ 3.12: Use Case บันทึกรายการเสื้อผ้าที่ถูกใจ

Use Case Title : บันทึกรายการเสื้อผ้าที่ถูกใจ	Use case Id : 9
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบแล้ว จะสามารถบันทึกรายการเสื้อผ้าที่ถูกใจได้ โดยกดปุ่มเพิ่มเป็นสินค้าโปรด ระบบก็จะทำการบันทึกข้อมูลรายการเสื้อผ้านั้น โดยในส่วนของการเพิ่มเสื้อผ้าเป็นสินค้าโปรดนั้นจะมีระดับให้สามารถเลือกได้ด้วยว่าชอบสินค้าในระดับใดใน 1-5	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถบันทึกรายการเสื้อผ้าได้	

ตารางที่ 3.13: Use Case บันทึกรายการร้านค้าที่ถูกใจ

Use Case Title : บันทึกรายการร้านค้าที่ถูกใจ	Use case Id : 10
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบแล้ว จะสามารถบันทึกรายการร้านค้าที่ถูกใจได้ โดยกดปุ่มเพิ่มเป็นร้านค้าโปรด ระบบก็จะทำการบันทึกข้อมูลรายการร้านค้านั้น	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถบันทึกรายการร้านค้าที่ถูกใจได้	

ตารางที่ 3.14: Use Case ดูบันทึกรายการเสื้อผ้าที่ถูกใจ

Use Case Title : ดูบันทึกรายการเสื้อผ้าที่ถูกใจ	Use case Id : 11
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบแล้ว จะสามารถดูบันทึกรายการเสื้อผ้าที่ถูกใจได้ โดยจะแสดงรายการเสื้อผ้าตามที่ผู้ใช้งานได้ทำการเพิ่ม และสามารถลบรายการสื้อผ้าที่ถูกใจได้เช่นกัน	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถดูบันทึกรายการเสื้อผ้าได้	

ตารางที่ 3.15: Use Case ดูบันทึกรายการร้านค้าที่ถูกใจ

Use Case Title : ดูบันทึกรายการร้านค้าที่ถูกใจ	Use case Id : 12
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ใช้งานสมัครสมาชิกและเข้าสู่ระบบแล้ว จะสามารถดูบันทึกรายการร้านค้าที่ถูกใจได้ โดยจะแสดงรายการร้านค้าตามที่ผู้ใช้งานได้ทำการเพิ่มและสามารถลบรายการร้านค้าที่ถูกใจได้เช่นกัน	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถดูบันทึกรายการร้านค้าที่ถูกใจได้	

ตารางที่ 3.16: Use Case ค้นหาเสื้อผ้า

Use Case Title : ค้นหาเสื้อผ้า	Use case Id : 13
Primary Actor : ผู้ใช้งาน	
Stakeholder Actor : -	
Main Flow : ในส่วนของค้นหาเสื้อผ้านี้ผู้ใช้งานสามารถค้นหาเสื้อผ้าหรือสินค้าที่ต้องการได้ โดยไม่จำเป็นต้องเข้าสู่ระบบ โดยระบบจะค้นหาเสื้อผ้าหรือสินค้าตามชื่อของสินค้าเท่านั้น	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ตจะไม่สามารถค้นหาสินค้าได้	
Exceptional Flow ที่ 2 : หากผู้ใช้กรอกข้อมูลที่ต้องการค้นหาเป็นสีเสื้อ หรือราคา เป็นต้น ระบบจะไม่สามารถทำการค้นหาได้	

ตารางที่ 3.17: Use Case เพิ่มร้านค้า

Use Case Title : เพิ่มร้านค้า	Use case Id : 14
Primary Actor : ร้านค้า	
Stakeholder Actor : -	
Main Flow : ในส่วนของการเพิ่มร้านค้า เมื่อผู้ใช้ทำการเข้าสู่ระบบเรียบร้อยแล้ว สามารถทำการเพิ่มร้านค้าได้ โดยการเพิ่มร้านค้านี้จะประกอบไปด้วยข้อมูลร้านค้าทั้งหมดและรายการเสื้อผ้าหรือสินค้าในร้านค้า	
Exceptional Flow ที่ 1 : หากร้านค้าไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถบันทึกข้อมูลร้านค้าได้	

ตารางที่ 3.18: Use Case แก้ไขข้อมูลร้านค้า

Use Case Title : แก้ไขข้อมูลร้านค้า	Use case Id : 15
Primary Actor : ร้านค้า	
Stakeholder Actor : -	
Main Flow : ในส่วนของการแก้ไขข้อมูลร้านค้า เมื่อร้านค้าทำการเข้าสู่ระบบเรียบร้อยแล้ว จะสามารถแก้ไขข้อมูลได้ โดยสามารถแก้ไขข้อมูลได้ทั้งหมด	
Exceptional Flow ที่ 1 : หากร้านค้าไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถแก้ไขข้อมูลร้านค้าได้	

ตารางที่ 3.19: Use Case เพิ่ม ลบ แก้ไขข้อมูลเสื้อผ้าในร้านค้า

Use Case Title : เพิ่ม ลบ แก้ไขข้อมูลเสื้อผ้าในร้านค้า	Use case Id : 16
Primary Actor : ร้านค้า	
Stakeholder Actor : -	
Main Flow : เมื่อร้านค้าเข้าสู่ระบบเรียบร้อยร้อย จะสามารถเพิ่มรายการเสื้อผ้า ลบ และแก้ไขข้อมูลเสื้อมูลเสื้อผ้าในร้านค้าได้	
Exceptional Flow ที่ 1 : หากร้านค้าไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถเพิ่มรายการเสื้อผ้า ลบ และแก้ไขข้อมูลเสื้อมูลเสื้อผ้าในร้านค้าได้	

ตารางที่ 3.20: Use Case ดูรายการการคนที่ถูกใจเสื้อผ้า

Use Case Title : ดูรายการการคนที่ถูกใจเสื้อผ้า	Use case Id : 17
Primary Actor : ร้านค้า	
Stakeholder Actor : -	
Main Flow : เมื่อร้านค้าสมัครสมาชิกและเข้าสู่ระบบแล้ว จะสามารถดูรายการการคนที่ถูกใจเสื้อผ้าได้	
Exceptional Flow ที่ 1 : หากร้านค้าไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถดูรายการการคนที่ถูกใจเสื้อผ้าได้	

ตารางที่ 3.21: Use Case ดูรายการการคนที่ถูกใจร้านค้า

Use Case Title : ดูรายการการคนที่ถูกใจร้านค้า	Use case Id : 18
Primary Actor : ร้านค้า	
Stakeholder Actor : -	
Main Flow : เมื่อร้านค้าสมัครสมาชิกและเข้าสู่ระบบแล้ว จะสามารถดูรายการการคนที่ถูกใจร้านค้าได้	
Exceptional Flow ที่ 1 : หากร้านค้าไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถดูรายการการคนที่ถูกใจร้านค้าได้	

ตารางที่ 3.22: Use Case เพิ่ม ลบ แก้ไขข้อมูลร้านค้า

Use Case Title : เพิ่ม ลบ แก้ไขข้อมูลร้านค้า	Use case Id : 19
Primary Actor : ผู้ดูแลระบบ	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ดูแลเข้าสู่ระบบแล้ว จะสามารถ เพิ่ม ลบ แก้ไขข้อมูลร้านค้าได้	
Exceptional Flow ที่ 1 : หากผู้ดูแลระบบไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถเพิ่ม ลบ แก้ไขข้อมูลร้านค้าได้	

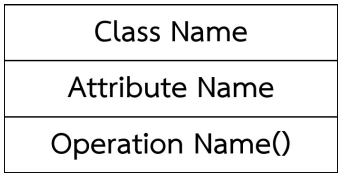
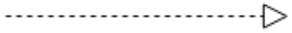
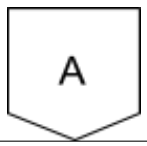
ตารางที่ 3.23: Use Case เพิ่ม ลบ แก้ไขข้อมูลผู้ใช้งาน

Use Case Title : เพิ่ม ลบ แก้ไขข้อมูลผู้ใช้งาน	Use case Id : 20
Primary Actor : ผู้ดูแลระบบ	
Stakeholder Actor : -	
Main Flow : เมื่อผู้ดูแลเข้าสู่ระบบแล้ว จะสามารถ เพิ่ม ลบ แก้ไขข้อมูลผู้ใช้งานได้	
Exceptional Flow ที่ 1 : หากผู้ดูแลระบบไม่เชื่อมต่ออินเทอร์เน็ตและยังไม่ได้เข้าสู่ระบบจะไม่สามารถเพิ่ม ลบ แก้ไขข้อมูลผู้ใช้งานได้	

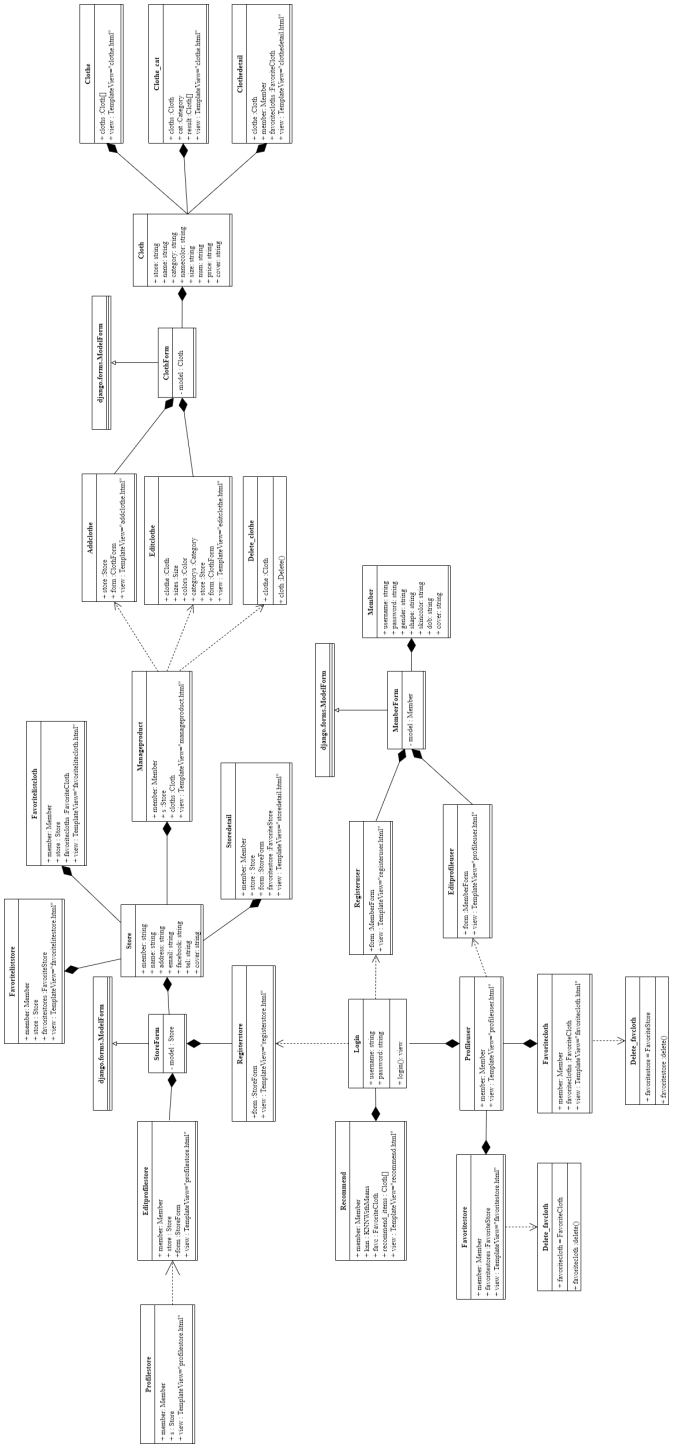
3.5 Class Diagram

Class Diagram คือแผนภาพที่ใช้แสดงคลาสและความสัมพันธ์ในแบบต่างๆ ระหว่างคลาส สัญลักษณ์ที่ใช้ในการเขียน Class Diagram แสดงในตารางที่ 3.24

ตารางที่ 3.24: สัญลักษณ์ของ Class Diagram

สัญลักษณ์	การใช้งาน
	คลาส สัญลักษณ์แทนด้วยสี่เหลี่ยมแบ่งเป็น 3 ส่วน ส่วนบน เป็นชื่อของ class ส่วนกลาง เป็นชื่อ Attribute และส่วนล่างเป็น Operation Name หรือ Method ใช้สำหรับเขียนฟังก์ชันในการทำงานของคลาสนั้น ๆ ชนิดของ Visibility ของ Method และ Attribute แบ่งเป็น 3 ชนิด ได้แก่ (a) Public แทนสัญลักษณ์ด้วยเครื่องหมายบวก (+) (b) Private แทนสัญลักษณ์ด้วยเครื่องหมายลบ (-) (c) Protected แทนสัญลักษณ์ด้วยเครื่องหมายชาร์ป ()
	Dependency Relationship หมายความว่า คลาสที่อยู่ฝั่งต้นลูกศรสามารถเรียกใช้คลาสที่อยู่ฝั่งหัวลูกศร
	Composition Relationship เป็นความสัมพันธ์ระหว่างออบเจ็กต์หรือคลาสแบบขึ้นต่อกันและมีความเกี่ยวข้องกันเสมอ
	Realization Relationship เป็นความสัมพันธ์ระหว่าง Object หรือ Class ในลักษณะของการสืบทอดคุณสมบัติจาก Class หนึ่ง (Super class) ไปยังอีก Class หนึ่ง (Subclass)
	Connector เป็นสัญลักษณ์แทนด้วยรูปห้าเหลี่ยมและมีชื่ออยู่ตรงกลาง จะสร้างสัญลักษณ์นี้ไว้เมื่อต้องการเชื่อมต่อคลาสที่อยู่คนละหน้า

Class Diagram แสดงความสัมพันธ์ในรูปแบบต่างๆ ระหว่างคลาสของแอปพลิเคชันระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี อธิบายได้ตามภาพที่ 3.19 ดังต่อไปนี้



รูปที่ 3.19: Class Diagram ระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี

จากรูปภาพที่ 3.19 สามารถอธิบายแผนภาพ Class Diagram ได้ดังนี้

ตารางที่ 3.25: อธิบาย Class Diagram ของคลาสพื้นฐานของระบบ

Class Diagram	คำอธิบาย
Login	คลาส Login เป็นคลาสที่ใช้เพื่อให้สมาชิกที่ได้ลงทะเบียนกับระบบเข้าสู่ระบบเพื่อใช้งานบริการจากระบบ ซึ่งสามารถเรียกใช้คลาส Registeruser Registerstore Recommend และคลาส Profileuser
Registeruser	คลาส Registeruser เป็นคลาสที่ใช้แสดงหน้าสมัครสมาชิกเพื่อให้ผู้ใช้ที่ต้องการลงทะเบียนเข้าสู่ระบบสามารถใช้งานได้ ซึ่งสามารถเรียกใช้คลาส MemberForm
MemberForm	คลาส MemberForm เป็น คลาส ที่ ใช้ ใน การ สร้าง ฐาน ข้อมูล ของ ผู้ ใช้ งาน จะ มี คลาส หลัก คือ django.forms.ModelForm ซึ่งสามารถเรียกใช้คลาส Member
Member	คลาส Member เป็นคลาสที่ใช้ในการเก็บข้อมูลผู้ใช้งาน
Registerstore	คลาส Registerstore เป็นคลาสที่ใช้แสดงหน้าสมัครสมาชิกร้านค้าเพื่อให้ผู้ใช้ที่ต้องการลงทะเบียนร้านค้าสามารถลงทะเบียนและมีร้านค้าได้ ซึ่งสามารถเรียกใช้คลาส StoreForm
StoreForm	คลาส StoreForm เป็น คลาส ที่ ใช้ ใน การ สร้าง ฐาน ข้อมูล ของ ร้าน ค้า จะ มี คลาส หลัก คือ django.forms.ModelForm ซึ่งสามารถเรียกใช้คลาส Store
Store	คลาส Store เป็นคลาสที่ใช้ในการเก็บข้อมูลร้านค้า
Recommend	คลาส Recommend เป็นคลาสที่ใช้แสดงรายการแนะนำเสื้อผ้าให้แก่ผู้ใช้งานหลังจากได้เข้าสู่ระบบแล้ว
Profileuser	คลาส Profileuser เป็นคลาสที่ใช้แสดงข้อมูลของผู้ใช้งานทั้งหมดหลังจากได้ทำการสมัครสมาชิกและเข้าสู่ระบบเรียบร้อยแล้ว ซึ่งสามารถเรียกใช้คลาส Editprofileuser Favoriteclothadd และคลาส Favoritestoreadd

ตารางที่ 3.26: อธิบาย Class Diagram ของคลาสพื้นฐานของระบบ (ต่อ)

Class Diagram	คำอธิบาย
Editprofileuser	คลาส Login เป็นคลาสที่ใช้เพื่อให้สมาชิกที่ได้ลงทะเบียคลาส Editprofileuser เป็นคลาสที่ใช้แสดงหน้ารายละเอียดข้อมูลของผู้ใช้งานทั้งหมดที่สามารถทำการแก้ไขข้อมูลได้ ซึ่งสามารถเรียกใช้คลาส MemberForm
Favoriteclothadd	คลาส Favoriteclothadd เป็นคลาสที่ใช้แสดงหน้ารายการสินค้าโปรดที่ผู้ใช้งานได้กดถูกใจ
Favoritestoreadd	คลาส Favoritestoreadd เป็นคลาสที่จะแสดงหน้ารายการร้านโปรดที่ผู้ใช้งานได้กดถูกใจ
Profilestore	คลาส Profilestore เป็นคลาสที่แสดงข้อมูลของร้านค้าทั้งหมดหลังจากได้ทำการสมัครสมาชิกและเข้าสู่ระบบเรียบร้อยแล้ว ซึ่งสามารถเรียกใช้คลาส Editprofilestore
Editprofilestore	คลาส Editprofilestore เป็นคลาสที่ใช้แสดงหน้ารายละเอียดข้อมูลของร้านค้าทั้งหมดที่สามารถทำการแก้ไขข้อมูลได้ ซึ่งสามารถเรียกใช้คลาส StoreForm
Store	คลาส Store เป็นคลาสที่ใช้แสดงรายการร้านค้าที่มีทั้งหมดในระบบ สามารถเรียกใช้คลาส Storedetail Favoriteliststore และคลาส Favoritelistcloth
Storedetail	คลาส Storedetail เป็นคลาสที่ใช้แสดงรายละเอียดของร้านค้า และรายการสินค้าที่มีในร้าน ซึ่งสามารถเรียกใช้คลาส Favoritestoreadd
Favoriteliststore	คลาส Favoriteliststore เป็นคลาสที่ใช้แสดงรายการคนที่กดถูกใจร้านค้า
Favoritelistcloth	คลาส Favoritelistcloth เป็นคลาสที่ใช้แสดงรายการคนที่กดถูกใจสินค้า
Manageproduct	คลาส Manageproduct เป็นคลาสที่ใช้แสดงหน้าตารางเพื่อจัดการรายการสินค้าในการเพิ่ม ลบ และแก้ไขสินค้า ซึ่งสามารถเรียกใช้คลาส Addclothe Editclothe และ Delete-clothe

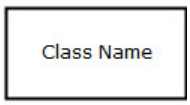
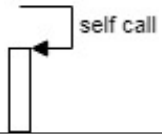
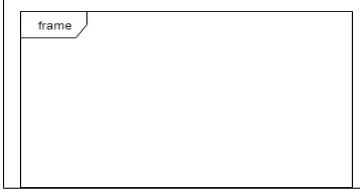
ตารางที่ 3.27: อธิบาย Class Diagram ของคลาสพื้นฐานของระบบ (ต่อ)

Class Diagram	คำอธิบาย
Addclothe	คลาส Addclothe เป็นคลาสที่ใช้แสดงหน้าเพิ่มสินค้า ซึ่งสามารถเรียกใช้คลาส ClothForm
Editclothe	คลาส Editclothe เป็นคลาสที่ใช้แสดงหน้ารายละเอียดข้อมูลของสินค้าที่สามารถทำการแก้ไขข้อมูลได้ ซึ่งสามารถเรียกใช้คลาส ClothForm
ClothForm	คลาส ClothForm เป็น คลาส ที่ ใช้ ใน การ สร้าง ฐาน ข้อมูล ของ เสื้อผ้า จะ มี คลาส หลัก คือ django.forms.ModelForm ซึ่งสามารถเรียกใช้คลาส Cloth
Delete-clothe	คลาส Delete-clothe เป็นคลาสที่ใช้ในการลบรายการเสื้อผ้าในร้านค้า
Clothe	คลาส Clothe เป็นคลาสที่ใช้แสดงรายการเสื้อผ้าที่มีทั้งหมดในระบบ ซึ่งสามารถเรียกใช้คลาส Clothedetail และ Clothe-cat
Clothedetail	คลาส Clothedetail เป็นคลาสที่ใช้แสดงรายละเอียดของเสื้อผ้า
Clothe-cat	คลาส Clothe-cat เป็นคลาสที่ใช้แสดงรายการเสื้อผ้าแบ่งตามประเภททั้งหมด 6 ประเภท ได้แก่ เสื้อ กางเกง กระโปรง สูท เดรส และอื่นๆ

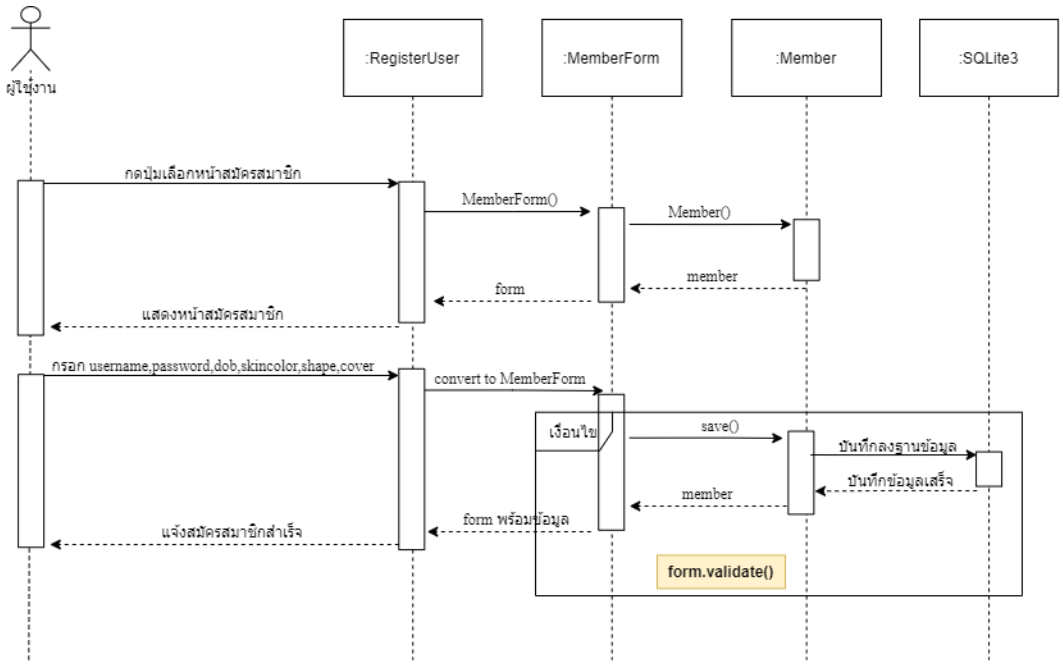
3.6 Sequence Diagram

Sequence Diagram เป็น Diagram ที่แสดงขั้นตอนการทำงานของแต่ละ Use Case ระหว่าง Object ต่างๆ ที่ส่งข้อความถึงกันและกัน โดย Sequence Diagram จะช่วยให้มองเห็นการทำงานของภาพรวมของระบบ ส่วนประกอบสัญลักษณ์ที่ใช้ในการเขียน Sequence Diagram แสดงดังตารางที่ 3.28

ตารางที่ 3.28: สัญลักษณ์ของ Sequence Diagram

สัญลักษณ์	การใช้งาน
	Class แสดงถึงการทำงานของ Use Case ในการส่งหรือรับข้อความ แทนด้วยสัญลักษณ์สี่เหลี่ยมมีชื่อคลาสอยู่ภายใน
	Lifeline หรือเส้นอายุขัย แสดงช่วงเวลาตั้งแต่เริ่มสร้าง object ในคลาสนั้น จนกระทั่ง object นั้นถูกทำลาย สัญลักษณ์แทนด้วยเส้นประ
	Focus of control หรือจุดควบคุม เป็นจุดควบคุมที่ object ใช้ทำการส่งหรือรับข้อความ สัญลักษณ์แทนด้วยสี่เหลี่ยม
	Message คือ ข้อความที่รับส่งระหว่าง Object สัญลักษณ์แทนด้วยลูกศรและประกอบด้วย 2 ส่วน คือ ข้อมูล (Data) และฟังก์ชัน (Function)
	Return Message เป็นข้อมูลที่ส่งกลับหลังจากทำงานเสร็จ
	Self call เป็นการเรียกฟังชันก์การทำงานภายในตัวเอง
	สร้างกรอบการทำงานของโปรแกรม เพื่อให้รู้ขอบเขตของการทำงานเช่น ลูป(loop)

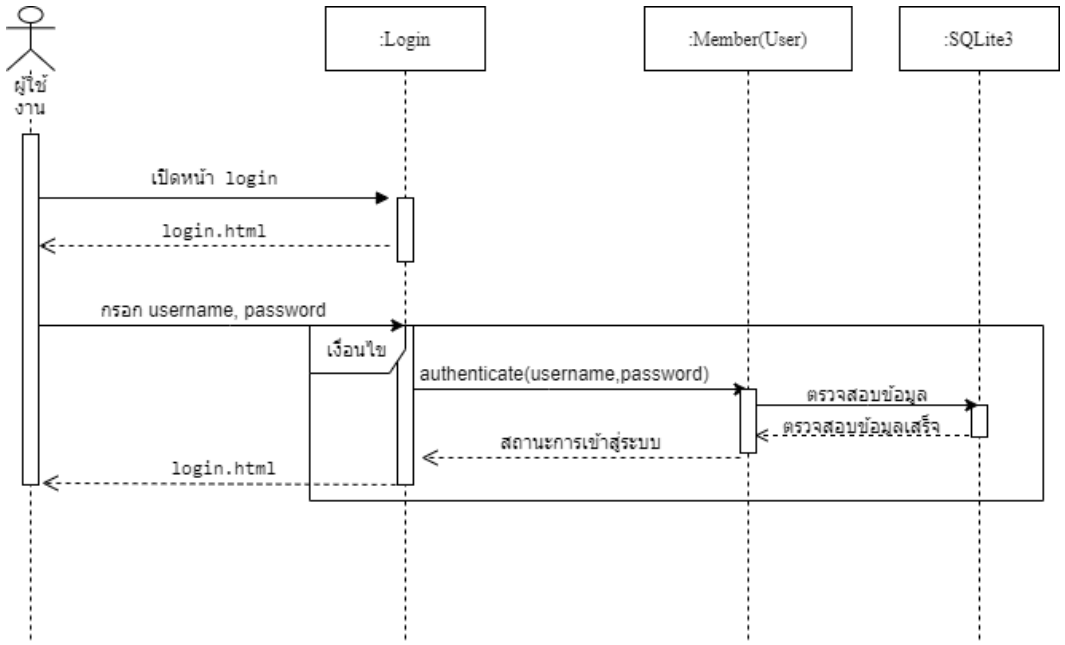
3.6.1 Sequence Diagram ของการสมัครสมาชิก



รูปที่ 3.20: Sequence Diagram ของการสมัครสมาชิก

จากภาพที่ 3.20 สามารถอธิบายแผนภาพ Sequence Diagram ของการสมัครสมาชิก ได้ดังนี้ เมื่อผู้ใช้กดสมัครสมาชิกระบบจะไปเรียกใช้ฟังก์ชัน RegisterUser() โดยจะแสดงหน้าสมัครสมาชิกส่งข้อความไปเรียกข้อมูลใน MemberForm() และส่งค่าข้อมูลทั้งหมดของสมาชิกมาในรูปแบบ form เมื่อผู้ใช้กรอกชื่อ รหัสผาน อีเมล์ เลือกเพศ เลือกวัน/เดือน/ปีเกิด เลือกรูปร่าง เลือกสีผิว เลือกรูปโปรไฟลเสร็จและกดปุมสมัครสมาชิกระบบจะไปเรียกฟังก์ชัน save() ซึ่งจะทำการเช็คเงื่อนไข ว่ากรอกข้อมูลครบทุกช่องหรือไม่ ถ้าครบแล้วจะส่งข้อมูล user ไปตรวจสอบในฐานข้อมูล SQLite3 และจะทำการสร้าง user เสร็จจะส่งสถานะการสมัครสมาชิกไปกลับไปให้คลาส Registeruser() และจะแสดงข้อมูลของผู้ใช้ถือว่าสมัครสมาชิกสำเร็จ หรือในกรณีกรอกข้อมูลไม่ครบทุกช่องตามที่กำหนดระบบจะแจ้งให้กรอกช่องที่ยังไม่ได้ทำการกรอก และทำการแจ้งสมัครสมาชิกสำเร็จ

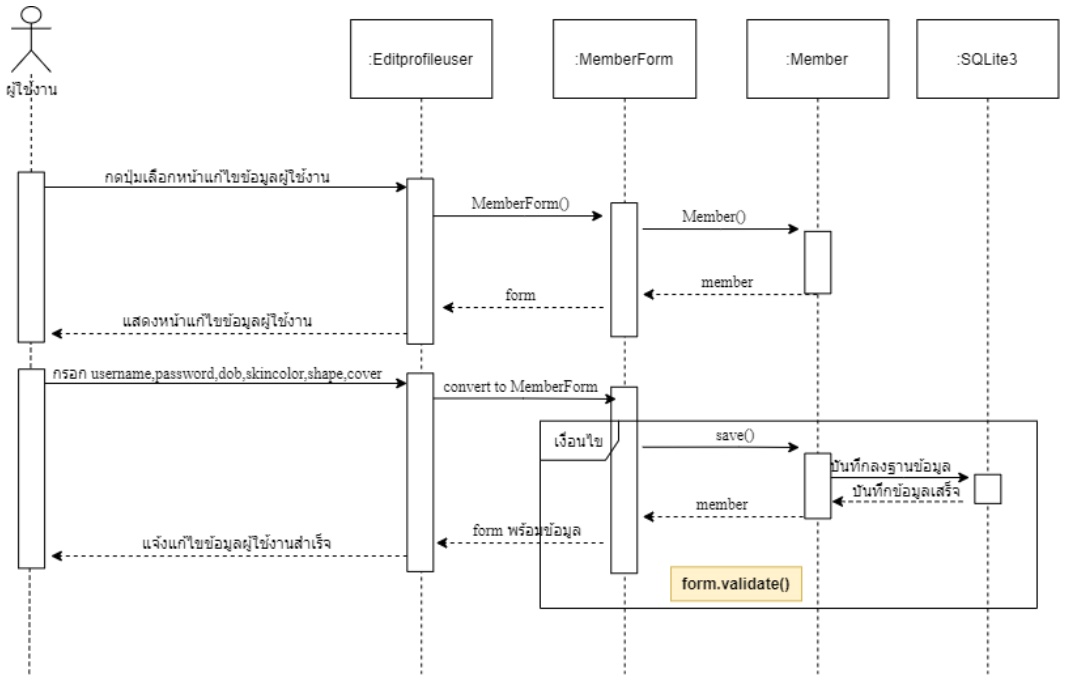
3.6.2 Sequence Diagram ของการเข้าสู่ระบบ



รูปที่ 3.21: Sequence Diagram ของการเข้าสู่ระบบ

จากภาพที่ 3.21 สามารถอธิบายแผนภาพ Sequence Diagram ของการเข้าสู่ระบบ ได้ดังนี้ เมื่อผู้ใช้งานกดปุ่มเข้าสู่ระบบ จากนั้นระบบจะเรียกใช้ฟังก์ชัน Login() โดยฟังก์ชัน Login() จะไปเรียกใช้ฟังก์ชัน Login() มาแสดงหน้าเข้าสู่ระบบต่อมาผู้ใช้กรอกชื่อผู้ใช้และรหัสผ่านเสร็จกดเข้าสู่ระบบทำให้ระบบไป เรียกใช้ฟงกชัน Login() โดยจะส่งข้อมูลชื่อผู้ใช้และรหัสผ่านไปตรวจสอบในส่วน security และ SQLite3 ทำการค้นหาและส่งสถานะการเข้าสู่ระบบกลับไปที่คลาส Login ซึ่งในคลาสนี้จะทำการตรวจสอบสถานะการเข้าสู่ระบบและจะแสดงหน้าหลักของระหากกรอกชื่อผู้ใช้และรหัสผ่านถูกต้อง แต่ถ้าหากกรอกไม่ถูกต้องระบบก็จะไม่ไปที่หน้าหลัก

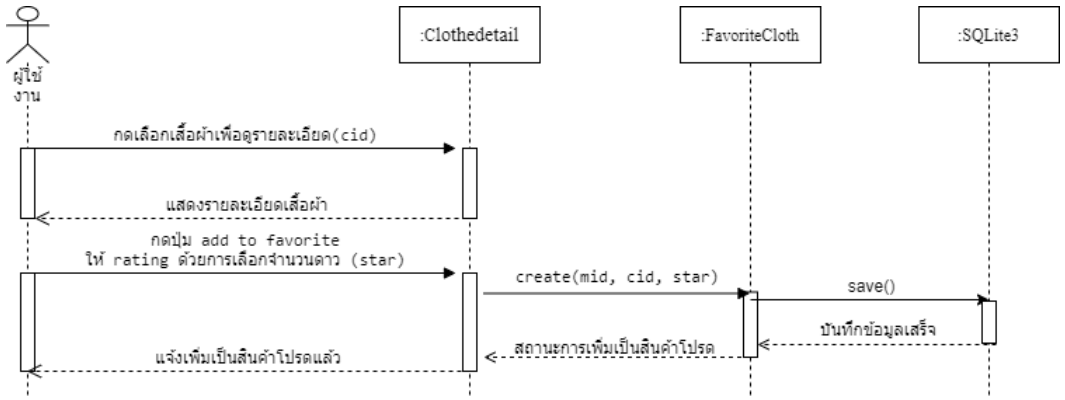
3.6.3 Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน



รูปที่ 3.22: Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน

จากภาพที่ 3.22 สามารถอธิบายแผนภาพ Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน ได้ดังนี้ เมื่อผู้ใช้กดเลือกหน้าแก้ไขข้อมูลผู้ใช้งาน ระบบจะเรียกใช้ฟังก์ชัน Editprofileuser() ส่งข้อความไปเรียกข้อมูลใน MemberForm() และส่งค่าข้อมูลทั้งหมดของสมาชิกมาในรูปแบบ form จากนั้นเมื่อผู้ใช้ทำการกรอกชื่อผู้ใช้ รหัสผ่าน วัน/เดือน/ปีเกิด สีผิว รูปร่าง เลือกรูปโปร์ไฟล์ และทำการแก้ไขข้อมูลผู้ใช้งาน ระบบจะทำการส่งข้อมูลทั้งหมดที่ผู้ใช้กรอกครบไปยังฐานข้อมูล SQLite3 ทำการบันทึกลงฐานข้อมูล โดยมีเงื่อนไขโดยต้องกรอกข้อมูลชื่อเป็นภาษาอังกฤษและต้องไม่ซ้ำกับฐานข้อมูล กรอกให้ครบทุกช่องก็จะสามารถทำการกดแก้ไขข้อมูลผู้ใช้งานได้ และทำการแจ้งแก้ไขข้อมูลผู้ใช้งานสำเร็จ

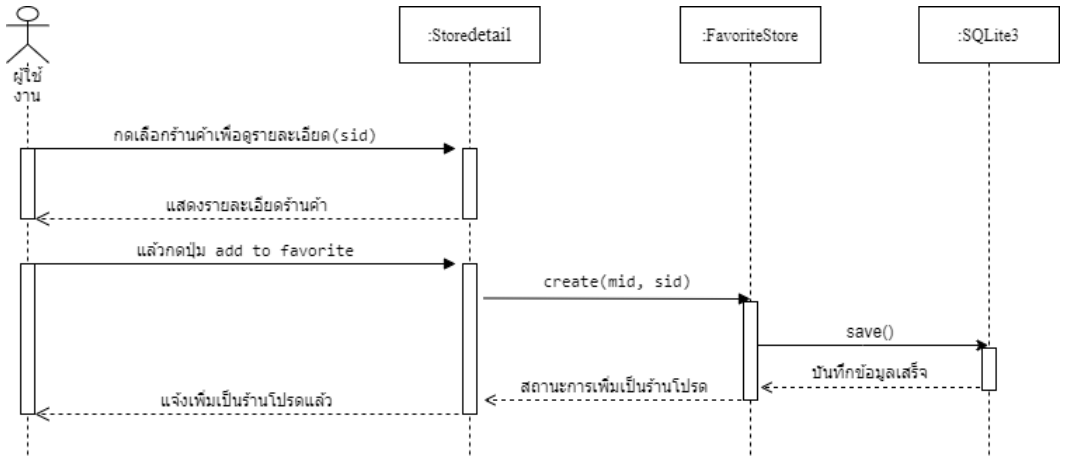
3.6.4 Sequence Diagram ของบันทึกรายการเสื้อผ้าที่ถูกใจ



รูปที่ 3.23: Sequence Diagram ของบันทึกรายการเสื้อผ้าที่ถูกใจ

จากภาพที่ 3.23 สามารถอธิบายแผนภาพ Sequence Diagram ของบันทึกรายการเสื้อผ้าที่ถูกใจ ได้ดังนี้ เมื่อผู้ใช้ต้องการกดบันทึกรายการเสื้อผ้าที่ถูกใจ ระบบจะเรียกใช้ฟังก์ชัน Clothdetail() และแสดงหน้ารายละเอียดเสื้อผ้าที่ผู้ใช้สนใจ เมื่อผู้ใช้ทำการกดปุ่มเพิ่มเป็นสินค้าโปรด พร้อมให้คะแนนระดับความชอบสิ้นค้าชิ้นนั้น ระบบจะส่งข้อมูลหมายเลยผู้ใช้งาน (mid) รหัสสินค้า (cid) และคะแนนความชอบสินค้า (star) ไปบันทึกลงในฐานข้อมูล SQLite3 และทำการแจ้งเตือนสถานะการเพิ่มเป็นสินค้าโปรดแล้ว

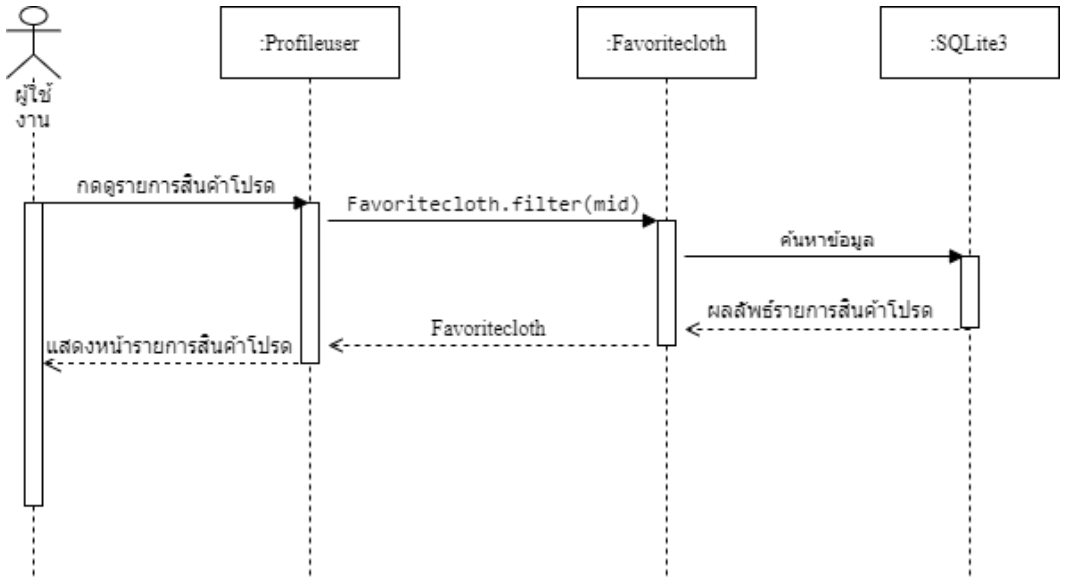
3.6.5 Sequence Diagram ของบันทึกรายการร้านค้าที่ถูกใจ



รูปที่ 3.24: Sequence Diagram ของบันทึกรายการร้านค้าที่ถูกใจ

จากภาพที่ 3.24 สามารถอธิบายแผนภาพ Sequence Diagram ของบันทึกรายการร้านค้าที่ถูกใจ ได้ดังนี้ เมื่อผู้ใช้ต้องการกดบันทึกรายการร้านค้าที่ถูกใจ ระบบจะเรียกใช้ฟังก์ชัน Storedetail() และแสดงหน้ารายละเอียดร้านค้าที่ผู้ใช้สนใจ เมื่อผู้ใช้ทำการกดปุ่มเพิ่มร้านโปรด ระบบจะส่งข้อมูลหมายเลยผู้ใช้งาน (mid) และรหัสร้านค้า (sid) ไปบันทึกลงในฐานข้อมูล SQLite3 และทำการแจ้งเตือนสถานะการเพิ่มเป็นร้านโปรดแล้ว

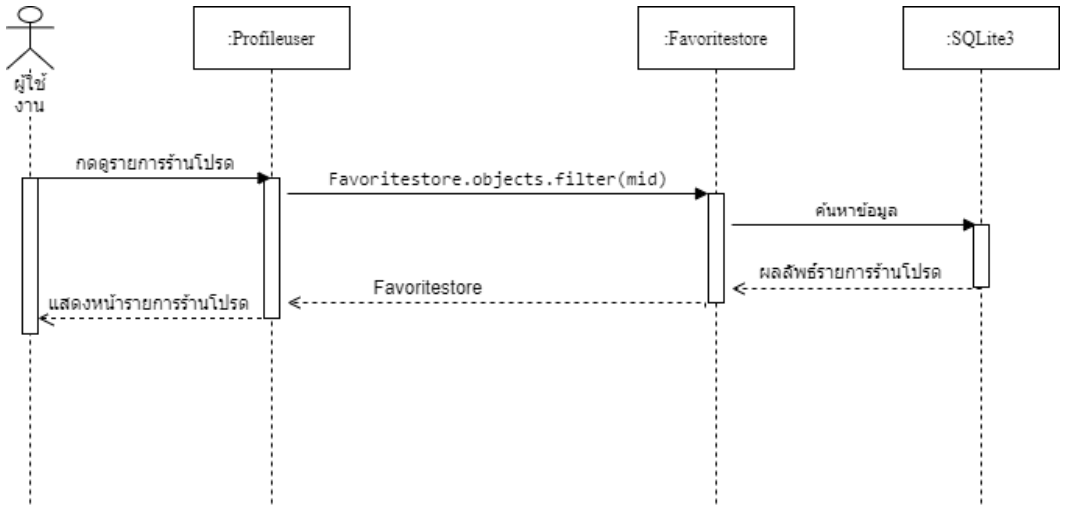
3.6.6 Sequence Diagram ของดูบันทึกรายการเสื้อผ้าที่ถูกใจ



รูปที่ 3.25: Sequence Diagram ของดูบันทึกรายการเสื้อผ้าที่ถูกใจ

จากภาพที่ 3.25 สามารถอธิบายแผนภาพ Sequence Diagram ของดูบันทึกรายการเสื้อผ้าที่ถูกใจ ได้ดังนี้ เมื่อผู้ใช้ต้องการกดดูบันทึกรายการเสื้อผ้าที่ถูกใจ ระบบจะเรียกใช้ฟังก์ชัน Profileuser() และแสดงหน้าข้อมูลผู้ใช้งาน เมื่อผู้ใช้ทำการกดปุ่มดูรายการสินค้าโปรด ระบบจะเรียกใช้ฟังก์ชัน Favoritecloth() โดยส่งรหัสผู้ใช้งาน (mid) ไปค้นหาในฐานข้อมูล SQLite3 และแสดงผลลัพธ์รายการสินค้าโปรดตามรายการที่ผู้ใช้ได้ทำการบันทึก

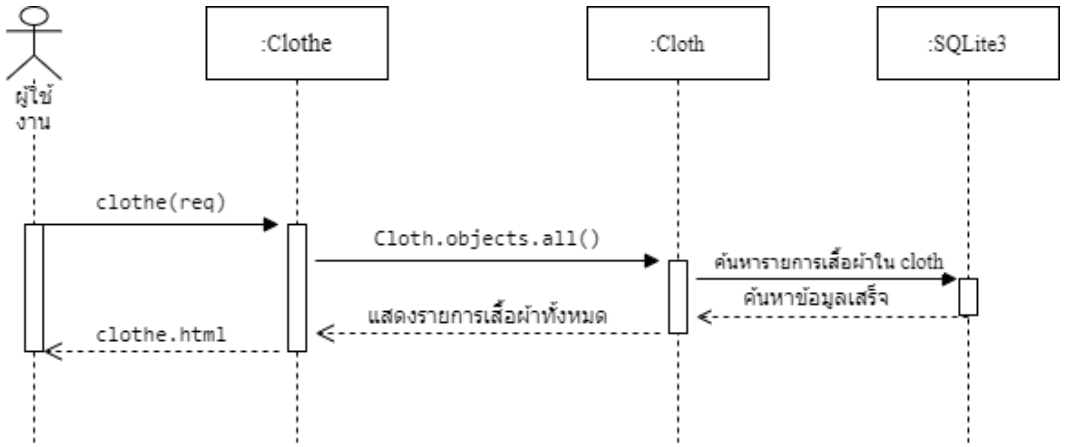
3.6.7 Sequence Diagram ของดูบันทึกรายการร้านค้าที่ถูกใจ



รูปที่ 3.26: Sequence Diagram ของดูบันทึกรายการร้านค้าที่ถูกใจ

จากภาพที่ 3.26 สามารถอธิบายแผนภาพ Sequence Diagram ของดูบันทึกรายการร้านค้าที่ถูกใจ ได้ดังนี้ เมื่อผู้ใช้ต้องการกดดูบันทึกรายการร้านค้าที่ถูกใจ ระบบจะเรียกใช้ฟังก์ชัน Profileuser() และแสดงหน้าข้อมูลผู้ใช้งาน เมื่อผู้ใช้ทำการกดปุ่มดูรายการสินค้าโปรด ระบบจะเรียกใช้ฟังก์ชัน Favoritestore() โดยส่งรหัสผู้ใช้งาน (mid) ไปค้นหาในฐานข้อมูล SQLite3 และแสดงผลลัพธ์รายการร้านโปรดตามรายการที่ผู้ใช้ได้ทำการบันทึก

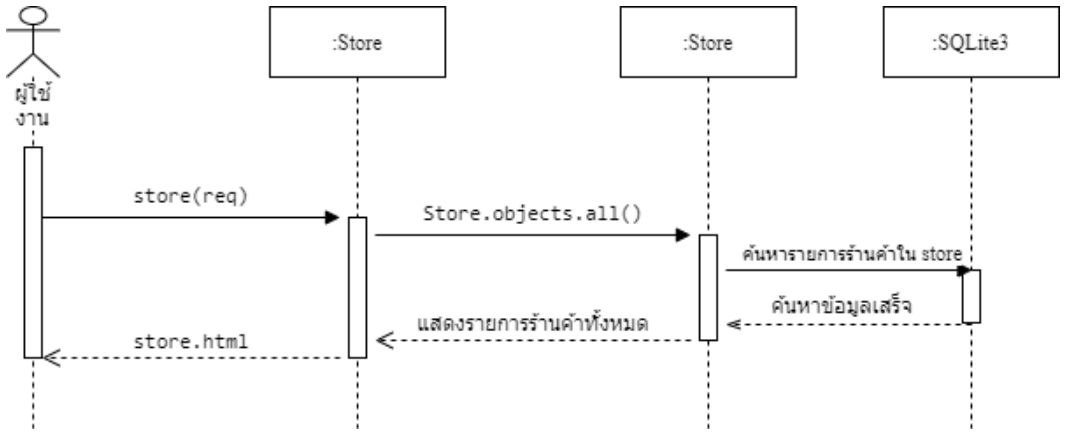
3.6.8 Sequence Diagram ของดูรายการเสื้อผ้าจากร้านค้า



รูปที่ 3.27: Sequence Diagram ของดูรายการเสื้อผ้าจากร้านค้า

จากภาพที่ 3.27 สามารถอธิบายแผนภาพ Sequence Diagram ของดูรายการเสื้อผ้าจากร้านค้า ได้ดังนี้ เมื่อผู้ใช้กดดูรายการเสื้อผ้าจากร้านค้า ระบบจะทำการเรียกใช้ฟังก์ชันก์ Clothe() โดยจะไปทำการดึงข้อมูลมาจากฐานข้อมูลตารางเสื้อผ้า Cloth จากฐานข้อมูล SQLite3 เมื่อค้นหาข้อมูลเสร็จระบบจะแสดงรายการเสื้อผ้าทั้งหมดไปยังหน้าเว็บ clothe.html

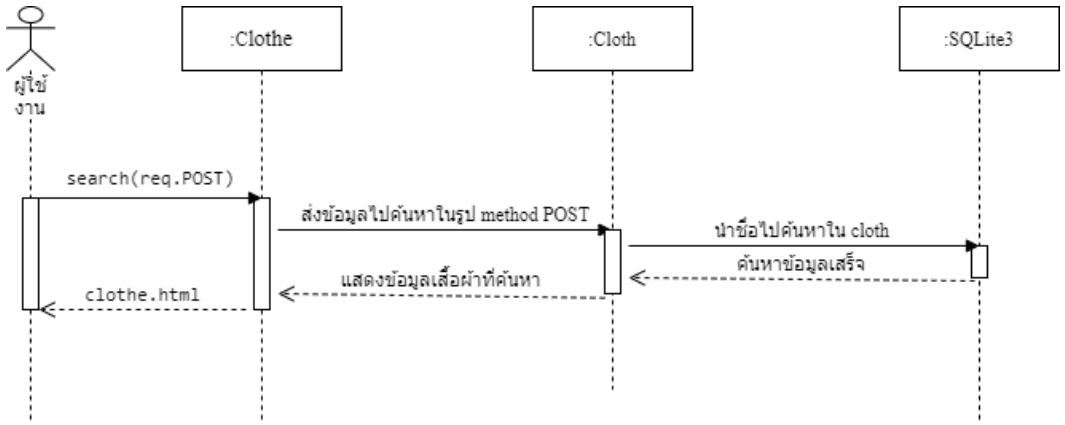
3.6.9 Sequence Diagram ของดูรายการร้านค้า



รูปที่ 3.28: Sequence Diagram ของดูรายการร้านค้า

จากภาพที่ 3.28 สามารถอธิบายแผนภาพ Sequence Diagram ของดูรายการร้านค้า ได้ดังนี้ เมื่อผู้ใช้กดดูรายการร้านค้า ระบบจะทำการเรียกใช้ฟังก์ชันก์ Store() โดยจะไปทำการดึงข้อมูลมาจากฐานข้อมูลตารางร้านค้า Store จากฐานข้อมูล SQLite3 เมื่อค้นหาข้อมูลเสร็จ ระบบจะแสดงรายการร้านค้าทั้งหมดไปยังหน้าเว็บ store.html

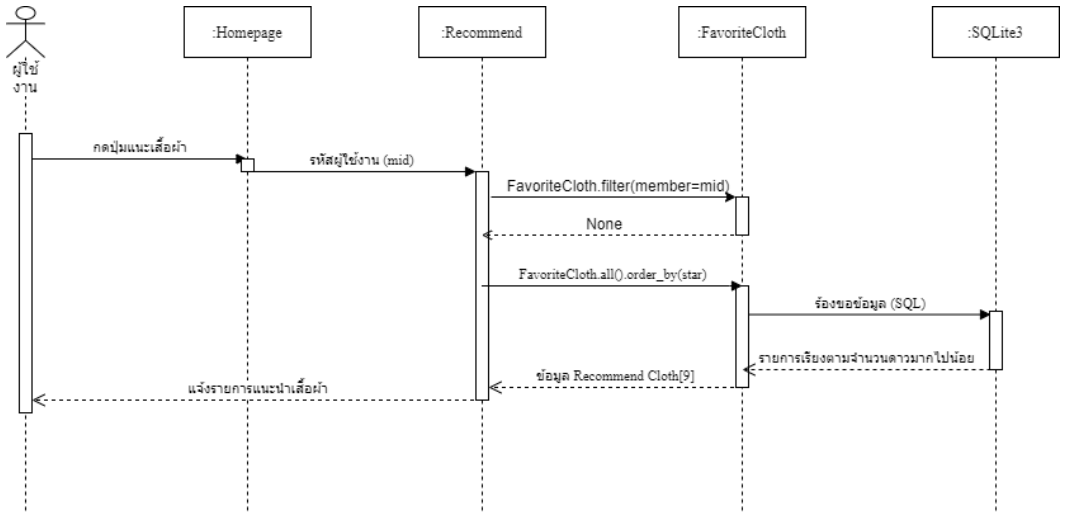
3.6.10 Sequence Diagram ของค้นหาเสื้อผ้า



รูปที่ 3.29: Sequence Diagram ของค้นหาเสื้อผ้า

จากภาพที่ 3.29 สามารถอธิบายแผนภาพ Sequence Diagram ของค้นหาเสื้อผ้า ได้ดังนี้ เมื่อผู้ใช้กดค้นหาเสื้อผ้า ระบบจะทำการเรียกใช้ฟังก์ชันก์ Clothe() โดยจะไปทำการดึงข้อมูลมาจากฐานข้อมูลตาราง Cloth จากฐานข้อมูล SQLite3 เมื่อเมื่อผู้ใช้ทำการกรอกชื่อสินค้าที่ต้องการค้นหาเสร็จ ระบบจะส่งข้อมูลไปในรูปแบบ method POST นำชื่อสินค้าที่ผู้ใช้กรอกไปค้นหาในตารางเสื้อผ้า Cloth ในฐานข้อมูล SQQLite3 เมื่อค้นหาข้อมูลเสร็จ ระบบจะแสดงข้อมูลเสื้อผ้าที่ผู้ใช้ต้องการไปยังหน้าเว็บ clothe.html

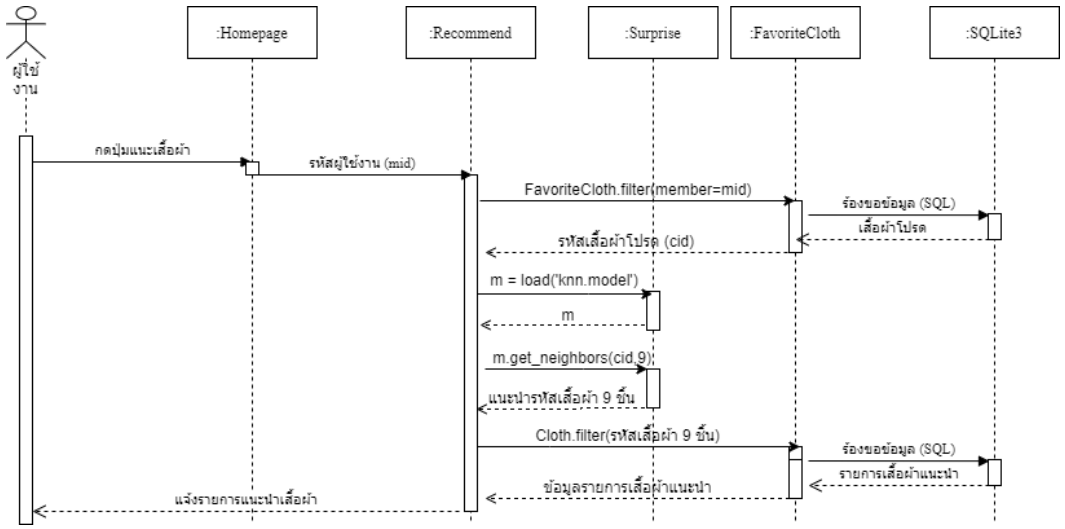
3.6.11 Sequence Diagram ของการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ



รูปที่ 3.30: Sequence Diagram ของการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ

จากภาพที่ 3.30 สามารถอธิบายแผนภาพ Sequence Diagram ของการแนะนำเสื้อผ้า กรณียังไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ ได้ดังนี้ เมื่อผู้ใช้เข้าสู่หน้าหลักของระบบและทำการกดปุ่มแนะนำเสื้อผ้า ระบบจะทำการส่งรหัสผู้ใช้ (mid) ไปเพื่อทำการค้นหาว่ารหัสผู้ใช้ที่ทำการเข้าสู่ระบบมาแล้วนั้นมีรายการบันทึกเสื้อผ้าที่ถูกใจหรือไม่ Recommend() เมื่อทำการตรวจสอบ FavoriteCloth() ไม่พบรายการบันทึกเสื้อผ้าที่ถูกใจระบบจะส่งค่าเป็น None และจากนั้นระบบจะทำการส่งข้อมูล FavoriteCloth.all.order-by(star) ส่งคำร้องขอไปยังฐานข้อมูล SQLite3 เพื่อให้ระบบทำการแนะนำรายการเสื้อผ้าที่มีการให้คะแนนหรือดาวมากที่สุด และแสดงรายการเรียงลำดับตามจำนวนคะแนนหรือดาวจากมากไปหาน้อย โดยข้อมูลที่นำมาแสดงจะมีทั้งหมด 9 ชิ้น Recommend Cloth[9] แสดงรายการแนะนำเสื้อผ้า Recommend() ให้แก่ผู้ใช้งานในกรณีที่ไม่มีเพิ่มรายการเสื้อผ้าที่ถูกใจ

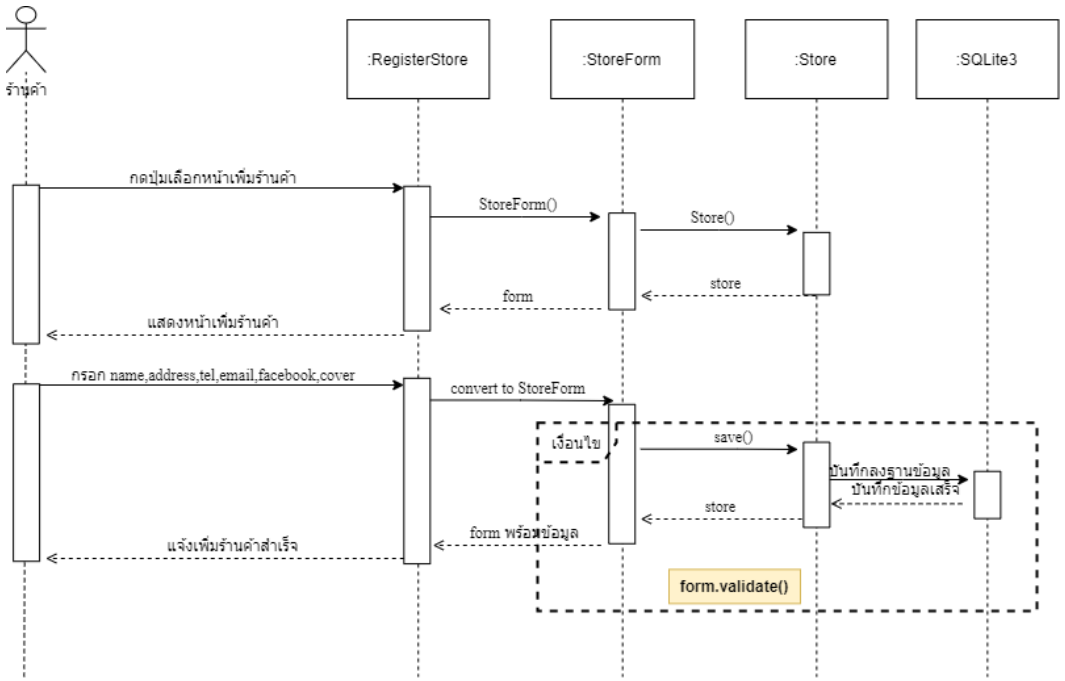
3.6.12 Sequence Diagram ของการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ



รูปที่ 3.31: Sequence Diagram ของการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ

จากภาพที่ 3.31 สามารถอธิบายแผนภาพ Sequence Diagram ของการแนะนำเสื้อผ้า กรณีที่มีรายการเสื้อผ้าที่ถูกใจ ได้ดังนี้ เมื่อผู้ใช้เข้าสู่หน้าหลักของระบบและทำการกดปุ่มแนะนำเสื้อผ้า ระบบจะทำการส่งรหัสผู้ใช้ (mid) ไปเพื่อทำการค้นหาว่ารหัสผู้ใช้ที่ทำการเข้าสู่ระบบมาแล้วนั้นมีรายการบันทึกเสื้อผ้าที่ถูกใจหรือไม่ Recommend() เมื่อทำการตรวจสอบ FavoriteCloth() มีรายการเสื้อผ้าที่ถูกใจ ระบบจะทำการร้องขอข้อมูลไปยังฐานข้อมูล SQLite3 แสดงรายการสินค้าโปรดของผู้ใช้ และนำรหัสสินค้าโปรด (cid) ไปใช้ในการคำนวณเพื่อทำนายการแนะนำเสื้อผ้า Surprise() เมื่อทำการโหลดข้อมูลเสร็จแล้วระบบจะส่งค่า (cid) ได้แนะนำรหัสสินค้า 9 ชิ้น มายังฟังก์ชันก์ Recommemend() และ Recomend() ได้ทำการส่งรหัสสินค้า 9 ชิ้น ไปร้องขอข้อมูลยังฐานข้อมูล SQLite3 และได้ข้อมูลรายการเสื้อผ้าแนะนำมาจาก FavoriteCloth() นำผลมาแสดงยัง Recommend() เป็นรายการแนะนำเสื้อผ้าให้แก่ผู้ใช้งานในกรณีที่มีรายการเสื้อผ้าที่ถูกใจ

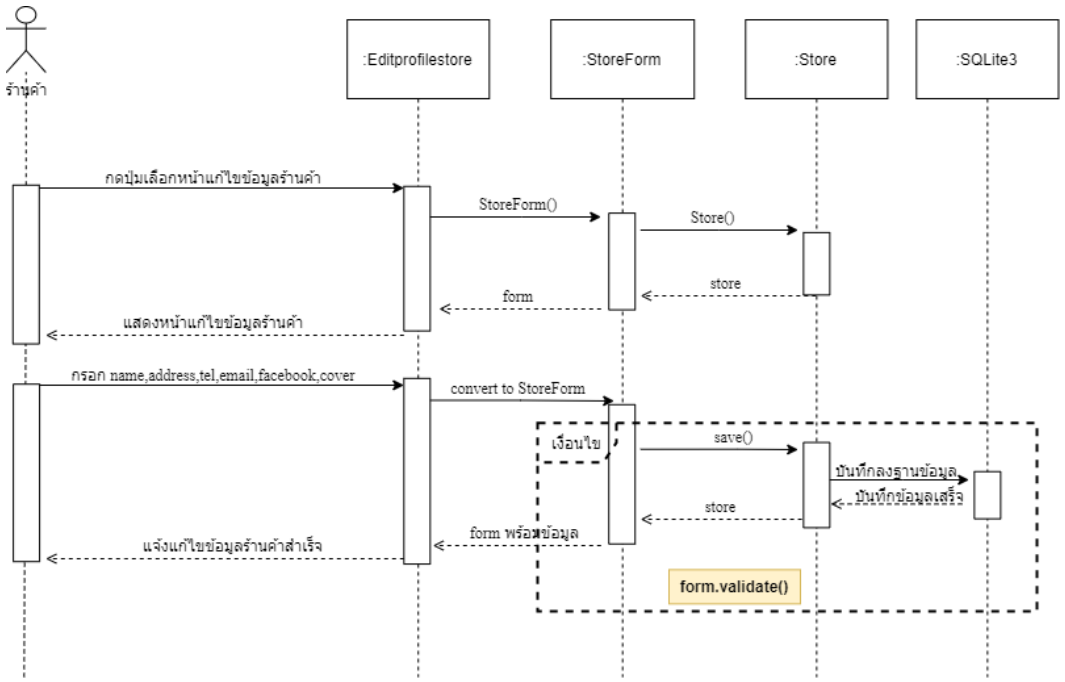
3.6.13 Sequence Diagram ของเพิ่มร้านค้า



รูปที่ 3.32: Sequence Diagram ของเพิ่มร้านค้า

จากภาพที่ 3.32 สามารถอธิบายแผนภาพ Sequence Diagram ของเพิ่มร้านค้า ได้ดังนี้ เมื่อร้านค้ากดปุ่มเพิ่มร้านค้า ระบบจะทำการเรียกใช้ฟังก์ชัน RegisterStore() โดยจะแสดงหน้าเพิ่มร้านค้า ส่งข้อความไปเรียกข้อมูลใน StoreForm() และส่งค่าข้อมูลทั้งหมดของร้านค้ามาในรูปแบบ form เมื่อร้านค้าทำการกรอกชื่อร้าน ที่อยู่ร้าน เบอร์โทรติดต่อ E-mail, Facebook เลือกรูปโปรไฟล์ร้านค้า และทำการกดสมัครสมาชิกหรือเพิ่มร้านค้า ระบบจะทำการส่งข้อมูลทั้งหมดที่ผู้ใช้กรอกครบไปยังฐานข้อมูล SQLite3 ทำการบันทึกลงฐานข้อมูล โดยมีเงื่อนไขคือต้องกรอกข้อมูลให้ครบทุกช่องตามที่กำหนด จึงจะทำการกดเพิ่มร้านค้าได้ และทำการแจ้งเพิ่มร้านค้าสำเร็จ

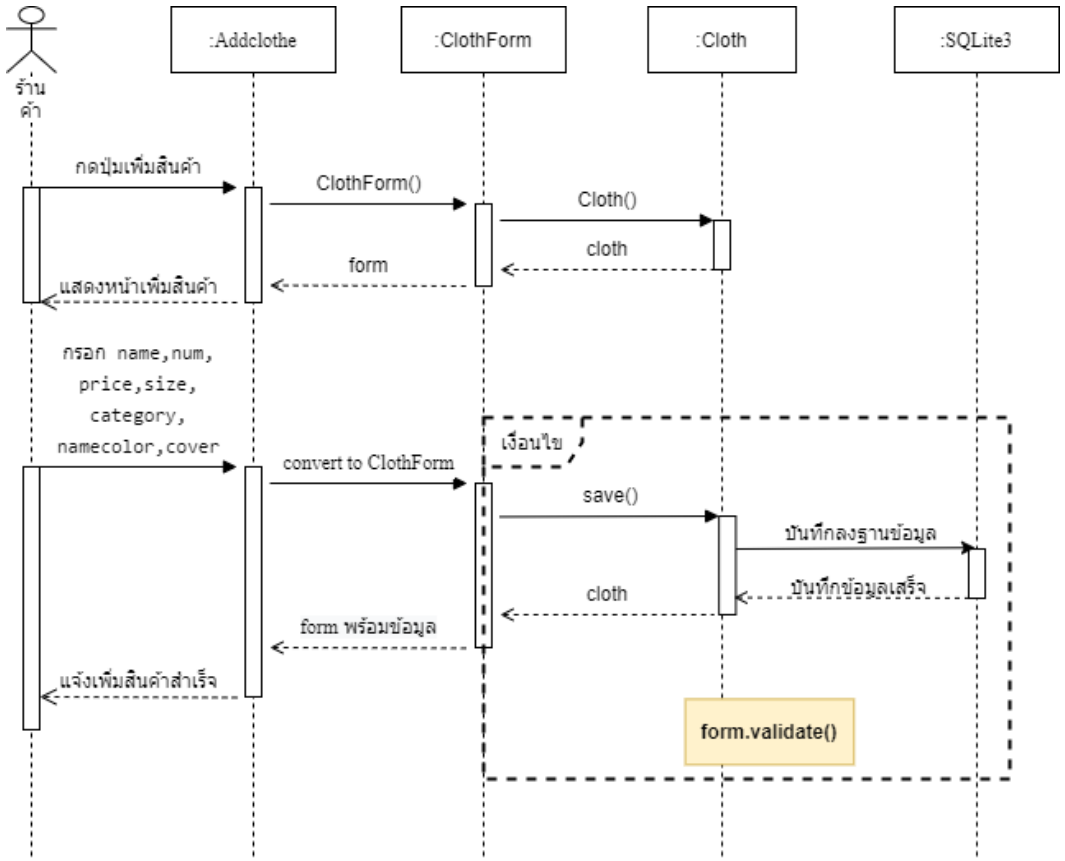
3.6.14 Sequence Diagram ของแก้ไขข้อมูลร้านค้า



รูปที่ 3.33: Sequence Diagram ของแก้ไขข้อมูลร้านค้า

จากภาพที่ 3.33 สามารถอธิบายแผนภาพ Sequence Diagram ของแก้ไขข้อมูลร้านค้า ได้ดังนี้ เมื่อร้านค้ากดปุ่มแก้ไขข้อมูลร้านค้า ระบบจะทำการเรียกใช้ฟังก์ชัน Editpropilestore() โดยจะแสดงหน้าแก้ไขข้อมูลร้านค้า ส่งข้อความไปเรียกข้อมูลใน StoreForm() และส่งค่าข้อมูลทั้งหมดของร้านค้ามาในรูปแบบ form เมื่อร้านค้าทำการกรอกชื่อร้าน ที่อยู่ร้าน เบอร์โทรติดต่อ E-mail, Facebook เลือกรูปโปรไฟล์ร้านค้า และทำการกดแก้ไขข้อมูลร้านค้า ระบบจะทำการส่งข้อมูลทั้งหมดที่ผู้ใช้กรอกครบไปยังฐานข้อมูล SQLite3 ทำการบันทึกลงฐานข้อมูล โดยมีเงื่อนไขคือต้องกรอกข้อมูลให้ครบทุกช่องตามที่กำหนด จึงจะทำการกดเพิ่มร้านค้าได้ และทำการแจ้งแก้ไขข้อมูลร้านค้าสำเร็จ

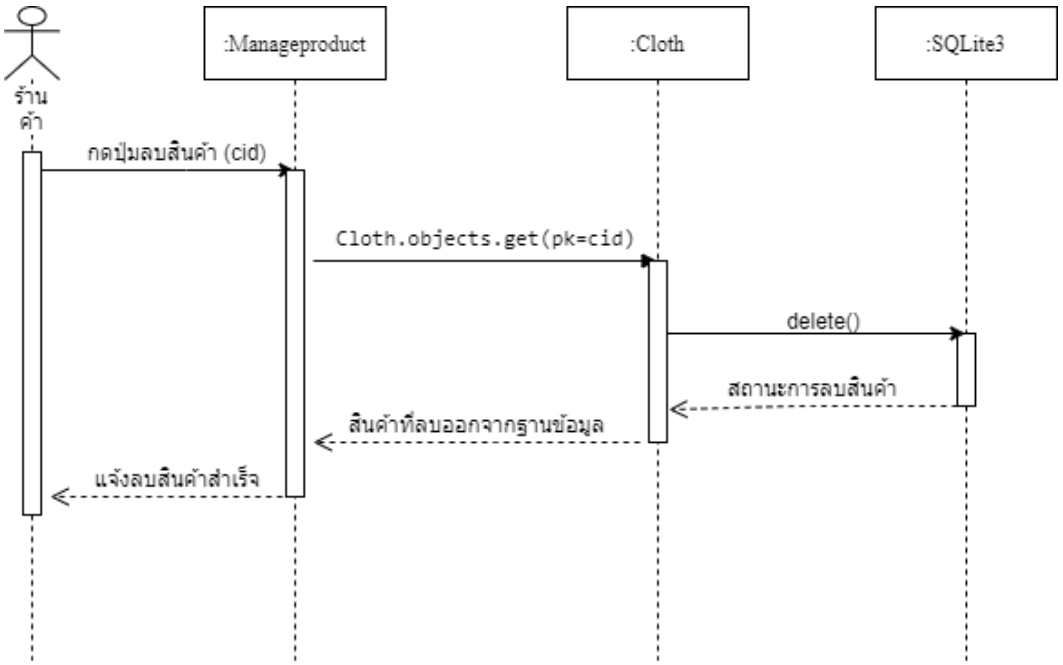
3.6.15 Sequence Diagram ของเพิ่มเสื้อผ้าในร้านค้า



รูปที่ 3.34: Sequence Diagram ของเพิ่มเสื้อผ้าในร้านค้า

จากภาพที่ 3.34 สามารถอธิบายแผนภาพ Sequence Diagram ของเพิ่มเสื้อผ้าในร้านค้า ได้ดังนี้ เมื่อร้านค้ากดเพิ่มเสื้อผ้าในร้านค้า ระบบจะทำการเรียกใช้ฟังก์ชัน Addclothe() โดยจะแสดงหน้าเพิ่มเสื้อผ้า ส่งข้อความไปเรียกข้อมูลใน ClothForm() และส่งค่าข้อมูลทั้งหมดของสินค้ามาในรูปแบบ form เมื่อร้านค้าทำการกรอกชื่อสินต้า จำนวนสินค้า ราคาสินค้า ขนาดสินค้า ประเภทสินค้า สีสินค้า เลือกรูปสินค้า และทำการกดเพิ่มเสื้อผ้า ระบบจะทำการส่งข้อมูลทั้งหมดที่ผู้ใช้กรอกครบไปยังฐานข้อมูล SQLite3 ทำการบันทึกลงฐานข้อมูล โดยมีเงื่อนไขคือต้องกรอกข้อมูลให้ครบทุกช่องตามที่กำหนด จึงจะทำการกดเพิ่มเสื้อผ้าได้ และทำการแจ้งเพิ่มเสื้อผ้าสำเร็จ

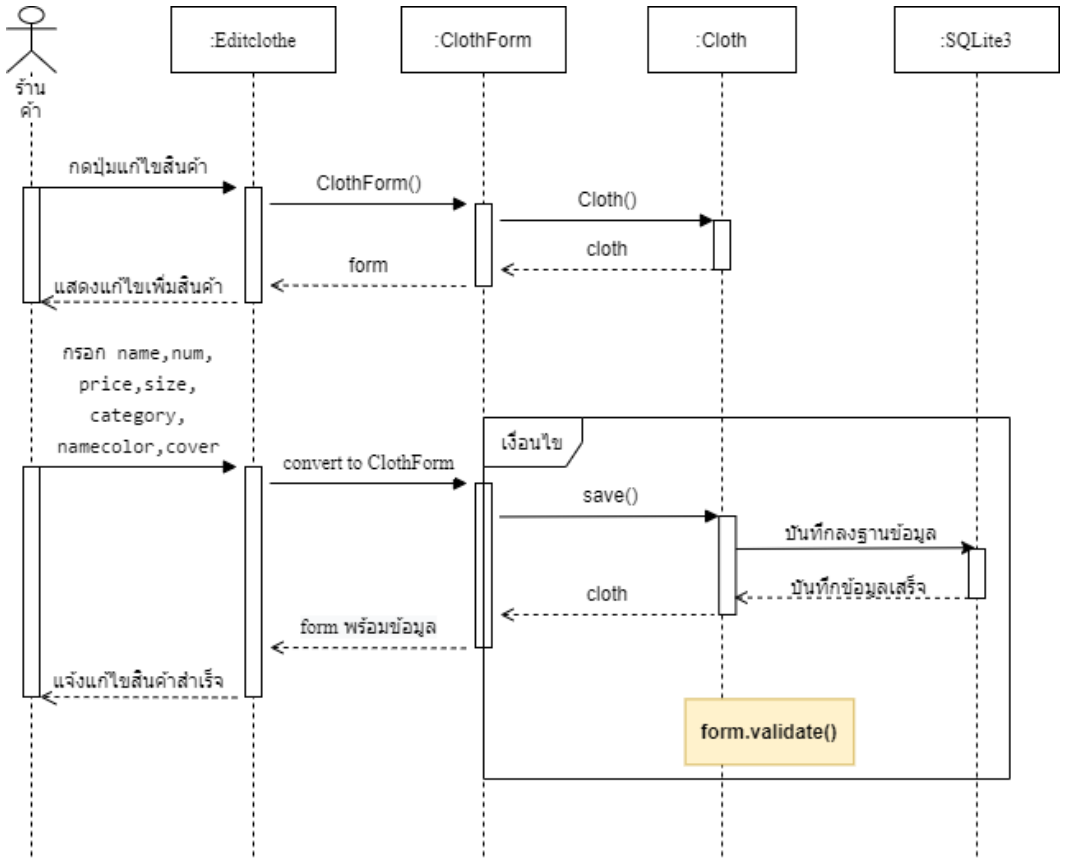
3.6.16 Sequence Diagram ของลบเสื้อผ้าในร้านค้า



รูปที่ 3.35: Sequence Diagram ของลบเสื้อผ้าในร้านค้า

จากภาพที่ 3.35 สามารถอธิบายแผนภาพ Sequence Diagram ของลบเสื้อผ้าในร้านค้า ได้ดังนี้ เมื่อผู้ใช้กดสมัครสมาชิกระบบจะไปเรียกใช้ฟังก์ชัน RegisterUser() โดยจะแสดงหน้าสมัครสมาชิกส่งข้อความไปเรียกข้อมูลใน MemberForm() และส่งค่าข้อมูลทั้งหมดของสมาชิกมาในรูปแบบ form เมื่อผู้ใช้กรอกชื่อ รหัสผาน อีเมล์ เลือกเพศ เลือกวัน/เดือน/ปีเกิด เลือกรูปร่าง เลือกสีผิว เลือกรูปโปรไฟลเสร็จและกดปุ่มสมัครสมาชิกระบบจะไปเรียกฟังก์ชัน save() ซึ่งจะทำการเช็คเงื่อนไข ว่ากรอกข้อมูลครบทุกช่องหรือไม่ ถ้าครบแล้วจะส่งข้อมูล user ไปตรวจสอบในฐานข้อมูล SQLite3 และจะทำการสร้าง user เสร็จจะส่งสถานะการสมัครสมาชิกไปกลับไปให้คลาส Registeruser() และจะแสดงข้อมูลของผู้ใช้ถือว่าสมัครสมาชิกสำเร็จ หรือในกรณีกรอกข้อมูลไม่ครบทุกช่องตามที่กำหนดระบบจะแจ้งให้กรอกช่องที่ยังไม่ได้ทำการกรอก และทำการแจ้งสมัครสมาชิกสำเร็จ

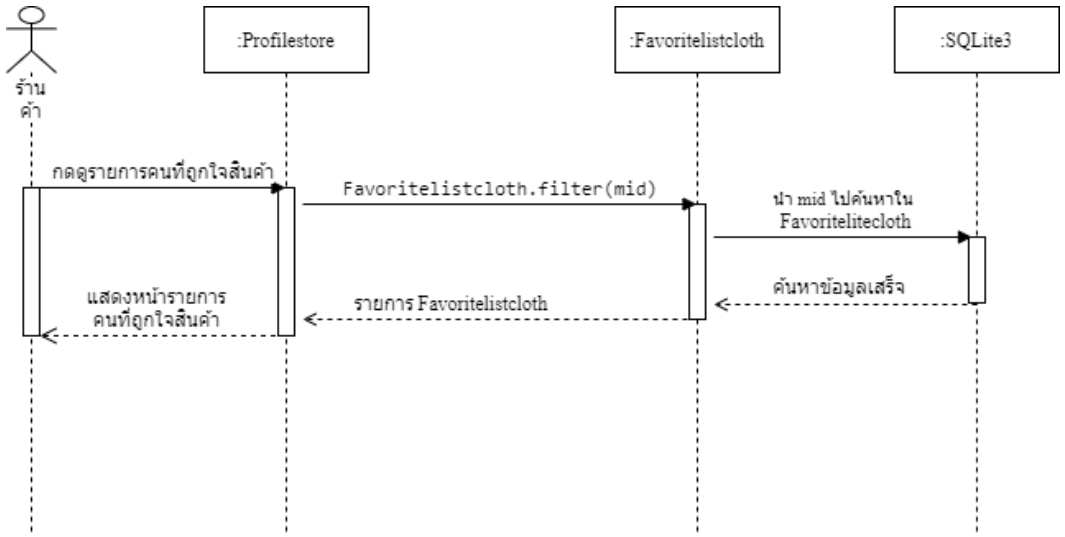
3.6.17 Sequence Diagram ของแก้ไขข้อมูลเสื้อผ้าในร้านค้า



รูปที่ 3.36: Sequence Diagram ของแก้ไขข้อมูลเสื้อผ้าในร้านค้า

จากภาพที่ 3.36 สามารถอธิบายแผนภาพ Sequence Diagram ของแก้ไขข้อมูลเสื้อผ้าในร้านค้า ได้ดังนี้ เมื่อร้านค้ากดแก้ไขข้อมูลเสื้อผ้า ระบบจะทำการเรียกใช้ฟังก์ชัน Editclothe() โดยจะแสดงหน้าแก้ไขข้อมูลเสื้อผ้า ส่งข้อความไปเรียกข้อมูลใน ClothForm() และส่งค่าข้อมูลทั้งหมดของสินค้าในรูปแบบ form เมื่อร้านค้าทำการกรอกชื่อสินต้า จำนวนสินค้า ราคาสินค้า ขนาดสินค้า ประเภทสินค้า สีสินค้า เลือกรูปสินค้า และทำการกดแก้ไขข้อมูลเสื้อผ้า ระบบจะทำการส่งข้อมูลทั้งหมดที่ผู้ใช้กรอกครบไปยังฐานข้อมูล SQLite3 ทำการบันทึกลงฐานข้อมูล โดยมีเงื่อนไขคือต้องกรอกข้อมูลให้ครบทุกช่องตามที่กำหนด จึงจะทำการกดเพิ่มเสื้อผ้าได้ และทำการแจ้งแก้ไขข้อมูลเสื้อผ้าสำเร็จ

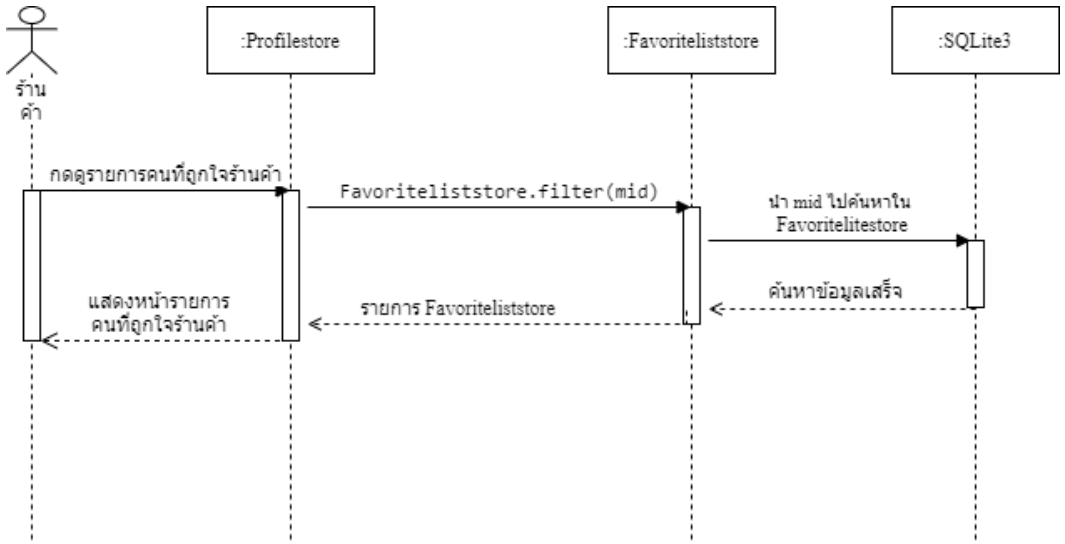
3.6.18 Sequence Diagram ของดูรายการคนที่ถูกใจเสื้อผ้า



รูปที่ 3.37: Sequence Diagram ของดูรายการคนที่ถูกใจเสื้อผ้า

จากภาพที่ 3.37 สามารถอธิบายแผนภาพ Sequence Diagram ของดูรายการคนที่ถูกใจเสื้อผ้า ได้ดังนี้ เมื่อร้านค้าต้องการกดดูรายการคนที่ถูกใจเสื้อผ้า ระบบจะเรียกใช้ฟังก์ชัน Profilestore() และแสดงหน้าข้อมูลร้านค้า เมื่อร้านค้าทำการกดปุ่มดูรายการคนที่ถูกใจสินค้า ระบบจะเรียกใช้ฟังก์ชัน Favoritelistcloth() โดยส่งรหัสผู้ใช้งาน (mid) ไปค้นหาในฐานข้อมูล SQLite3 และแสดงผลลัพธ์รายการรายการคนที่ถูกใจสินค้าตามรายการที่ผู้ใช้ได้ทำการบันทึก

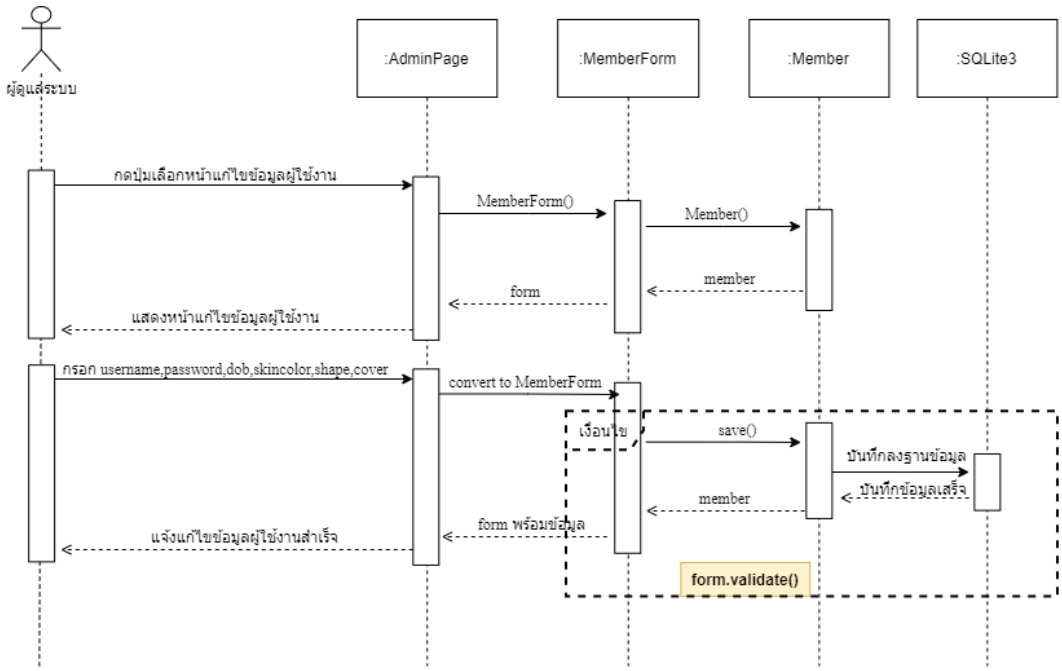
3.6.19 Sequence Diagram ของดูรายการคนที่ถูกใจร้านค้า



รูปที่ 3.38: Sequence Diagram ของดูรายการคนที่ถูกใจร้านค้า

จากภาพที่ 3.38 สามารถอธิบายแผนภาพ Sequence Diagram ของดูรายการคนที่ถูกใจร้านค้า ได้ดังนี้ เมื่อร้านค้าต้องการกดดูรายการคนที่ถูกใจร้านค้า ระบบจะเรียกใช้ฟังก์ชัน Profilestore() และแสดงหน้าข้อมูลร้านค้า เมื่อร้านค้าทำการกดปุ่มดูรายการคนที่ถูกใจร้านค้า ระบบจะเรียกใช้ฟังก์ชัน Favoriteliststore() โดยส่งรหัสผู้ใช้งาน (mid) ไปค้นหาในฐานข้อมูล SQLite3 และแสดงผลลัพธ์รายการรายการคนที่ถูกใจร้านค้าตามรายการที่ผู้ใช้ได้ทำการบันทึก

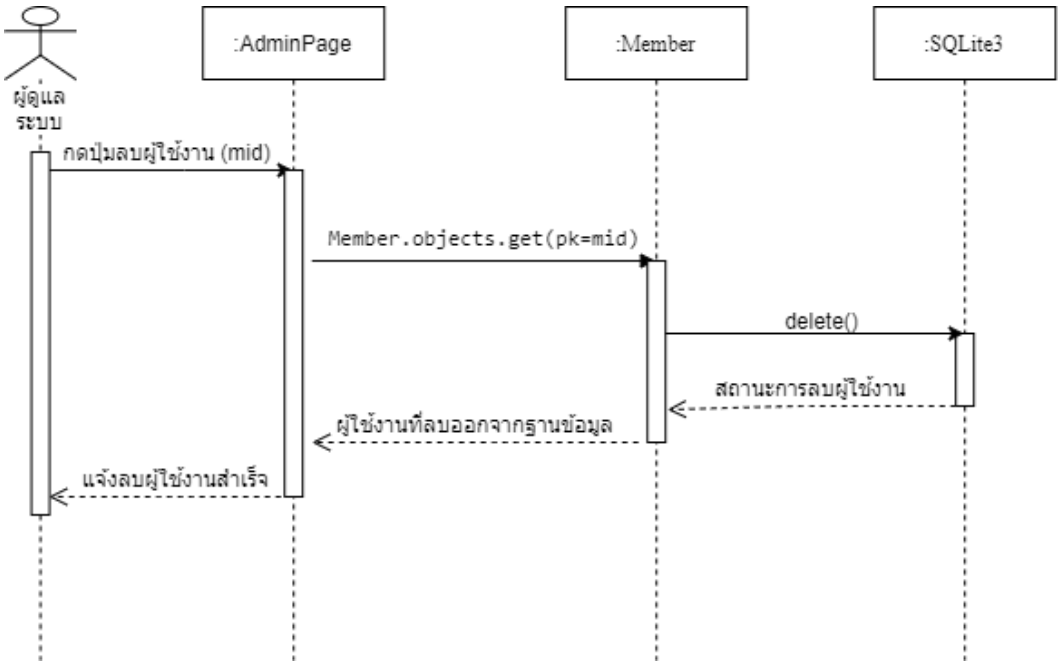
3.6.20 Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน



รูปที่ 3.39: Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน

จากภาพที่ 3.39 สามารถอธิบายแผนภาพ Sequence Diagram ของแก้ไขข้อมูลผู้ใช้งาน ได้ดังนี้ เมื่อผู้ดูแลระบบกดเลือกหน้าแก้ไขข้อมูลผู้ใช้งาน ระบบจะเรียกใช้ฟังก์ชัน AdminPage() ส่งข้อความไปเรียกข้อมูลใน MemberForm() และส่งค่าข้อมูลทั้งหมดของสมาชิกมาในรูปแบบ form จากนั้นเมื่อผู้ใช้ทำการกรอกชื่อผู้ใช้ รหัสผ่าน วัน/เดือน/ปีเกิด สีผิว รูปร่าง เลือกรูปโปร์ไฟล์ และทำการแก้ไขข้อมูลผู้ใช้งาน ระบบจะทำการส่งข้อมูลทั้งหมดที่ผู้ใช้กรอกครบไปยังฐานข้อมูล SQLite3 ทำการบันทึกลงฐานข้อมูล โดยมีเงื่อนไขโดยต้องกรอกข้อมูลชื่อเป็นภาษาอังกฤษและต้องไม่ซ้ำกับฐานข้อมูล กรอกให้ครบทุกช่องก็จะสามารถทำการกดแก้ไขข้อมูลผู้ใช้งานได้ และทำการแจ้งแก้ไขข้อมูลผู้ใช้งานสำเร็จ

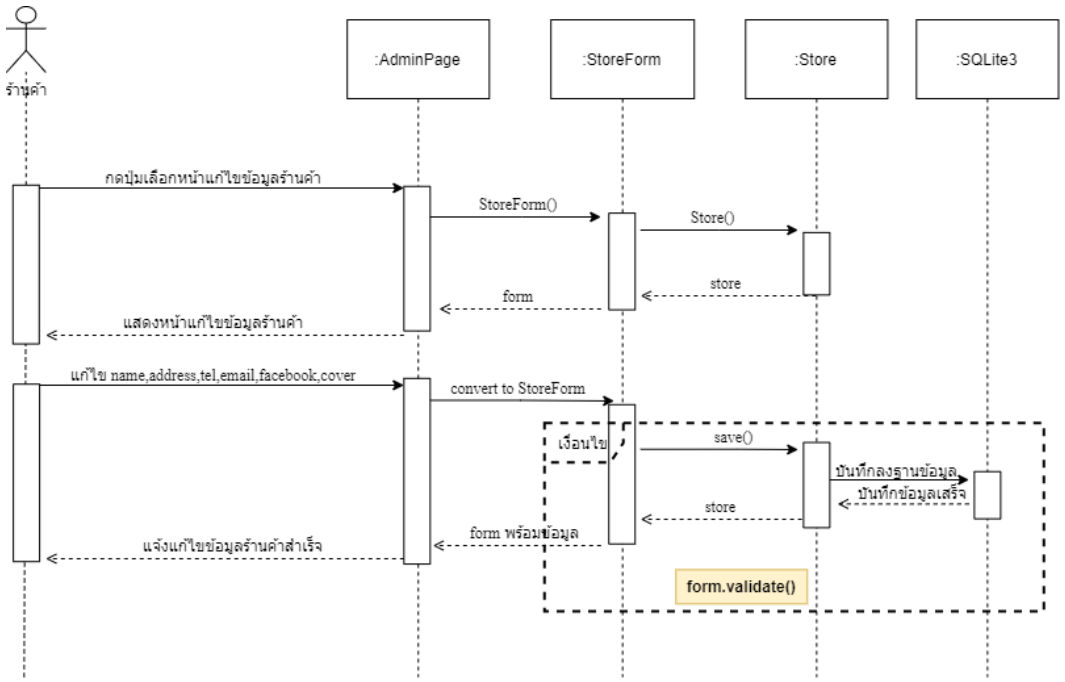
3.6.21 Sequence Diagram ของลบผู้ใช้งาน



รูปที่ 3.40: Sequence Diagram ของลบผู้ใช้งาน

จากภาพที่ 3.40 สามารถอธิบายแผนภาพ Sequence Diagram ของลบผู้ใช้งาน ได้ดังนี้ เมื่อผู้ดูแลระบบต้องการลบผู้ใช้งาน ระบบจะเรียกใช้ฟังก์ชัน AdminPage() และทำการเลือกผู้ใช้ที่ต้องการลบ จากนั้นระบบจะส่งรหัสผู้ใช้ (mid) ไปยังฐานข้อมูล เมื่อร้านค้าทำการกดลบผู้ใช้งาน ระบบจะแสดงสถานะการลบผู้ใช้งานและทำการลบผู้ใช้งานออกจากฐานข้อมูล และแจ้งลบผู้ใช้งานสำเร็จ

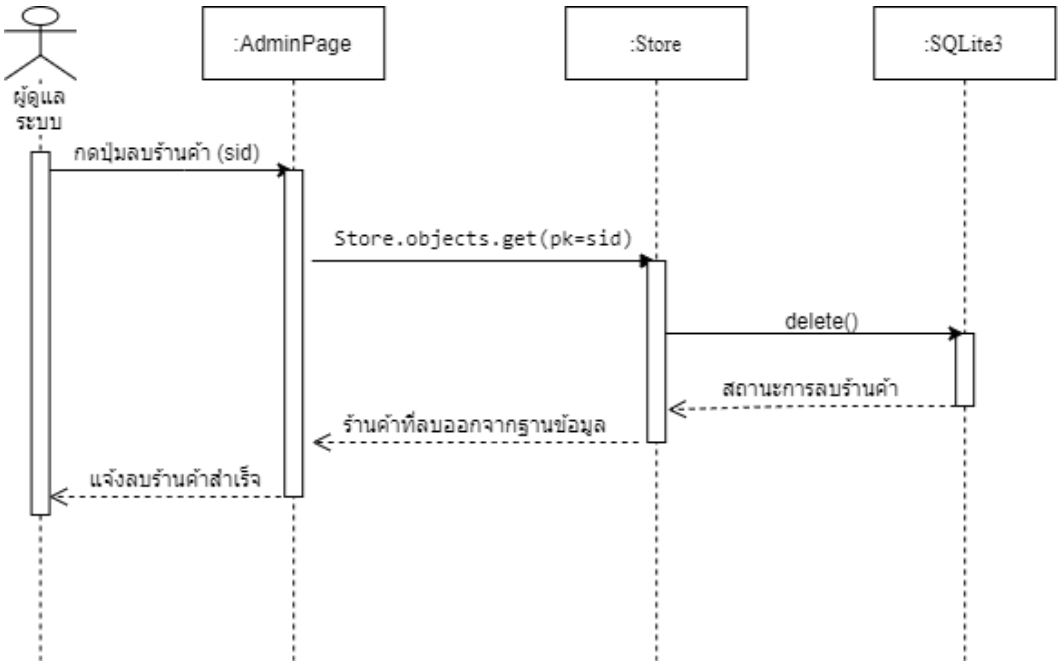
3.6.22 Sequence Diagram ของแก้ไขข้อมูลร้านค้า



รูปที่ 3.41: Sequence Diagram ของแก้ไขข้อมูลร้านค้า

จากภาพที่ 3.41 สามารถอธิบายแผนภาพ Sequence Diagram ของแก้ไขข้อมูลร้านค้า ได้ดังนี้ เมื่อผู้ดูแลระบบกดปุ่มแก้ไขข้อมูลร้านค้า ระบบจะทำการเรียกใช้ฟังก์ชัน AdminPage() โดยจะแสดงหน้าแก้ไขข้อมูลร้านค้า ส่งข้อความไปเรียกข้อมูลใน StoreForm() และส่งค่าข้อมูลทั้งหมดของร้านค้ามาในรูปแบบ form เมื่อร้านค้าทำการกรอกชื่อร้าน ที่อยู่ร้าน เบอร์โทรติดต่อ E-mail, Facebook เลือกรูปโปรไฟล์ร้านค้า และทำการกดแก้ไขข้อมูลร้านค้า ระบบจะทำการส่งข้อมูลทั้งหมดที่ผู้ใช้กรอกครบไปยังฐานข้อมูล SQLite3 ทำการบันทึกลงฐานข้อมูล โดยมีเงื่อนไขคือต้องกรอกข้อมูลให้ครบทุกช่องตามที่กำหนด จึงจะทำการกดเพิ่มร้านค้าได้ และทำการแจ้งแก้ไขข้อมูลร้านค้าสำเร็จ

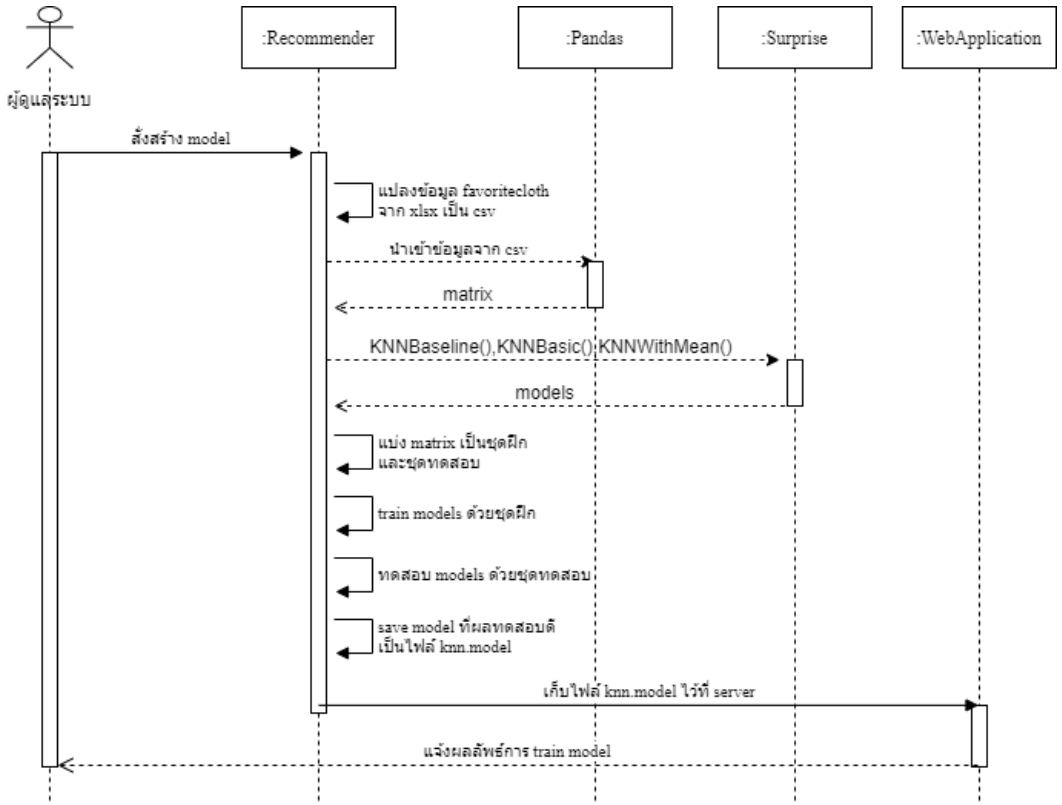
3.6.23 Sequence Diagram ของลบร้านค้า



รูปที่ 3.42: Sequence Diagram ของลบร้านค้า

จากภาพที่ 3.42 สามารถอธิบายแผนภาพ Sequence Diagram ของลบร้านค้า ได้ดังนี้ เมื่อผู้ดูแลระบบต้องการลบร้านค้า ระบบจะเรียกใช้ฟังก์ชัน AdminPage() และทำการเลือกผู้ใช้ที่ต้องการลบ จากนั้นระบบจะส่งรหัสร้านค้า (sid) ไปยังฐานข้อมูล เมื่อร้านค้าทำการกดลบร้านค้า ระบบจะแสดงสถานะการลบร้านค้าและทำการลบร้านค้าออกจากฐานข้อมูล และแจ้งลบร้านค้าสำเร็จ

3.6.24 Sequence Diagram ของฝึกแบบจำลองสำหรับระบบแนะนำ

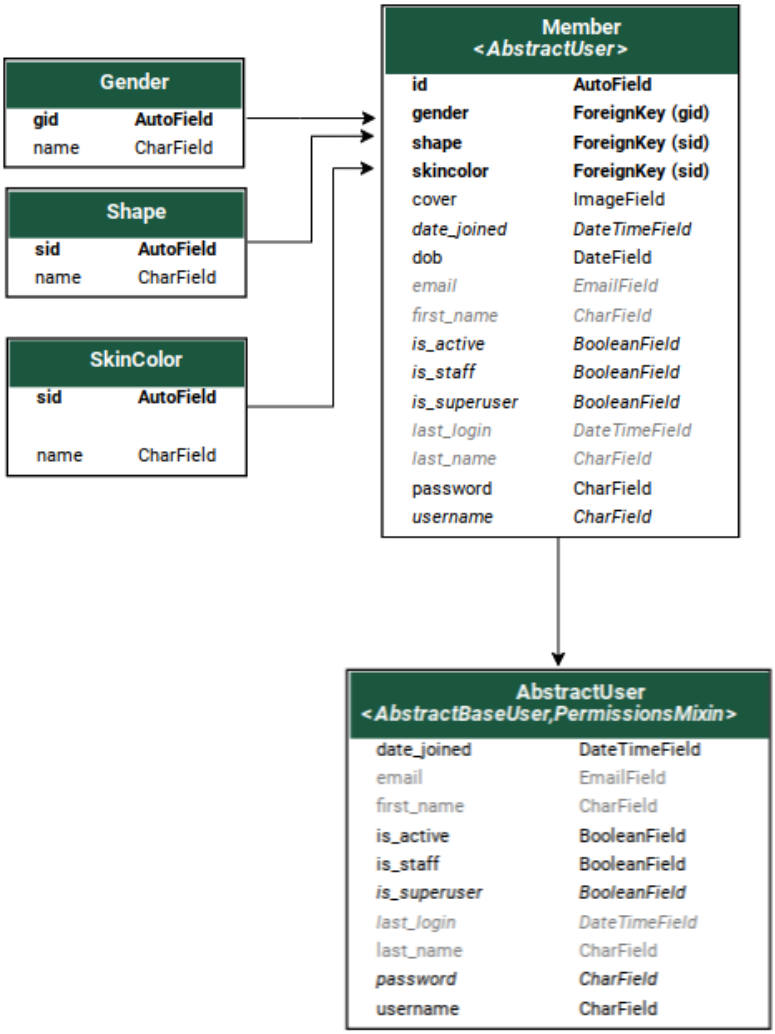


รูปที่ 3.43: Sequence Diagram ของฝึกแบบจำลองสำหรับระบบแนะนำ

จากภาพที่ 3.43 สามารถอธิบายแผนภาพ Sequence Diagram ของฝึกแบบจำลองสำหรับระบบแนะนำ ได้ดังนี้ ผู้ดูแลระบบทำการสั่งสร้าง models Recommender() จากนั้นทำการแปลงข้อมูล favoritecloth จาก xlax เป็น csv และทำการนำส่งข้อมูลจาก csv ไปยัง Pandas() ได้ผลลัพธ์กลับมา matrix จากนั้นทำการคำนวณค่าโดยใช้เทคนิคทั้งหมด 3 เทคนิคดังนี้คือ KNNBaseline(), KNNBasic() และ KNNWithMean() ไปยัง Surprise() และได้ models มาที่ Recommender() จากนั้นทำการแบ่ง matrix เป็นชุดฝึกและชุดทดสอบ แล้วทำการ train models ด้วยชุดฝึก และทดสอบ models ด้วยชุดทดสอบ เมื่อได้ผลลัพธ์แล้วทำการ save modelsที่ผลทดสอบดีที่สุดเป็นไฟล์ knn.model เก็บไฟล์ knn.model ไปไว้ที่ server WebApplication() และแจ้งผลลัพธ์การ train model

3.7 โครงสร้างฐานข้อมูล SQLite3

ฐานข้อมูล SQLite เป็นโปรแกรมฐานข้อมูลที่มีขนาดเล็กมาก (ไม่ถึง 1MB) เก็บฐานข้อมูลเป็นไฟล์โดยไม่จำเป็นต้องมีเซิร์ฟเวอร์ ทำให้ถูกใช้ในหลายๆ โปรแกรมหรือถูกติดตั้งลงไปในอุปกรณ์พกพาหลายชนิดๆ เช่น iPhone, Android เพื่อใช้ในการเก็บข้อมูล



รูปที่ 3.44: ตารางฐานข้อมูลสมาชิกหรือผู้ใช้งาน

ตารางฐานข้อมูลสมาชิกหรือผู้ใช้งาน Member จะประกอบไปด้วย Gender, Shape, Skincolor และ AbstractUser ในส่วนของ AbstractUser นี้ Django จะสร้างขึ้นมาให้อัตโนมัติ

ตารางที่ 3.29: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Member

Key	คำอธิบาย
id	สำหรับเก็บรหัสสมาชิก
gender	สำหรับเก็บข้อมูลเพศ โดยดึงข้อมูลมาจากตาราง Gender
shape	สำหรับเก็บข้อมูลรูปร่าง โดยดึงข้อมูลมาจากตาราง Shape
skincolor	สำหรับเก็บข้อมูลสีผิว โดยดึงข้อมูลมาจากตาราง SkinColor
cover	สำหรับเก็บข้อมูลรูปภาพผู้ใช้งาน
dob	สำหรับเก็บข้อมูลวันที่
password	สำหรับเก็บข้อมูลรหัสผ่าน
username	สำหรับเก็บข้อมูลชื่อผู้ใช้งาน

ตารางที่ 3.30: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Gender

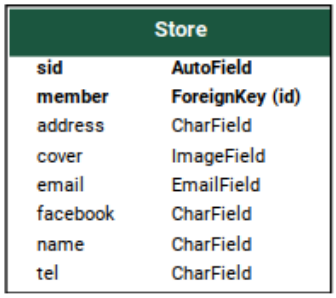
Key	คำอธิบาย
id	สำหรับเก็บรหัสเพศ
name	สำหรับเก็บชื่อเพศ

ตารางที่ 3.31: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Shape

Key	คำอธิบาย
id	สำหรับเก็บรหัสรูปร่าง
name	สำหรับเก็บชื่อรูปร่าง

ตารางที่ 3.32: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ SinColor

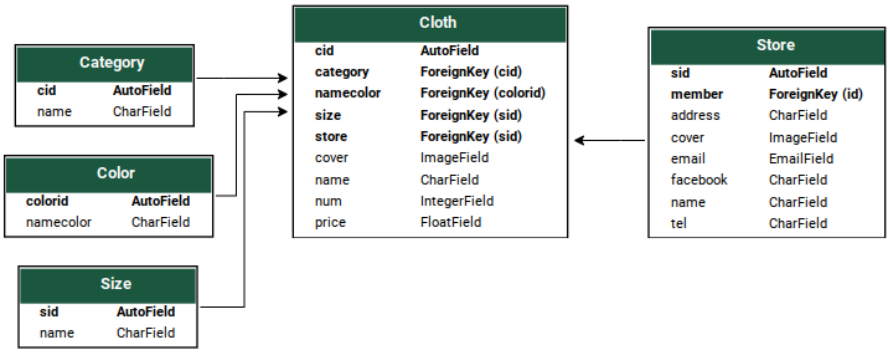
Key	คำอธิบาย
id	สำหรับเก็บรหัสสีผิว
name	สำหรับเก็บชื่อสีผิว



รูปที่ 3.45: ตารางฐานข้อมูลร้านค้า

ตารางที่ 3.33: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Store

Key	คำอธิบาย
sid	สำหรับเก็บรหัสร้านค้า
member	สำหรับเก็บชื่อผู้ใช้ ซึ่งจะดึงข้อมูลมาจากตาราง Member
address	สำหรับเก็บข้อมูลที่อยู่ร้านค้า
cover	สำหรับเก็บข้อมูลรูปภาพร้านค้า
email	สำหรับเก็บข้อมูลอีเมล์ร้านค้า
facebook	สำหรับเก็บข้อมูลเฟสบุ๊กร้านค้า
name	สำหรับเก็บชื่อร้านค้า
tel	สำหรับเก็บข้อมูลชื่อผู้ใช้งาน



รูปที่ 3.46: ตารางฐานข้อมูลเสื้อผ้า

ตารางฐานข้อมูลเสื้อผ้า Cloth จะประกอบไปด้วย Category, Color และ Size

ตารางที่ 3.34: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Cloth

Key	คำอธิบาย
cid	สำหรับเก็บรหัสสินค้า
category	สำหรับเก็บประเภทสินค้า โดยดึงข้อมูลมาจากตาราง Category
namecolor	สำหรับเก็บสีสินค้า โดยดึงข้อมูลมาจากตาราง Color
size	สำหรับเก็บขนาดสินค้า โดยดึงข้อมูลมาจากตาราง Size
store	สำหรับเก็บข้อมูลร้านค้า โดยดึงข้อมูลมาจากตาราง Store
cover	สำหรับเก็บรูปภาพสินค้า
name	สำหรับเก็บชื่อสินค้า
num	สำหรับเก็บจำนวนสินค้า
price	สำหรับเก็บราคาสินค้า

ตารางที่ 3.35: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Category

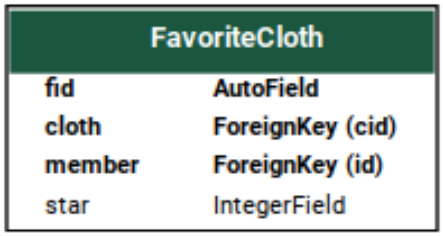
Key	คำอธิบาย
cid	สำหรับเก็บรหัสประเภทสินค้า
name	สำหรับเก็บชื่อประเภทสินค้า

ตารางที่ 3.36: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Color

Key	คำอธิบาย
colorid	สำหรับเก็บรหัสสีสินค้า
namecolor	สำหรับเก็บชื่อสีสินค้า

ตารางที่ 3.37: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Size

Key	คำอธิบาย
sid	สำหรับเก็บรหัสขนาดสินค้า
name	สำหรับเก็บชื่อขนาดสินค้า



รูปที่ 3.47: ตารางฐานข้อมูลสินค้าโปรด

ตารางที่ 3.38: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ FavoriteCloth

Key	คำอธิบาย
fid	สำหรับเก็บรหัสสินค้าโปรด
cloth	สำหรับเก็บชื่อข้อมูลสินค้า โดยดึงข้อมูลมาจากตาราง Cloth
member	สำหรับเก็บชื่อข้อมูลชื่อผู้ใช้ โดยดึงข้อมูลมาจากตาราง Member
star	สำหรับเก็บชื่อข้อมูลอันดับความชอบสินค้า

รูปที่ 3.48: ตารางฐานข้อมูลร้านโปรด

ตารางที่ 3.39: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ FavoriteStore

Key	คำอธิบาย
fid	สำหรับเก็บรหัสร้านโปรด
member	สำหรับเก็บชื่อข้อมูลชื่อผู้ใช้ โดยดึงข้อมูลมาจากตาราง Member
store	สำหรับเก็บชื่อข้อมูลชื่อร้าค้า โดยดึงข้อมูลมาจากตาราง Store

รูปที่ 3.49: ตารางฐานข้อมูลแนะนำเสื้อผ้า

ตารางที่ 3.40: อธิบายโครงสร้างฐานข้อมูลของ datamodel ของ Recommend

Key	คำอธิบาย
rid	สำหรับเก็บรหัสรายการแนะนำเสื้อผ้า
favorite	สำหรับเก็บชื่อข้อมูลเสื้อผ้าที่ผู้ใช้ได้กดเพิ่มเป็นสินค้าโปรด จากตาราง FavoriteCloth
member	สำหรับเก็บชื่อข้อมูลชื่อผู้ใช้ โดยดึงข้อมูลมาจากตาราง Member

บทที่ 4

การพัฒนาระบบ

ในบทนี้จะกล่าวถึงการสร้างระบบงานของระบบเว็บแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรี โดยนำผลที่ได้จากการวิเคราะห์และออกแบบระบบมาสร้างเป็นระบบงานซึ่งจะอธิบายถึงตัวอย่างการเขียนโปรแกรมการทำงานของระบบในส่วนต่างๆ ดังต่อไปนี้

4.1 การพัฒนาในส่วนการสมัครสมาชิกหรือเพิ่มข้อมูล

เมื่อผู้ใช้ต้องการทำการสมัครสมาชิก การเพิ่มร้านค้า และการเพิ่มสินค้า โดยผู้พัฒนาจะยกตัวอย่างการทำงานแค่การสมัครสมาชิก เนื่องจากการทำงานของทั้งการสมัครสมาชิก การเพิ่มร้านค้า และการเพิ่มสินค้ามีการทำงานเหมือนกัน มีรายละเอียดการทำงานดังนี้

รูปที่ 4.1: การทำงานของระบบเมื่อผู้ใช้สมัครสมาชิกหรือเพิ่มข้อมูล

จากภาพที่ 4.1 โครงสร้างของไฟล์ views.py อธิบายการทำงานได้ดังนี้ สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มสมัครสมาชิกระบบจะเข้ามาทำงานในฟังก์ชัน registeruser
- บรรทัดที่ 2 เป็นการสร้าง form เพื่อเก็บข้อมูลของผู้ใช้งานโดยจะสร้างแยกออกไปยังไฟล์ forms.py
- บรรทัดที่ 3-11 เป็นการตรวจสอบว่าข้อมูลที่กรอกเข้ามาถูกต้องตรงตามที่กำหนดในไฟล์ forms.py ถ้าครบและถูกต้องให้ทำการบันทึกข้อมูลได้ แต่ถ้าไม่ถูกต้องให้แสดง error ในส่วนที่ผิดนั้นออกมา

- บรรทัดที่ 12 เป็นการตั้งค่าการแสดงวัน เดือน ปี
- บรรทัดที่ 13 เป็นการส่งค่าไปแสดงยังหน้า registeruser.html
- บรรทัดที่ 14-19 เป็นการกำหนดค่าตัวแปรที่จะให้แสดงในหน้า registeruser.html

```
1
2
3
4
```

รูปที่ 4.2: การทำงานของระบบเมื่อผู้ใช้สมัครสมาชิกหรือเพิ่มข้อมูล (ต่อ)

จากภาพที่ 4.2 โครงสร้างของไฟล์ forms.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1-4 เป็นการสร้าง form ข้อมูลแยกออกมาเพื่อทำการส่งค่าไปยังไฟล์ views.py เพื่อให้ได้ข้อมูลที่ครบและถูกต้องตามที่กำหนด

4.2 การพัฒนาในส่วนการเข้าสู่ระบบ

เมื่อผู้ใช้กรอกข้อมูลทั้งหมดเสร็จทำการกดปุ่มเข้าสู่ระบบ ระบบจะมีรายละเอียดการทำงานดังนี้

```
1
2
3
4
5
6
7
8 เขายังไม่ได้กรอก(’ login/password ครั้งแรกที่เข้าหน้านี้()’)
9 return render(req, ’lookinggreat/login.html’)
```

รูปที่ 4.3: การทำงานของระบบเมื่อผู้ใช้เข้าสู่ระบบ

จากภาพที่ 4.3 โครงสร้างของไฟล์ views.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดเข้าสู่ระบบจะเข้ามาทำงานในฟังก์ชัน login()
- บรรทัดที่ 2-3 เป็นการยิงข้อมูล username และ password ที่ผู้ใช้กรอกเข้ามาเพื่อนำไปตรวจสอบกับข้อมูลในฐานข้อมูล
- บรรทัดที่ 4-6 หากผู้ใช้กรอก username หรือ password ไม่ตรงกับข้อมูลในฐานข้อมูลและกดปุ่มเข้าสู่ระบบ ระบบจะแสดงหน้าเข้าสู่ระบบเหมือนเดิม
- บรรทัดที่ 7-9 หากผู้ใช้ยังไม่ได้กรอก username และ password แล้วกดปุ่มเข้าสู่ระบบ ระบบจะแสดงหน้าเข้าสู่ระบบเหมือนเดิม

4.3 การพัฒนาในส่วนการแสดงข้อมูลทั้งหมด

เมื่อผู้ใช้กดเลือกรายการเสื้อผ้าทั้งหมด และรายการร้านค้าทั้งหมด โดยผู้พัฒนาจะยกตัวอย่างการทำงานแค่การแสดงรายการเสื้อผ้าทั้งหมด เนื่องจากการทำงานของทั้งการรายการเสื้อผ้าทั้งหมด และรายการร้านค้าทั้งหมดมีการทำงานเหมือนกัน ระบบจะมีรายละเอียดการทำงานดังนี้

รูปที่ 4.4: การทำงานของระบบเมื่อผู้ใช้กดดูรายการเสื้อผ้าทั้งหมด

จากภาพที่ 4.4 โครงสร้างของไฟล์ views.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มเสื้อผ้าทั้งหมดระบบจะเข้ามาทำงานในฟังก์ชัน clothe()
- บรรทัดที่ 2 เป็นการเรียกดูข้อมูลทั้งหมดที่มีในตารางของเสื้อผ้า
- บรรทัดที่ 3-9 เป็นการค้นหาเสื้อผ้าจากชื่อของเสื้อผ้า
- บรรทัดที่ 10 เป็นการส่งค่าไปแสดงยังหน้า clothe.html
- บรรทัดที่ 11-12 เป็นการกำหนดค่าตัวแปรที่จะให้แสดงในหน้า clothe.html

รูปที่ 4.5: การทำงานของระบบเมื่อผู้ใช้กดดูรายการเสื้อผ้าทั้งหมด (ต่อ)

จากภาพที่ 4.5 โครงสร้างของไฟล์ views.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มเลือกดูเฉพาะเสื้อระบบจะเข้ามาทำงานในฟังก์ชัน clothe cat() โดยระบบได้แยกประเภทของเสื้อผ้าทั้งหมดออกเป็น 6 ประเภท ได้แก่ เสื้อ กางเกง กระโปรง เดรส สูท และอื่นๆ
- บรรทัดที่ 2 เป็นการดึงข้อมูลจากตารางของประเภทเสื้อผ้ามาแสดง โดยจะแสดงตามประเภทที่ผู้ใช้ได้เลือก
- บรรทัดที่ 3 เป็นการเรียกดูข้อมูลทั้งหมดที่มีในตารางของเสื้อผ้า
- บรรทัดที่ 4-7 เป็นการแสดงผลหลังจากได้ทำงานกดเลือกประเภทเสื้อผ้านั้น
- บรรทัดที่ 8 เป็นการส่งค่าไปแสดงยังหน้า clothe.html
- บรรทัดที่ 9-10 เป็นการกำหนดค่าตัวแปรที่จะให้แสดงในหน้า clothe.html

4.4 การพัฒนาในส่วนการแสดงรายละเอียดของข้อมูล

เมื่อผู้ใช้กดปุ่มดูรายละเอียดสินค้า และรายละเอียดร้านค้า โดยผู้พัฒนาจะยกตัวอย่างการทำงานแค่การดูรายละเอียดสินค้า เนื่องจากการทำงานของทั้งการดูรายละเอียดสินค้า และรายละเอียดร้านค้ามีการทำงานเหมือนกัน ระบบจะมีรายละเอียดการทำงานดังนี้

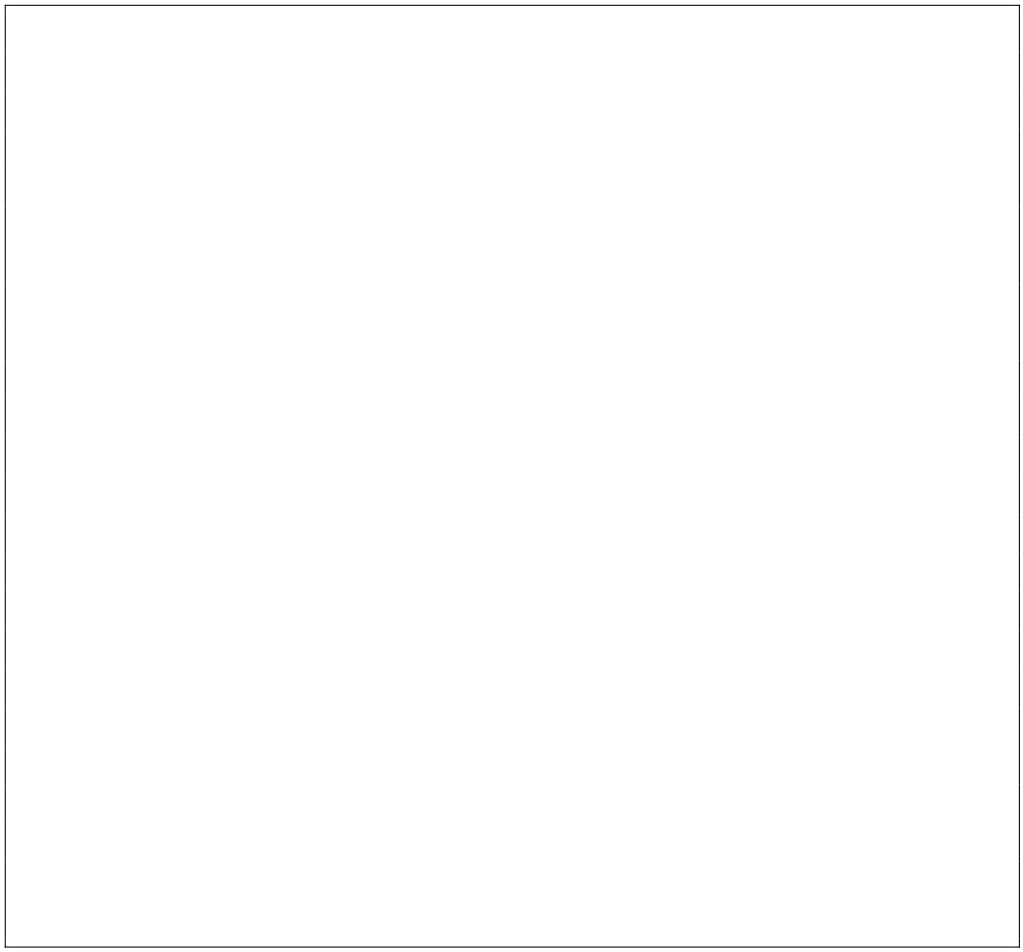
รูปที่ 4.6: การทำงานของระบบเมื่อผู้ใช้กดดูรายละเอียดสินค้า

จากภาพที่ 4.6 โครงสร้างของไฟล์ views.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มดูรายละเอียดสินค้าระบบจะเข้ามาทำงานในฟังก์ชัน clothedetail()
- บรรทัดที่ 2 เป็นการแสดงรายละเอียดของสินค้าตามที่ผู้ใช้ได้ทำการเลือกดู
- บรรทัดที่ 3-4 เป็นการตรวจสอบว่าผู้ใช้ได้เป็นสมาชิกและเข้าสู่ระบบแล้วหรือไม่ ถ้ายังไม่เข้าสู่ระบบ ระบบจะไม่แสดงปุ่มเพิ่มเป็นสินค้าโปรด
- บรรทัดที่ 5-10 เป็นการตรวจสอบว่าผู้ใช้ได้เป็นสมาชิกและเข้าสู่ระบบแล้วหรือไม่ ถ้าเข้าสู่ระบบเรียบร้อยแล้ว ระบบจะแสดงปุ่มเพิ่มเป็นสินค้าโปรดและสามารถกดเพิ่มได้
- บรรทัดที่ 11 เป็นการส่งค่าไปแสดงยังหน้า clothedetail.html
- บรรทัดที่ 12-15 เป็นการกำหนดค่าตัวแปรที่จะให้แสดงในหน้า clothedetail.html

4.5 การพัฒนาในส่วนการแก้ไขข้อมูล

เมื่อผู้ใช้กดปุ่มแก้ไขสินค้า แก้ไขข้อมูลผู้ใช้งาน และแก้ไขร้านค้า โดยผู้พัฒนาจะยกตัวอย่างการทำงานแค่การแก้ไขสินค้า เนื่องจากการทำงานของทั้งการแก้ไขสินค้า แก้ไขข้อมูลผู้ใช้งาน และแก้ไขร้านค้ามีการทำงานเหมือนกัน ระบบจะมีรายละเอียดการทำงานดังนี้



รูปที่ 4.7: การทำงานของระบบเมื่อผู้ใช้กดแก้ไขสินค้า

จากภาพที่ 4.7 โครงสร้างของไฟล์ views.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มแก้ไขสินค้าระบบจะเข้ามาทำงานในฟังก์ชัน editcloth()
- บรรทัดที่ 2 เป็นการแสดงรายละเอียดของสินค้าตามที่ผู้ใช้ได้ทำการเลือกแก้ไข
- บรรทัดที่ 3-6 เป็นการเรียกดูข้อมูลจากตารางต่างๆ เพื่อนำมาแสดงในรายละอียดของสินค้าแต่ละชิ้น
- บรรทัดที่ 7-15 เป็นการตรวจสอบว่าข้อมูลที่ได้กรอกไปนั้นถูกต้องและตรงตามที่กำหนดในไฟล์ forms.py หรือไม่ ถ้าครบและถูกต้องให้ทำการบันทึกข้อมูลได้ แต่ถ้าไม่ถูกต้องให้แสดง error ในส่วนที่ผิดนั้นออกมา
- บรรทัดที่ 16 เป็นการส่งค่าไปแสดงยังหน้า editcloth.html

- บรรทัดที่ 17-23 เป็นการกำหนดค่าตัวแปรที่จะให้แสดงในหน้า editcloth.html

```
1
2
3
4
```

รูปที่ 4.8: การทำงานของระบบเมื่อผู้ใช้กดแก้ไขสินค้า (ต่อ)

จากภาพที่ 4.8 โครงสร้างของไฟล์ forms.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1-4 เป็นการสร้าง form ข้อมูลแยกออกมาเพื่อทำการส่งค่าไปยังไฟล์ views.py เพื่อให้ได้ข้อมูลที่ครบและถูกต้องตามที่กำหนด

4.6 การพัฒนาในส่วนการลบข้อมูล

เมื่อผู้ใช้กดปุ่มลบสินค้า ลบรายการสินค้าโปรด และลบรายการร้านโปรด โดยผู้พัฒนาจะยกตัวอย่างการทำงานแค่การลบสินค้า เนื่องจากการทำงานของทั้งการลบสินค้า การลบรายการสินค้าโปรด และการลบร้านค้าโปรดมีการทำงานที่เหมือนกัน ระบบจะมีรายละเอียดการทำงานดังนี้

รูปที่ 4.9: การทำงานของระบบเมื่อผู้ใช้กดลบสินค้า

จากภาพที่ 4.9 โครงสร้างของไฟล์ views.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มลบสินค้าระบบจะเข้ามาทำงานในฟังก์ชัน delete clothe()
- บรรทัดที่ 2-3 เป็นการลบสินค้าตามที่ผู้ใช้ได้ทำการเลือก
- บรรทัดที่ 4 เป็นการส่งค่าไปยังหน้าหลักที่ใช้งานอยู่คือหน้า manageproduct.html

4.7 การพัฒนาในส่วนการเพิ่มเป็นสินค้าโปรด

เมื่อผู้ใช้กดปุ่มเพิ่มเป็นสินค้าโปรด และเพิ่มเป็นร้านโปรด ระบบจะมีการทำงาน แสดงดังรูปที่ 4.10 โดยผู้พัฒนาจะยกตัวอย่างการทำงานแค่การเพิ่มเป็นสินค้าโปรด เนื่องจากการทำงานของทั้งเพิ่มเป็นสินค้าโปรดและเพิ่มเป็นร้านโปรดมีการทำงานที่เหมือนกัน ระบบจะมีรายละเอียดการทำงานดังนี้

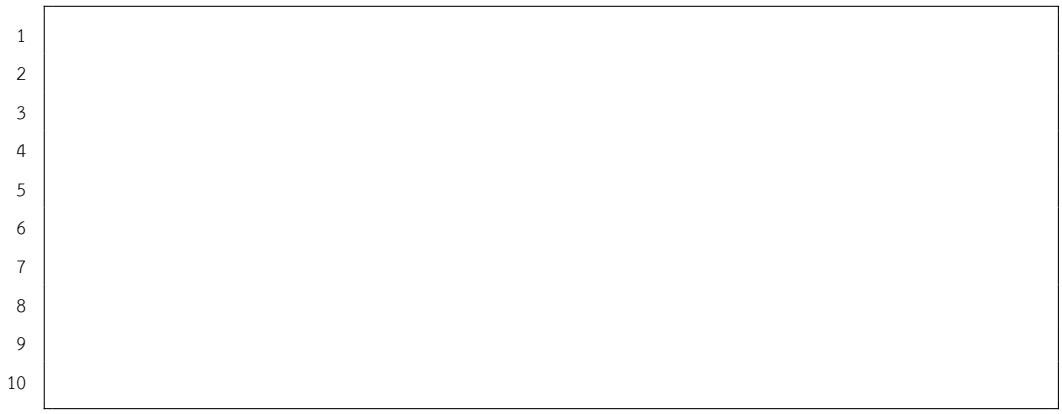
รูปที่ 4.10: การทำงานของระบบเมื่อผู้ใช้กดเพิ่มเป็นสินค้าโปรด

จากภาพที่ 4.10 โครงสร้างของไฟล์ views.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มเพิ่มเป็นสินค้าโปรดระบบจะเข้ามาทำงานในฟังก์ชัน favoriteclothadd()
- บรรทัดที่ 2 เป็นการรับค่าของสินค้าที่ผู้ใช้เลือกเพื่อทำการเพิ่มเป็นสินค้าโปรด
- บรรทัดที่ 3 เป็นการกำหนดค่าตัวแปลเพื่อนำไปใช้ในการการเก็บค่าของสินค้าโปรด
- บรรทัดที่ 4 เป็นการตรวจสอบว่าผู้ใช้ได้ทำการเข้าสู่ระบบแล้วหรือไม่
- บรรทัดที่ 5-6 เป็นการเก็บค่าของข้อมูลสินค้าโปรดและทำการบันทึกลงในรายการสินค้าโปรด
- บรรทัดที่ 7 เป็นการส่งค่าไปยังหน้าหลักที่ใช้งานอยู่คือหน้า clothedetail.html

4.8 การพัฒนาในส่วนการดูรายการคนที่ถูกใจสินค้า

เมื่อผู้ใช้กดปุ่มดูรายการคนที่ถูกใจสินค้า และถูกใจร้านค้า โดยผู้พัฒนาจะยกตัวอย่างการทำงานแค่การดูรายการคนที่ถูกใจสินค้า เนื่องจากการทำงานของทั้งรายการคนที่ถูกใจสินค้าและรายการคนที่ถูกใจร้านค้ามีการทำงานที่เหมือนกัน ระบบจะมีรายละเอียดการทำงานดังนี้



รูปที่ 4.11: การทำงานของระบบเมื่อผู้ใช้กดดูรายการคนที่ถูกใจสินค้า

จากภาพที่ 4.11 โครงสร้างของไฟล์ views.py สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มดูรายการคนที่ถูกใจสินค้าระบบจะเข้ามาทำงานในฟังก์ชัน favoritelitecloth()
- บรรทัดที่ 2-4 เมื่อผู้ใช้เข้าสู่ระบบเรียนร้อยแล้ว ระบบจะทำการแสดงชื่อผู้ใช้งาน ร้านค้า และรายการคนที่ถูกใจร้านค้า
- บรรทัดที่ 5 เป็นการส่งค่าไปแสดงยังหน้า favoritelitecloth.html
- บรรทัดที่ 6-9 เป็นการกำหนดค่าตัวแปรที่จะให้แสดงในหน้า favoritelitecloth.html

4.9 การพัฒนาในส่วนแนะนำสินค้า

4.9.1 การฝึกต้นแบบของระบบผู้แนะนำ

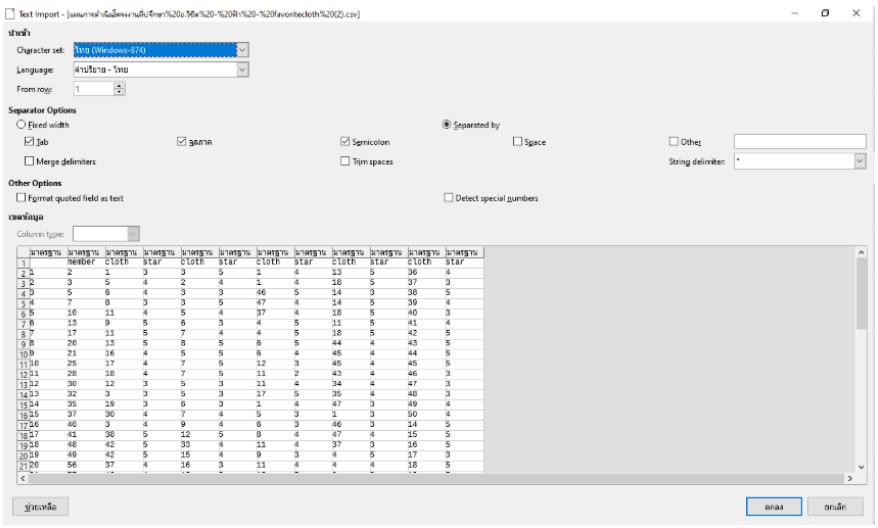
ลำดับในการฝึกต้นแบบของระบบผู้แนะนำมีดังนี้

No.	Member	Cloth	Star	Cloth	Star	Cloth	Star	Cloth	Star	Cloth	Star
1	2	1	3	3	5	1	4	13	5	36	4
2	3	5	4	2	4	1	4	18	5	37	3
3	5	6	4	3	3	46	5	14	3	38	5
4	7	8	3	3	5	47	4	14	5	39	4
5	10	11	4	5	4	37	4	18	5	40	3
6	13	9	5	6	3	4	5	11	5	41	4
7	17	11	5	7	4	4	5	18	5	42	5
8	20	13	5	8	5	6	5	44	4	43	5
9	21	16	4	5	5	6	4	45	4	44	5
10	25	17	4	7	5	12	3	45	4	45	5

รูปที่ 4.12: ภาพประกอบตารางการจำลองข้อมูลการให้ดาว ตัวอย่าง 10 แถว จาก 60 แถว

จากภาพที่ 4.12 สามารถอธิบายได้ดังนี้

- เป็นภาพประกอบตารางการจำลองข้อมูลการให้ดาว ตัวอย่าง 10 แถว จาก 60 แถว ข้อมูลจะเป็น excel ประเภท xlsx โดยข้อมูลในส่วนนี้จะนำไปใช้ในการสร้าง model เพื่อนำไปใช้ในการวิเคราะห์ผลในการแนะนำรายการเสื้อผ้า



รูปที่ 4.13: ภาพประกอบไฟล์จากการ download แปลงข้อมูล favoritecloth จาก xlsx เป็น csv

จากภาพที่ 4.13 สามารถอธิบายได้ดังนี้

- เป็นตารางที่ได้จากการ download แปลงข้อมูล favoritecloth จาก xlsx เป็น csv เพื่อนำไปใช้ในการจำลองข้อมูลเพื่อสร้าง model

```python
import datetime

def favoritecloth():
    fname = 'แผนการดำเนินโครงงานที่ปรึกษา อ.วิษิต - ฟ้า - favoritecloth.csv'
    with open(fname, 'r') as f:
        with open('../notebooks/FavoriteCloth.csv', 'w') as of:
            of.writelines(f'member,cloth,star\n')
            for line in f.readlines()[1:]:
                #2,1,3,3,5,1,4,13,5,36,4
                a = line.split(',')
                of.writelines(f'{a[1]},{a[2]},{a[3]}\n')
                of.writelines(f'{a[1]},{a[4]},{a[5]}\n')
                of.writelines(f'{a[1]},{a[6]},{a[7]}\n')
                of.writelines(f'{a[1]},{a[8]},{a[9]}\n')
                of.writelines(f'{a[1]},{a[10]},{a[11]}\n')

if __name__ == '__main__':
    favoritecloth()
```

รูปที่ 4.14: คำสั่งแปลง 11-column csv เป็น 3-column csv ที่สามารถอ่านด้วย pandas ได้

จากภาพที่ 4.14 สามารถอธิบายได้ดังนี้

- เป็นการเขียนคำสั่งแปลงข้อมูลจากที่เป็น 11 column จากไฟล์ csv ที่ download มาให้เป็น 3 column csv เพื่อที่จะสามารถอ่านด้วย pandas ได้

```
In [24]: sim_options={'user_based': False, }
         #algos = [ SVDpp(), SlopeOne(), NormalPredictor(), KNNBaseline(), KNNBasic(),  KNNWithZScore(), BaselineOnly(), CoClustering() ]
         algos = [ KNNBaseline(sim_options=sim_options), KNNBasic(sim_options=sim_options), KNNWithMeans() ]

In [26]: benchmark = []
         for algo in algos:
             results = cross_validate(algo, data, measures=['RMSE', 'MAE'], cv=10, verbose=False)

             # Get results & append algorithm name
             tmp = pd.DataFrame.from_dict(results).mean(axis=0)
             tmp = tmp.append(pd.Series([str(algo).split(' ')[0].split('.')[-1]], index=['Algorithm']))
             benchmark.append(tmp)

         pd.DataFrame(benchmark).set_index('Algorithm').sort_values('test_rmse')

         Estimating biases using als...
         Computing the msd similarity matrix...
         Done computing similarity matrix.
         Estimating biases using als
```

รูปที่ 4.15: คำสั่งอ่าน 3-column csv เป็น dataframe -> dataset transcoder

จากภาพที่ 4.15 สามารถอธิบายได้ดังนี้

- เป็นคำสั่งอ่าน 3 column csv เป็น dataframe และไปเป็น dataset transcoder

Algorithm	test_rmse	test_mae	fit_time	test_time
KNNBaseline	0.913945	0.719058	0.001919	0.000853
KNNBasic	0.978462	0.738293	0.001180	0.000504
KNNWithMeans	1.014723	0.809938	0.002225	0.000641

รูปที่ 4.16: Algorithm name ที่ใช้ในการประมวณผลของระบบ ได้แก่ KNNBaseline(), KNNBasic(), KNNWithMeans()

จากภาพที่ 4.16 สามารถอธิบายได้ดังนี้

- เป็นตารางที่ได้จากการรันอัลกอริทึม ค่าที่ได้จะประกอบไปด้วย Root Mean Squared Error (RMSE) หรือ Root Mean Squared Deviation (RMSD), Mean Abolute Error (MAE) หรือ L1 Loss คือ การคำนวน Error, fit$_t imetest_t ime$

```
In [30]: from surprise import accuracy

In [31]: k = 4
         knn = KNNWithMeans(k=k, sim_options={'user_based': False, })

In [32]: trainset, testset = train_test_split(data, test_size=.25)
         len(trainset.ur)

Out[32]: 60

In [33]: trainset = data.build_full_trainset()
         _, testset = train_test_split(data, test_size=1.0)

In [34]: knn.fit(trainset)

         Computing the msd similarity matrix...
         Done computing similarity matrix.

Out[34]: <surprise.prediction_algorithms.knns.KNNWithMeans at 0x159a08adac8>

In [35]: predictions = knn.test(testset)

In [36]: threshhold = 0.3
         corrects = sum([1 for p in predictions if abs(p.r_ui-p.est) < threshhold ])
         size = len(predictions)

In [37]: corrects*100/size

Out[37]: 55.0
```

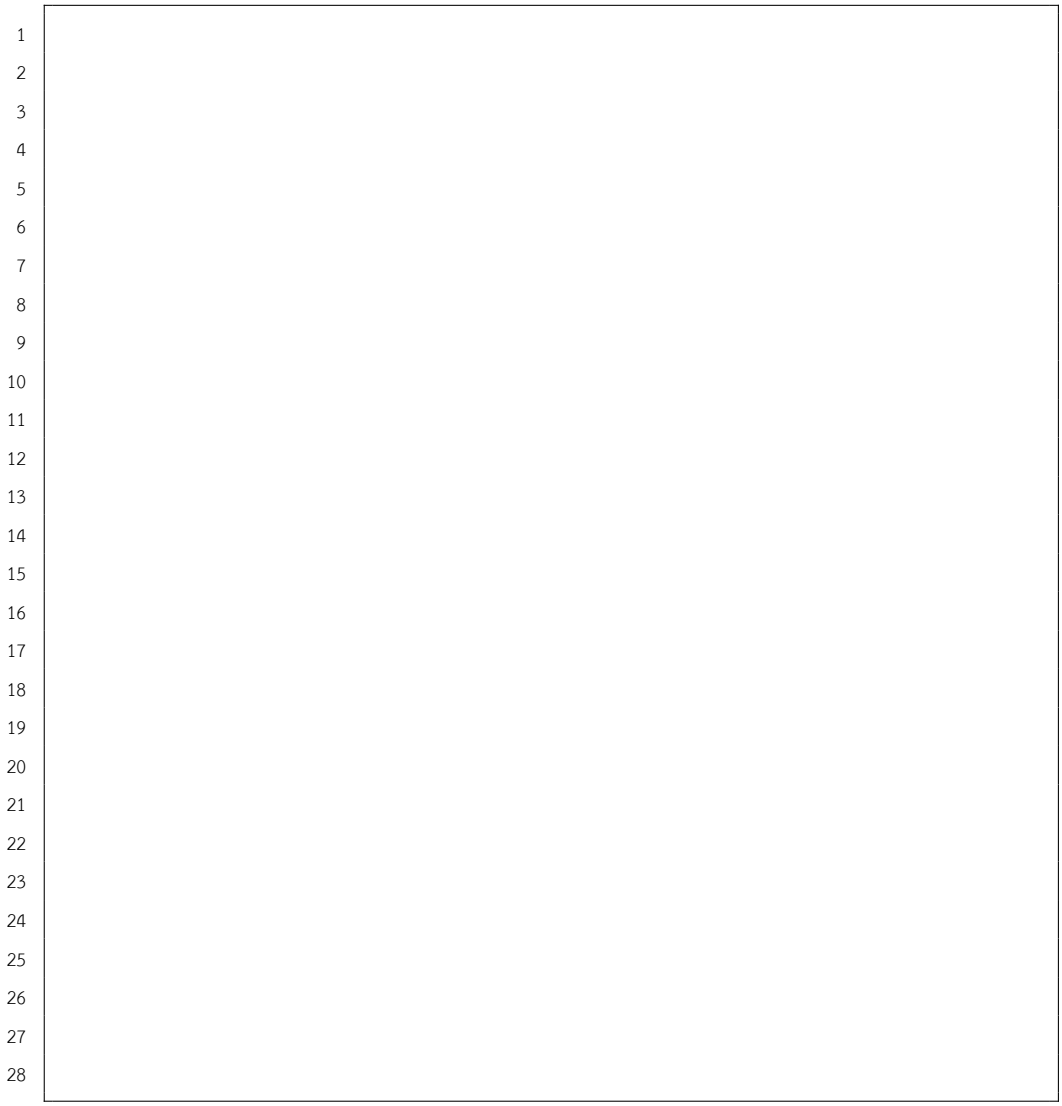
รูปที่ 4.17: การวิเคราะห์ข้อมูลเพื่อนำไปใช้ในฟังก์ชันการแนะเสื้อผ้า

จากภาพที่ 4.17 สามารถอธิบายได้ดังนี้

- บรรทัดที่ In [30] เป็นคำสั่งในการนำเข้าฟังก์ชัน surprise
- บรรทัดที่ In [31] เป็นคำสั่งในการหาค่าการแนะนำสินค่าตาม item base คือการแนะนำตามสินค้า
- บรรทัดที่ In [32] - Out [32] เป็นคำสั่งในการเทรนข้อมูลเสื้อผ้าที่มีในระบบทั้งหมด 60 ชิ้น
- บรรทัดที่ In [33] เป็นคำสั่งในการ set data
- บรรทัดที่ In [34] - Out [34] เป็นการรันข้อมูลที่ได้จากการ train
- บรรทัดที่ In [35] - In[37] เป็นการหาค่าความเป็นไปได้ โดยทดสอบจากข้อมูลที่ได้จากการ train
- บรรทัดที่ Out[37] เป็นค่าที่ได้จากการหาความถูกต้องจาก 100

4.9.2 การนำต้นแบบมาแนะนำเสื้อผ้า

เมื่อผู้ใช้กดปุ่มแนะนำเสื้อผ้า ระบบจะมีรายละเอียดการทำงานดังนี้



รูปที่ 4.18: การทำงานของระบบเมื่อผู้ใช้กดดูรายการแนะนำเสื้อผ้า

จากภาพที่ 4.18 โครงสร้างของไฟล์ views.py สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 เมื่อผู้ใช้กดปุ่มดูรายการแนะนำเสื้อผ้าระบบจะเข้ามาทำงานในฟังก์ชัน recommend()
- บรรทัดที่ 2-3 หากผู้ใช้ยังไม่ทำการสมัครสมาชิก ระบบจะส่งข้อมูลไปยังหน้าสมัครสมาชิก
- บรรทัดที่ 4-5 หากผู้ใช้ได้ทำการสมัครสมาชิกและเข้าสู่ระบบเรียบร้อยแล้ว จะแสดงหน้าแนะนำเสื้อผ้า
- บรรทัดที่ 6 เป็นการโหลด model ที่ได้จากการ train เสร็จเรียบร้อยแล้ว
- บรรทัดที่ 7-10 เป็นการรับค่าข้อมูลสินค้าโปรดของผู้ใช้งาน และสินค้าที่จะทำการแนะนำ

- บรรทัดที่ 11-16 เป็นการแสดงรายการแนะนำเสื้อผ้าตามสินค้าที่ผู้ใช้ชื่นชอบ โดยเก็บค่าเป็นแบบ array โดยที่ถ้าเป็นสมาชิกและมีประวัติการกดบันทึกสินค้าที่ถูกใจ ระบบจะแนะนำสินค้าตาม item
- บรรทัดที่ 17-23 เป็นการแสดงรายการแนะนำเสื้อผ้าให้แก่ผู้ใช้ที่สมัครสมาชิกเป็นผู้ใช้งานใหม่ และเข้าสู่ระบบเรียบร้อยแล้ว ระบบจะแนะนำสินค้าตาม item โดยจะแสดงรายการสินค้าที่มีผู้ชื่นชอบมากที่สุดให้แก่ผู้ใช้งานรายใหม่ได้ทราบ โดยเก็บค่าเป็นแบบ array
- บรรทัดที่ 24 เป็นการส่งค่าที่ได้ทั้งหมดไปแสดงยังหน้า recommend.html
- บรรทัดที่ 25-27 เป็นการกำหนดค่าตัวแปรที่จะให้แสดงในหน้า recommend.html

บทที่ 5

การทดสอบระบบ

หลังจากที่ได้ผ่านขั้นตอนสำหรับการพัฒนาระบบทั้งหมดแล้ว ขั้นตอนต่อไปคือการทดสอบระบบเพื่อตรวจสอบการทำงานของแต่ละฟังก์ชันกันรวมถึงตรวจสอบความถูกต้องของการทำงาน ทั้งระบบเพื่อให้การพัฒนาโปรแกรมเป็นไปอย่างมีประสิทธิภาพ ทดสอบระบบเพื่อหาข้อผิดพลาด และแก้ไขข้อบกพร่องนั้นก่อนการใช้งานจริง โดยในการทดสอบระบบจะแบ่งเป็น 2 ส่วน ได้แก่ การทดสอบในส่วนฟังก์ชันของระบบ และการทดสอบประสิทธิภาพของโมเดล

5.1 การทดสอบในส่วนฟังก์ชันก์ของระบบ

5.1.1 ผลการทดสอบการสมัครสมาชิกผู้ใช้งาน

ตารางที่ 5.1: ผลการทดสอบการสมัครสมาชิกผู้ใช้งาน

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบการสมัครสมาชิกผู้ใช้งาน	ผู้ใช้กดปุ่มสมัครสมาชิก	ระบบแสดงหน้าสมัครสมาชิก ซึ่งข้อมูลที่แสดงในหน้านั้นจะประกอบไปด้วย ชื่อผู้ใช้ รหัสผ่าน อีเมล์ วัน/เดือน/ปีเกิด รูปร่าง สีผิว ปุ่มเลือกรูปโปรไฟล์ และปุ่มสมัครสมาชิกผู้ใช้งาน
	ผู้ใช้กดปุ่มสมัครสมาชิกโดยกรอกข้อมูลไม่ครบทุกช่องตามที่กำหนดและกดปุ่มสมัครสมาชิก	ระบบแสดง error ในช่องที่ผู้ใช้ยังไม่ได้ทำการกรอก
	ผู้ใช้กรอกข้อมูลครบทุกช่องและกดปุ่มสมัครสมา-ชิก	ระบบ แสดง หน้า สมัคร สมาชิก เหมือน โดย ไม่มี error

5.1.2 ผลการทดสอบการเข้าสู่ระบบผู้ใช้งาน

ตารางที่ 5.2: ผลการทดสอบการเข้าสู่ระบบผู้ใช้งาน

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบการเข้าสู่ระบบผู้ใช้งาน	ผู้ใช้เข้ามาในหน้าเข้าสู่ระบบ	ระบบแสดงหน้าเข้าสู่ระบบ
	ผู้ใช้กดปุ่มเข้าสู่ระบบโดยไม่กรอกชื่อผู้ใช้และรหัสผ่าน	ระบบแสดงหน้าเข้าสู่ระบบเหมือนเดิม
	ผู้ใช้กดปุ่มเข้าสู่ระบบโดยไม่กรอกชื่อผู้ใช้และรหัสผ่านไม่ถูกต้อง	ระบบ แสดง หน้า เข้า สู่ ระบบ เหมือน เดิม โดย ไม่มี error
	ผู้ใช้กดปุ่มเข้าสู่ระบบโดยไม่กรอกชื่อผู้ใช้และรหัสผ่านถูกต้อง	ระบบแสดงหน้าเข้าใช้งานระบบได้พร้อมบอกชื่อผู้ใช้ที่เข้าสู่ระบบนั้น

5.1.3 ผลการทดสอบหน้าเสื้อผ้าทั้งหมด

ตารางที่ 5.3: ผลการทดสอบหน้าเสื้อผ้าทั้งหมด

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบหน้าเสื้อผ้าทั้งหมด	ผู้ใช้กดเมนูเสื้อผ้าทั้งหมด	ระบบ แสดง หน้า รายการ เสื้อผ้า ทั้งหมด ซึ่ง แบ่ง หมวด หมู่ ออก เป็น ทั้งหมด 6 ประเภท ได้แก่ เสื้อ กางเกง กระโปรง เดรส สูท และอื่นๆ
	ผู้ใช้กดเลือก 1 ประเภท	ระบบ แสดง หน้า รายการ เสื้อผ้า ตาม ประเภท ที่ เลือก
	ผู้ใช้กดดูรายละเอียดสินค้า	ระบบแสดงหน้ารายละเอียดสินค้า

5.1.4 ผลการทดสอบหน้ารายละเอียดสินค้า

ตารางที่ 5.4: ผลการทดสอบหน้ารายละเอียดสินค้า

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบรายละเอียดสินค้า	ผู้ใช้กดปุ่มดูรายละเอียดสินค้า	ระบบแสดงหน้ารายละเอียดสินค้า
	ผู้ใช้กดปุ่มเพิ่มเป็นสินค้าโปรด พร้อมเลือกระดับความชอบสินค้านั้น	ระบบจะทำการเพิ่มสินค้าชิ้นนั้นเข้าไปในรายการสินค้าโปรด พร้อมบันทึกระดับความชอบสินค้านั้น และสถานะสินค้าก็จะเปลี่ยนเป็นเพิ่มเป็นสินค้าโปรดแล้ว ในกรณีที่ผู้ใช้งานยังไม่สมัครสมาชิกและเข้าสู่ระบบ ระบบจะไม่แสดงปุ่มเพิ่มเป็นสินค้าโปรด

5.1.5 ผลการทดสอบหน้าร้านค้าทั้งหมด

ตารางที่ 5.5: ผลการทดสอบหน้าร้านค้าทั้งหมด

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบหน้าร้านค้าทั้งหมด	ผู้ใช้กดปุ่มเมนูร้านค้าทั้งหมด	ระบบแสดงหน้ารายการร้านค้าทั้งหมด
	ผู้ใช้กดปุ่มรายละเอียดร้านค้า	ระบบ แสดง หน้า ราย ละเอียด ข้อมูล ร้าน ค้า นั้น พร้อมแสดงรายการสินค้าที่มีในร้าน

5.1.6 ผลการทดสอบหน้ารายละเอียดร้านค้า

ตารางที่ 5.6: ผลการทดสอบหน้ารายละเอียดร้านค้า

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบหน้ารายละเอียดร้านค้า	ผู้ใช้กดปุ่มดูรายละเอียดร้านค้า	ระบบ แสดง หน้า ราย ละเอียด ข้อมูล ร้าน ค้า นั้น พร้อมแสดงรายการสินค้าที่มีในร้าน
	ผู้ใช้กดปุ่มเพิ่มเป็นร้านโปรด	ระบบจะทำการเพิ่มร้านค้านั้นเข้าไปในรายการร้านโปรดและสถานะร้านค้าก็จะเปลี่ยนเป็น เพิ่มเป็นร้านโปรดแล้ว ในกรณีที่ผู้ใช้งานยังไม่สมัครสมาชิกและเข้าสู่ระบบ ระบบจะไม่แสดงปุ่มเพิ่มเป็นร้านโปรด

5.1.7 ผลการทดสอบหน้าข้อมูลผู้ใช้งาน

ตารางที่ 5.7: ผลการทดสอบหน้าข้อมูลผู้ใช้งาน

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบหน้าข้อมูลผู้ใช้งาน	ผู้ใช้กดที่ชื่อผู้ใช้และกดเลือกข้อมูลผู้ใช้งาน	ระบบแสดงหน้าข้อมูลผู้ใช้งาน
	ผู้ใช้กดปุ่มแก้ไขข้อมูล	ระบบแสดงหน้าแก้ไขข้อมูลผู้ใช้งาน ซึ่งข้อมูลที่แสดงในหน้านั้นจะประกอบไปด้วย ชื่อผู้ใช้ รหัสผ่าน อีเมล์ วัน/เดือน/ปีเกิด รูปร่าง สีผิว ปุ่มเลือกรูปโปรไฟล์ และปุ่มบันทึกการแก้ไขข้อมูล
	ผู้ใช้กดปุ่มรายการสินค้าโปรด	ระบบแสดงหน้ารายการสินค้าโปรด แสดงรายการเสื้อผ้าที่ผู้ใช้ได้กดเพิ่มเป็นสินค้าโปรด และสามารถกดลบรายการสินค้าโปรดได้
	ผู้ใช้กดปุ่มรายการร้านโปรด	ระบบแสดงหน้ารายการร้านโปรด แสดงรายการร้านค้าที่ผู้ใช้ได้กดเพิ่มเป็นร้านโปรด และสามารถกดลบรายการร้านโปรดได้

5.1.8 ผลการทดสอบหน้าสมัครสมาชิกร้านค้า

ตารางที่ 5.8: ผลการทดสอบหน้าสมัครสมาชิกร้านค้า

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบหน้าสมัครสมาชิกร้านค้า	ร้านค้ากดที่ชื่อผู้ใช้และกดเพิ่มร้านค้า	ระบบแสดงหน้าสมัครสมาชิกร้านค้า ซึ่งจะต้องทำการกรอกข้อมูลร้านดังนี้ ชื่อร้านค้า อีเมล์ ที่อยู่ร้านค้า เบอร์โทรติดต่อ Facebook ปุ่มเลือกรูปร้านค้า และปุ่มสมัครสมาชิกร้านค้า
	ร้านค้ากดปุ่มสมัครสมาชิกโดยกรอกข้อมูลไม่ครบทุกช่องตามที่กำหนดและกดปุ่มสมัครสมาชิก	ระบบแสดง error ในช่องที่ผู้ใช้ยังไม่ได้ทำการกรอก

5.1.9 ผลการทดสอบหน้าข้อมูลร้านค้า

ตารางที่ 5.9: ผลการทดสอบหน้าข้อมูลร้านค้า

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบหน้าข้อมูลร้านค้า	ร้านค้ากดที่ชื่อผู้ใช้และกดเลือกข้อมูลร้านค้า	ระบบแสดงหน้าข้อมูลร้านค้า
	ร้านค้ากดแก้ไขข้อมูลร้านค้า	ระบบ แสดง หน้า แก้ไข ข้อมูล ร้าน ค้า ซึ่ง ข้อมูล ที่ แสดง ใน หน้า นี้ จะ ประกอบ ไป ด้วย ชื่อ ผู้ใช้ รหัส ผ่าน ชื่อร้านค้า อีเมล์ ที่อยู่ร้านค้า เบอร์โทรติดต่อ Facebook ปุ่มเลือกรูปร้านค้า และปุ่มแก้ไขข้อมูล ร้านค้า
	ร้านค้ากดที่ชื่อผู้ใช้และกดเลือกข้อมูลร้านค้า	ระบบ แสดง หน้า จัดการ ข้อมูล สินค้า ซึ่ง ข้อมูล ที่ แสดง ใน หน้า นี้ จะ ประกอบ ไป ด้วย รายการ สินค้า ทั้งหมดที่มีในร้านที่สามารถทำการแก้ไขสินค้าและ ลบสินค้าได้
	ร้านค้ากดปุ่มเพิ่มสินค้าในหน้าจัดการข้อมูลสินค้า	ระบบแสดงหน้าเพิ่มสินค้า ซึ่งข้อมูลที่แสดงในหน้า นี้จะประกอบไปด้วย ชื่อร้านค้า ชื่อสินค้า ประเภท สินค้า จำนวนสินค้า ขนาดสินค้า สีสินค้า ราคา ปุ่ม เลือกรูปสินค้า และปุ่มบันทึก
	ร้านค้ากดปุ่มดูรายการคนที่ถูกใจสินค้า	ระบบแสดงหน้ารายการคนที่ถูกใจเสื้อผ้า
	ร้านค้ากดปุ่มดูรายการคนที่ถูกใจร้านค้า	ระบบแสดงหน้ารายการคนที่ถูกใจร้านค้า

5.1.10 ผลการทดสอบหน้าแนะนำเสื้อผ้า

ตารางที่ 5.10: ผลการทดสอบหน้าแนะนำเสื้อผ้า

การทดสอบ	เงื่อนไขการทดสอบ	ผลการทดสอบ
ทดสอบหน้าแนะนำเสื้อผ้า	ผู้ใช้ยังไม่ทำการสมัครสมาชิกแล้วกดปุ่มแนะนำเสื้อผ้า	ระบบแสดงหน้าสมัครสมาชิกผู้ใช้งาน
	ผู้ใช้ทำการสมัครสมาชิกแล้ว แต่ยังไม่เข้าสู่ระบบแล้วกดปุ่มแนะเสื้อผ้า	ระบบแสดงหน้าสมัครสมาชิกผู้ใช้งาน
	ผู้ใช้เข้าสู่ระบบเรียบร้อยแล้วกดปุ่มแนะนำเสื้อผ้าในกรณียังไม่เพิ่มรายการเสื้อผ้าที่ถูกใจ	ระบบแสดงหน้าแนะนำเสื้อผ้า โดยในกรณีนี้จะแสดงรายการเสื้อผ้าที่มีคนใช้คะแนนมากที่สุดมาแนะนำ 9 ชิ้น
	ผู้ใช้เข้าสู่ระบบเรียบร้อยแล้วกดปุ่มแนะนำเสื้อผ้าในกรณีมีรายการเสื้อผ้าที่ถูกใจแล้ว	ระบบแสดงหน้าแนะนำเสื้อผ้า โดยในกรณีนี้จะแสดง รายการ เสื้อผ้า ตาม สิ้น ค้า ที่ ผู้ ใช้ ถูกใจ มาแนะนำ 9 ชิ้น

5.2 การทดสอบประสิทธิภาพของโมเดล

5.2.1 ผลการทดสอบประสิทธิภาพโดยใช้ RMSE และ MAE

ผลการทดสอบประสิทธิภาพของชุดข้อมูล FavoriteCloth จำนวน 300 รายการ เมื่อทดสอบด้วย เทคนิค KNNBaseline, KNNBasic, และ KNNWithMeans แล้ว ผลลัพธ์ของการประเมินประสิทธิภาพด้วยวิธี K-fold crossvalidtion โดยแบ่งเป็น 10 กลุ่มทดสอบ แล้วแสดงค่าเฉลี่ย จะได้ผลลัพธ์ดังตารางต่อไปนี้

ตารางที่ 5.11: ผลลัพธ์ของการประเมินประสิทธิภาพด้วยวิธี K-fold crossvalidtion

เทคนิค	RMSE	MAE	เวลาฝึก	เวลาทดสอบ
KNNBaseline	0.913945	0.719058	0.001919	0.000853
KNNBasic	0.978462	0.738293	0.001180	0.000504
KNNWithMeans	1.014723	0.809938	0.002225	0.000641

5.2.2 ผลการทดสอบความถูกต้องของโมเดลโดยวัดจากประเมินของ Rating

การวัดประสิทธิภาพนี้จะใช้ data ทั้งหมด 300 รายการเพื่อสร้างต้นแบบ จากนั้นนับจำนวนค่าประเมินที่ยอมรับได้ เมื่อเปรียบเทียบกับ rating จริง โดยกำหนดช่วงค่าที่ยอมรับได้ h เรียกว่าค่า threshold เช่นถ้า h = 0.5 เมื่อ rating จริงในรายการ j มีค่า 4 ดาว แสดงว่าช่วงค่าประเมินที่ยอมรับได้นั้นจะอยู่ในช่วง 3 ดาวครึ่ง ถึง สี่ดาวครึ่ง จากนั้นค่าร้อยละความถูกต้อง จะได้ผลลัพธ์ดังตารางต่อไปนี้

ตารางที่ 5.12: ผลลัพธ์การทดสอบประสิทธิภาพของ 3 เทคนิคด้วยช่วงค่าและแปลงเป็นร้อยละ

เทคนิค	h=0.3	h=0.4	h=0.5	h=0.6
KNNBaseline	47.67	63.67	75.33	86.00
KNNBasic	49.00	56.00	73.67	84.33
KNNWithMeans	46.67	59.67	68.0	79.67

สรุปผลลัพธ์ที่ได้จากการทดสอบประสิทธิภาพของ 3 เทคนิคด้วยช่วงค่าและแปลงเป็นร้อยละ เป็น h= 0.3 0.4 0.5 และ 0.6 ผลที่ได้คือ KNN ที่ดีที่สุดที่ควรเลือกใช้คือ คือ KNNBaseline ค่า h= 0.4 0.5 และ 0.6 และ ผลลัพธ์การทดสอบประสิทธิภาพตามลำดับคือ 63.67 75.33 และ 86.00 เพราะค่าที่ได้มีเป็นค่าที่ยอมรับได้มากที่สุดเมื่อแปลงเป็นร้อยละแล้ว

บทที่ 6

สรุปและข้อเสนอแนะ

การดำเนินโครงงานเพื่อพัฒนาระบบเว็บแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรีนี้ พบว่าระบบสามารถทำงานได้ตามที่วิเคราะห์และออกแบบไว้ แต่ก็พบปัญหาและอุปสรรคระหว่างการพัฒนา ในบทนี้ผู้พัฒนาจึงขอสรุปความสามารถของระบบชี้แจงปัญหาและอุปสรรค พร้อมเสนอ แนวทางในการพัฒนระบบเว็บแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรีต่อตามลำดับ

6.1 สรุปความสามารถของระบบ

โดยแบ่งความสามารถของระบบตามประเภทของผู้ใช้งานดังนี้

1. ผู้ใช้งาน
    - สามารถสมัครสมาชิกได้
    - สามารถเข้าสู่เข้าสู่ระบบได้
    - สามารถแก้ไขข้อมูลผู้ใช้งานได้
    - สามารถดูรายการเสื้อผ้าจากร้านค้าได้
    - สามารถดูรายการร้านค้าได้
    - สามารถดูรายการแนะนำเสื้อผ้าได้
    - สามารถบันทึกรายการเสื้อผ้าที่ถูกใจได้
    - สามารถบันทึกรายการร้านค้าที่ถูกใจได้
    - สามารถดูบันทึกรายการเสื้อผ้าที่ถูกใจได้
    - สามารถดูบันทึกรายการร้านค้าที่ถูกใจได้
    - สามารถดูค้นหาเสื้อผ้าได้
2. ร้านค้า
    - สามารถสมัครสมาชิกได้
    - สามารถเข้าสู่ระบบได้
    - สามารถแก้ไขข้อมูลร้านค้าได้
    - สามารถเพิ่ม ลบ แก้ไข ข้อมูลเสื้อผ้าในร้านค้าได้
    - สามารถดูรายการคนที่ถูกใจเสื้อผ้าได้
    - สามารถดูรายการคนที่ถูกใจร้านค้าได้
3. ผู้ดูแลระบบ

- สามารถเข้าสู่ระบบได้
- สามารถเพิ่ม ลบ แก้ไข ข้อมูลร้านค้าได้
- สามารถเพิ่ม ลบ แก้ไข ข้อมูลผู้ใช้งานได้

6.2 ปัญหาและอุปสรรคในการพัฒนา

1. ข้อมูลในการทดสอบมีจำนวนน้อย ทำให้ความถูกต้องของการทำนายมีความแม่นยำน้อย แนวทางการแก้ไข : เพิ่มชุดข้อมูลที่มีความสัมพันธ์กันเพื่อให้ผลการทำนายมีความแม่นยำมากยิ่งขึ้น
2. ผู้พัฒนาได้เขียน server สำหรับทำการ train ข้อมูล และเป็นที่สำหรับเก็บข้อมูลแม่แบบ (model) ที่ได้จากการเทรนข้อมูลออกมาเป็นไฟล์นามสกุล yaml ให้ฝั่ง client เรียกใช้งานต่อได้ แต่ปัญหาที่พบคือต้องเขียนข้อมูลขึ้นมาเอง แนวทางการแก้ไข : เขียนไฟล์ข้อมูลใน excel แล้วโหลดเป็น .csv ไปใช้ server ขึ้นมาเพื่อให้ทำการ train ข้อมูลและทำนาย (predict) ข้อมูลให้เสร็จสิ้นทางฝั่ง server แล้วฝั่ง client มีหน้าที่ในการรอรับข้อมูลอย่างเดียว

6.3 แนวทางการพัฒนาต่อ

1. เพิ่มในส่วนของการเทรนข้อมูลในรายการแนะนำเสื้อผ้า ให้ผู้ดูแลระบบสามารถกดเทรนได้อัตโนมัติ
2. เพิ่มส่วนติดต่อกับสังคมออนไลน์ ได้แก่ Facebook Twitter และ Instagram
3. เพิ่มในส่วนของฟังก์ชันของผู้ใช้งานให้มีการสามารถซื้อขายสินค้าได้
4. เพิ่มในส่วนของฟังก์ชันของผู้ใช้งานให้มีการสามารถรีวิวสินค้าได้

บรรณานุกรม

[1] AmplySoft (2556). เริ่มต้นการเขียนเว็บไซต์ ทำเว็บไซต์ด้วยภาษา python กับ django framework [ออนไลน์]. สืบค้นเมื่อ มีนาคม 2563. จาก http://www.amplysoft.com/knowledge/what-is-django-framework-python.html. .

[2] Microsoft (2560). รู้จักกับ visual studio code (วิชวล สตูดิโอ โค้ด) โปรแกรมฟรีจากค่ายไมโครซอฟท์ [ออนไลน์]. สืบค้นเมื่อ มีนาคม 2563. จาก https://www.mindphp.com/บทความ/microsoft/4829-visual-studio-code.html. .

[3] Blogger (2561). ทดลองใช้ pythonanywhere สร้าง django web app อย่างง่าย [ออนไลน์]. สืบค้นเมื่อ มีนาคม 2563. จาก http://s5901012610091.blogspot.com/2018/02/pythonanywhere-django-web-app.html. .

[4] Thiengkunakrit, W. (2560). เริ่มพัฒนา web application กับภาษา python ด้วย django framework [ออนไลน์]. สืบค้นเมื่อ มีนาคม 2563. จาก https://www.codeburst.io/เริ่มพัฒนา-web-application-กับภาษา-python-ด้วย-django-framework-38ce132ac706. .

[5] bccrwp.org (2562). Sqlite เทียบกับ mysql เทียบกับ postgresql: การเปรียบเทียบระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ [ออนไลน์]. สืบค้นเมื่อ มีนาคม 2563. จาก https://th.bccrwp.org/compare/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems-8f2704/. .

[6] Sarayut Nonsiri, P. (). ภาษาโปรแกรม python คืออะไร [ออนไลน์]. สืบค้นเมื่อ มีนาคม 2563. จาก https://www.9experttraining.com/articles/python-คืออะไร. .

[7] Vorhies, W. (2560). 5 ประเภทของผู้แนะนำ [ออนไลน์]. สืบค้นเมื่อ มีนาคม 2563. จาก https://www.datasciencecentral.com/profiles/blogs/5-types-of-recommenders. .

[8] Blog, P. (). Evolution of recommendation systems [ออนไลน์]. สืบค้น เมื่อ มีนาคม 2563. จาก https://kchaiso.wordpress.com/ tag/ collaborative-filtering/., dpu.ac.th/ dpurc/ assets/ uploads/ magazine/ e37psnbrivk8g4k.pdf , https://www.developers.ascendcorp.com/สร้าง-recommendation-engine-จาก-collaborative-filtering-3d1d68849102. .

[9] Wongnai [ออนไลน์]. สืบค้นเมื่อ 2563. จาก https://www.wongnai.com/ .

[10] Phiu-Nual, K. (2555). ความหมายและความสำคัญของ system architecture (sa) [ออนไลน์]. สืบค้นเมื่อ มีนาคม 2563. จาก https://goo.gl/6ZhGQo. .

ภาคผนวก

ภาคผนวก ก

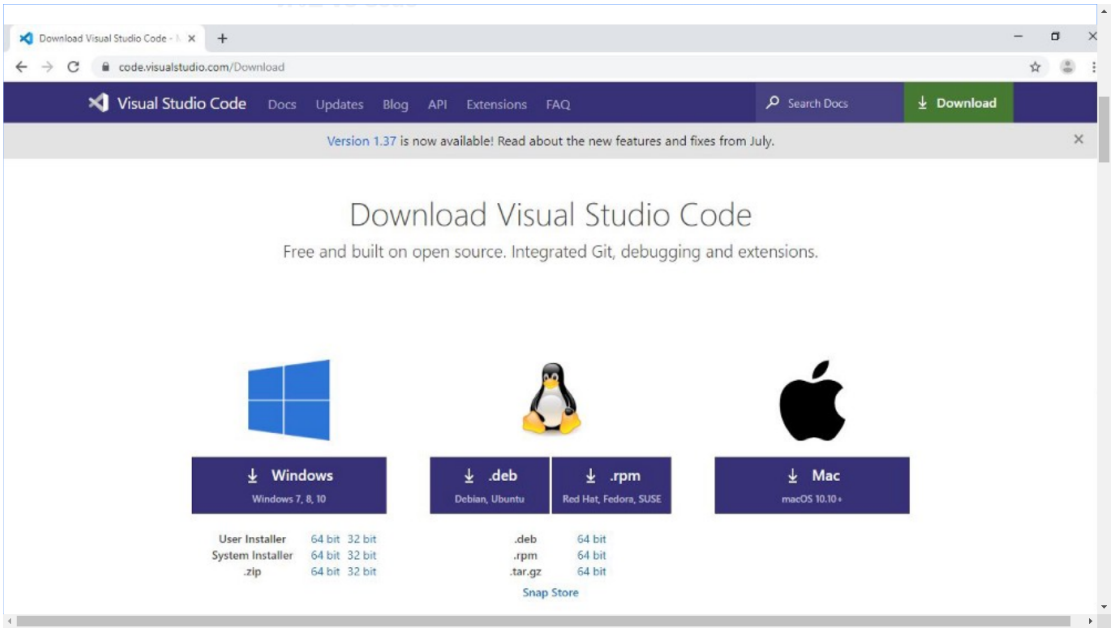
การติดตั้งเครื่องมือที่ใช้พัฒนาโปรแกรม

การติดตั้งเครื่องมือที่ใช้ในการพัฒนาเว็บแอพพลิเคชันระบบแนะนำการแต่งกายสุภาพบุรุษและสุภาพสตรีมีโปรแกรมที่จำเป็นในการพัฒนาระบบดังต่อไปนี้

- การติดตั้ง Visual Studio Code
- การเริ่มต้นใช้ PythonAnywhere

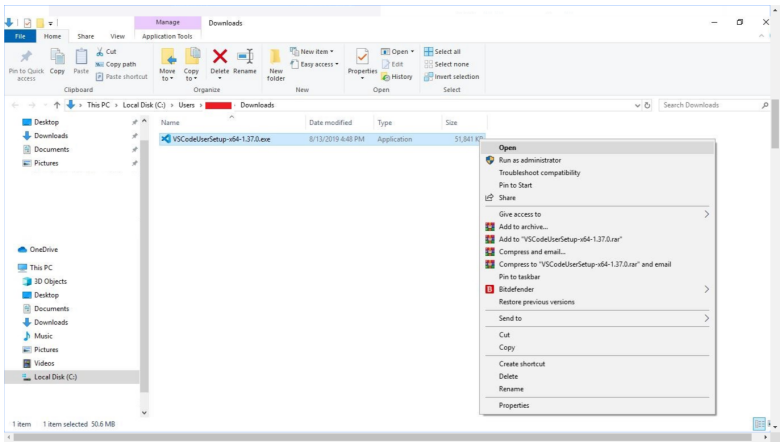
ก.1 ขั้นตอนการติดตั้งโปรแกรม Visual Studio Code หรือ VS Code

1. เข้าไปที่เว็บไซต์ https://code.visualstudio.com/ และ Download โปรแกรม VS Code โดยเลือกให้ตรงกับ OS ของเครื่องคอมพิวเตอร์



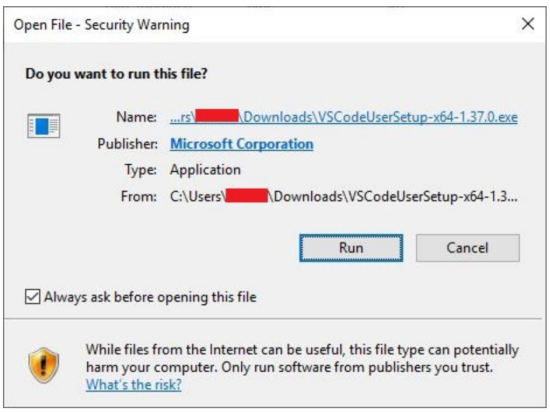
รูปที่ ก.1: หน้าเว็บดาวน์โหลด Visual Studio Code

2. หลังดาวน์โหลดโปรแกรมเสร็จจะได้ไฟล์ ดับเบิ้ลคลิก หรือคลิกขวาและกด “Open” โปรแกรมที่ดาวน์โหลดมา



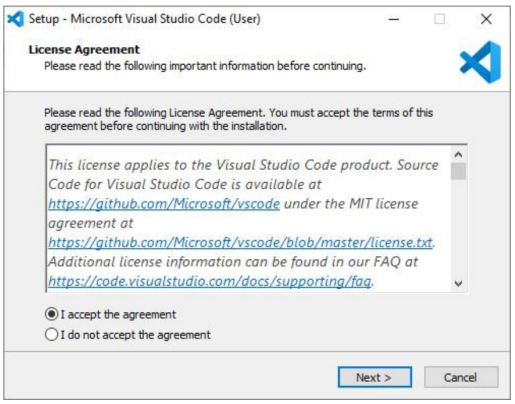
รูปที่ ก.2: ไฟล์โปรแกรมที่โหลดมาได้

3. หลังจากคลิกขวาและกด “Open” โปรแกรมที่ดาวน์โหลดมาแล้วระบบจะแสดงหน้าต่างดังรูปที่ ก.3 คลิกปุ่ม “Run”



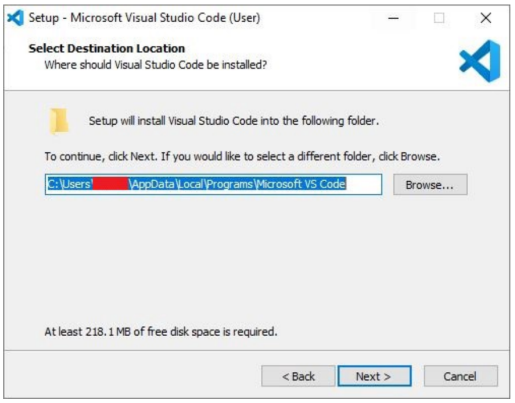
รูปที่ ก.3: หน้าต่างแจ้งเตือนก่อนการติดตั้งโปรแกรม

4. หลังจากคลิก Run แล้ว จะแสดงหน้าต่างดังรูปที่ ก.4 เลือก “I accept the agreement” และคลิกปุ่ม “Next”



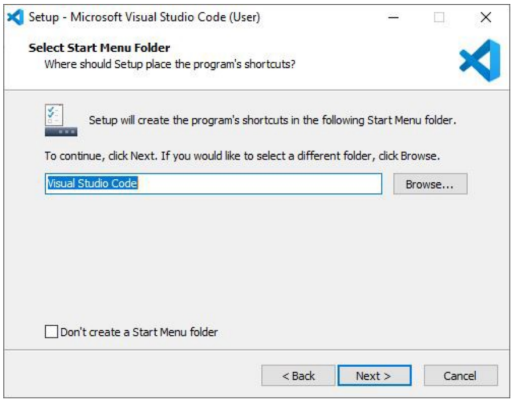
รูปที่ ก.4: หน้าต่างข้อตกลงการใช้งาน Visual Studio Code

5. เลือกพื้นที่ในการจัดเก็บโปรแกรม (แนะนำให้ใช้ Default ที่ให้มา) และคลิกปุ่ม “Next”



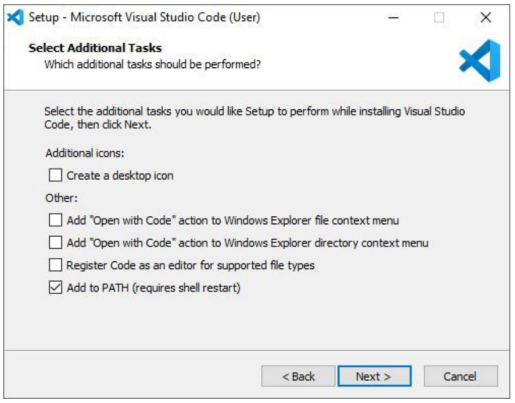
รูปที่ ก.5: หน้าต่างที่จัดเก็บไฟล์ต่างๆ ของ Visual Studio Code

6. จากนั้นจะแสดงหน้าต่างการจัดการซอร์ดคัทของ Visual Studio Code ทำการกด “Next”



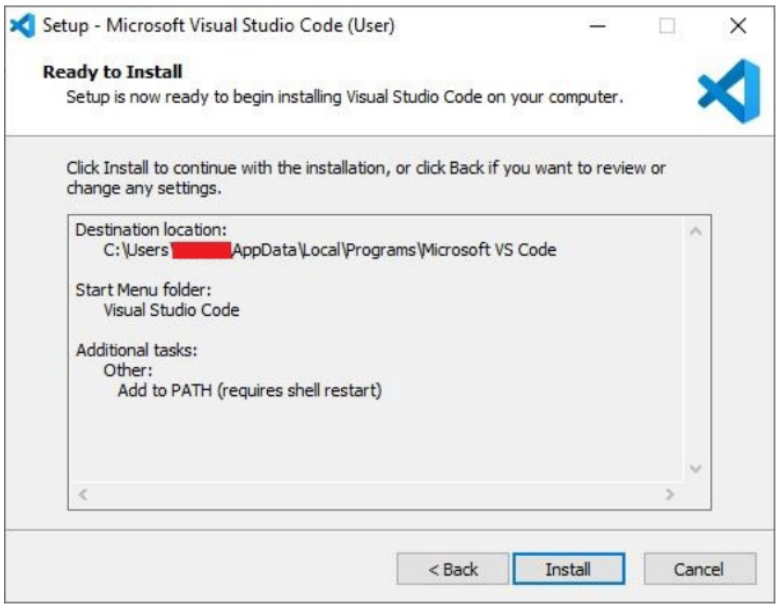
รูปที่ ก.6: หน้าต่างการจัดการซอร์ดคัทของ Visual Studio Code

7. เลือกส่วนเพิ่มงานให้เลือก Create a desktop icon และ Add to PATH (Requires Shell Restart) จากนั้นให้คลิกปุ่ม “Next”



รูปที่ ก.7: หน้าต่างเริ่มทำการติดตั้งทำการกดของ Visual Studio Code

8. คลิกปุ่ม “Install” เพื่อติดตั้งโปรแกรม

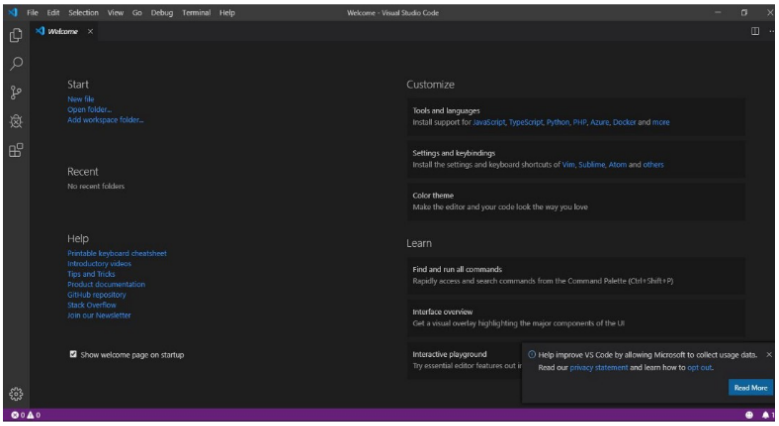


รูปที่ ก.8: หน้าต่างสุดท้ายการติดตั้งของ Visual Studio Code

9. คลิกปุ่ม “Finish” เสร็จสิ้นการติดตั้งโปรแกรม VS Code

รูปที่ ก.9: หน้าต่างการติดตั้งของ Visual Studio Code เสร็จสิ้น

10. Visual Studio Code

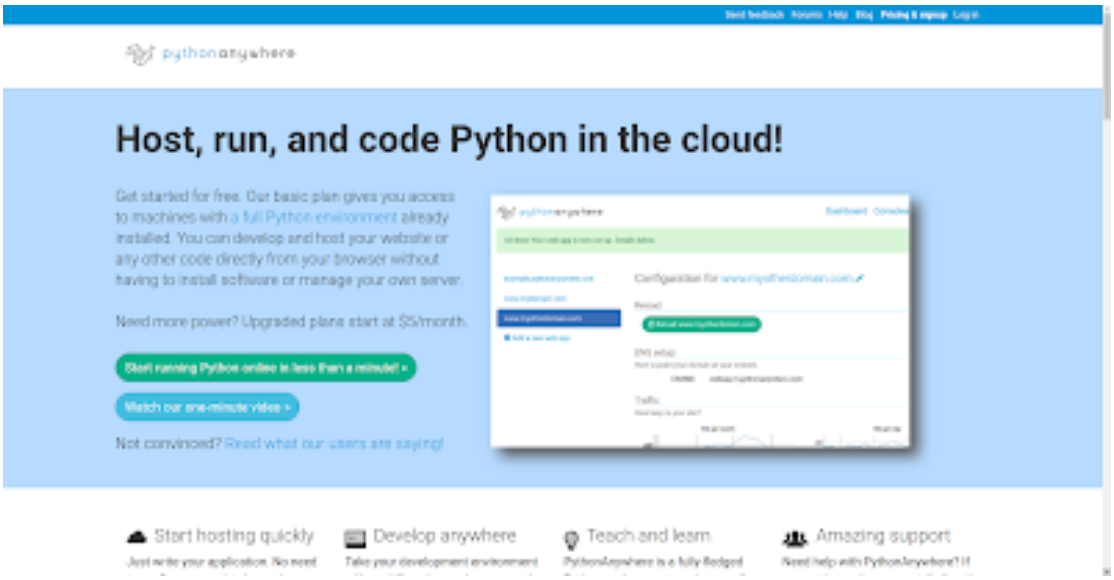


รูปที่ ก.10: หน้าต่างเมื่อเขาโปรแกรมหลังติดตั้งเสร็จของ Visual Studio Code

ก.2 การเริ่มต้นใช้ PythonAnywhere

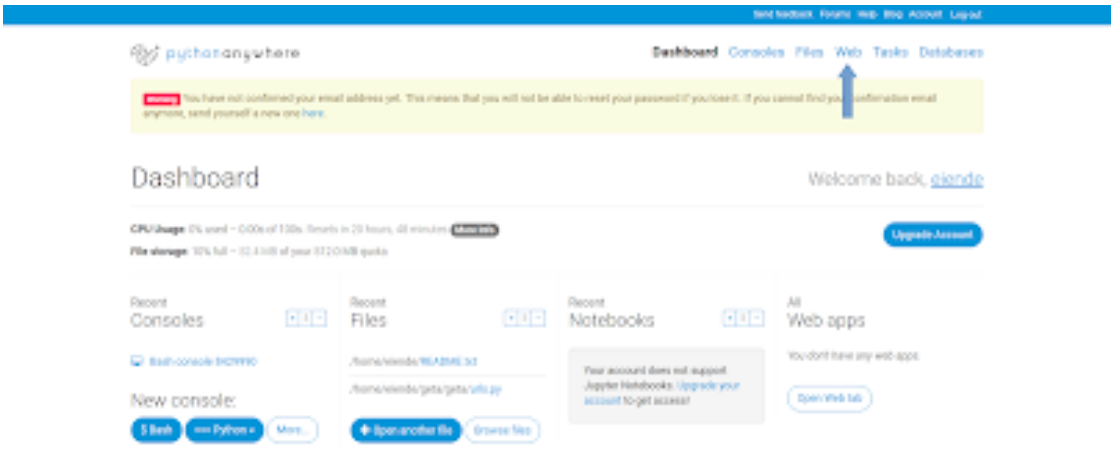
ตัวอย่างแนะนำการใช้งาน pythonanywhere พัฒนา web app

1. pythonanywhere เป็น web hosting สำหรับภาษา Python แบบฟรี โดยจะนำ web app ที่เราสร้างไปเก็บไว้บน cloud ทำให้สามารถเข้าถึง web ได้แบบออนไลน์



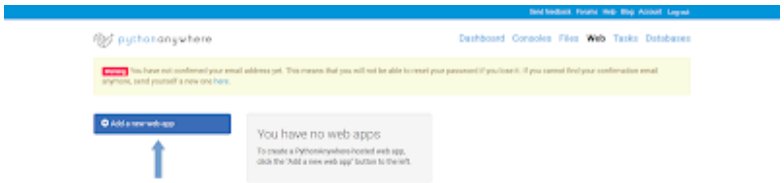
รูปที่ ก.11: หน้าแรกของ pythonanywhere ก่อนเข้าสู่ระบบ

2. เมื่อทำการสมัครสมาชิกและเข้าสู่ระบบเรียบร้อยแล้ว ให้กดไปที่ Web



รูปที่ ก.12: หน้าแรกหลังจากทำการเข้าสู่ระบบเสร็จ

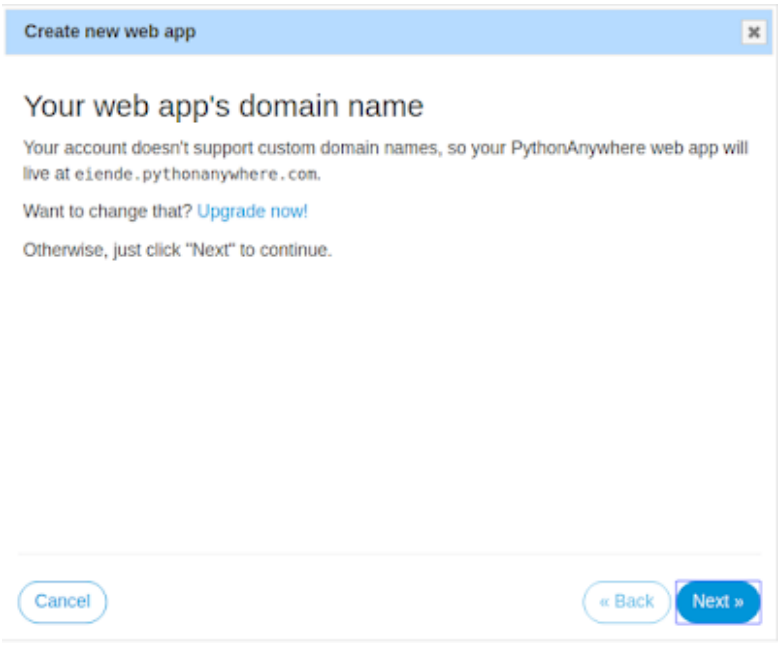
3. กดสร้าง web app ใหม่



รูปที่ ก.13: หน้าเริ่มการสร้างเว็บ

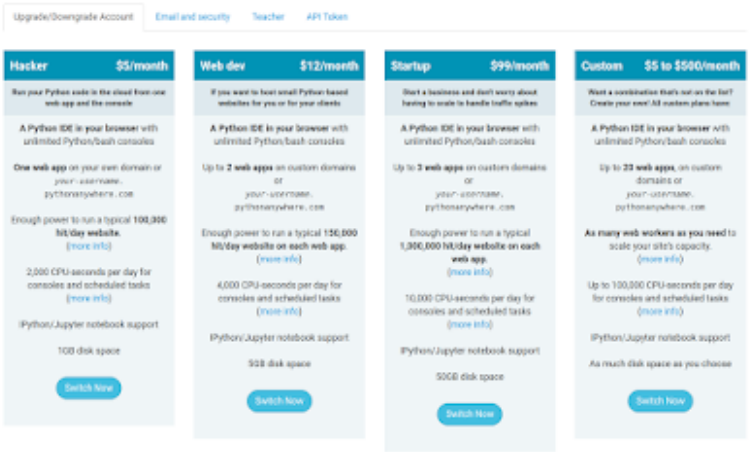
4. จะมีหน้าต่างปรากฏขึ้นบอกว่า domain name ของ web เรานั้นจะมีชื่อเป็น username.pythonanywhere.com ซึ่งในที่นี้คือ lookinggreat.pythonanywhere.com

   ถ้าต้องการเปลี่ยนชื่อ web ต้อง upgrade account แต่ถ้าไม่ต้องการสามารถกด Next เพื่อสร้าง web app ต่อได้เลย



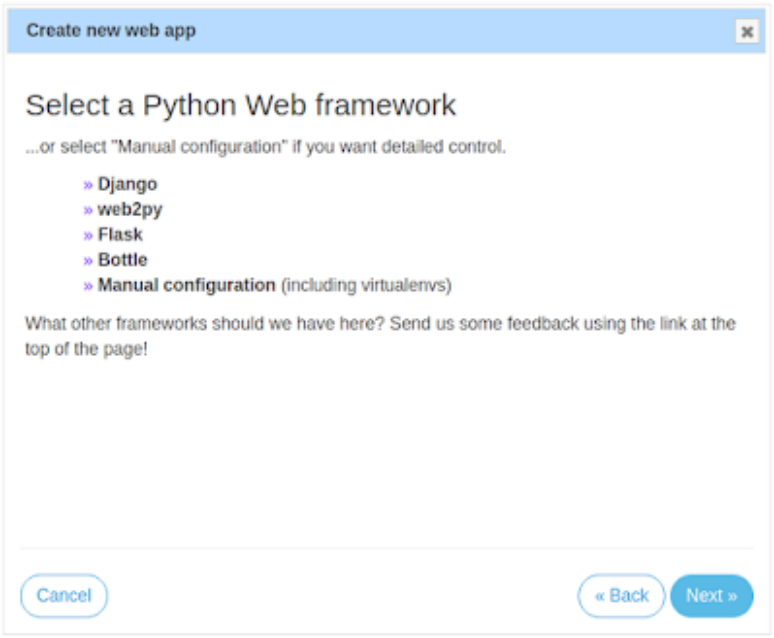
รูปที่ ก.14: หน้าสร้าง web app

5. การ upgrade account ต้องจ่ายเงินเพิ่มแล้วจะได้สิทธิพิเศษตามแต่ละขั้น



รูปที่ ก.15: หน้าเงื่อนไขการ upgrade account

6. ถ้าหากกด Next ต่อมาจะแสดงหน้าต่างให้เลือก framework ที่จะใช้ในการสร้าง web ขึ้นมา ในที่นี้ได้เลือกใช้ Django

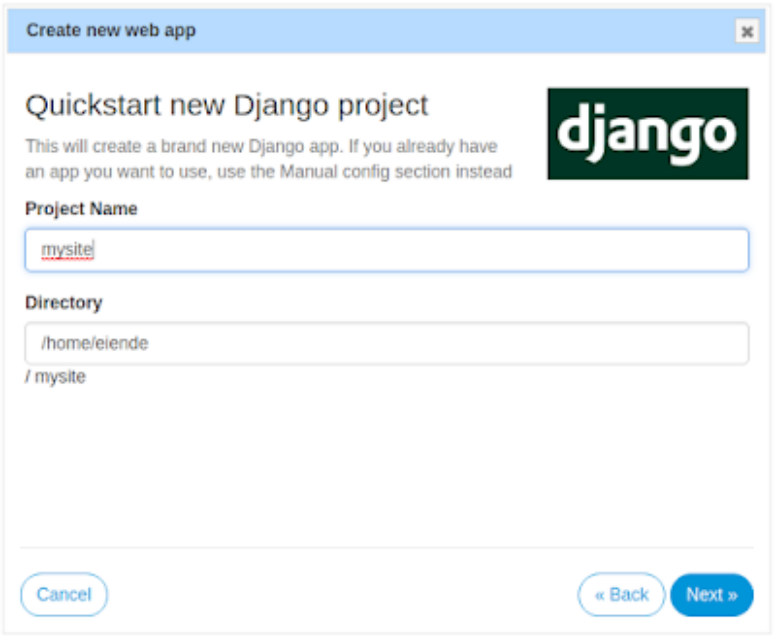


รูปที่ ก.16: หน้าการสร้าง web

7. จากนั้นเลือก version ของ python ที่จะใช้

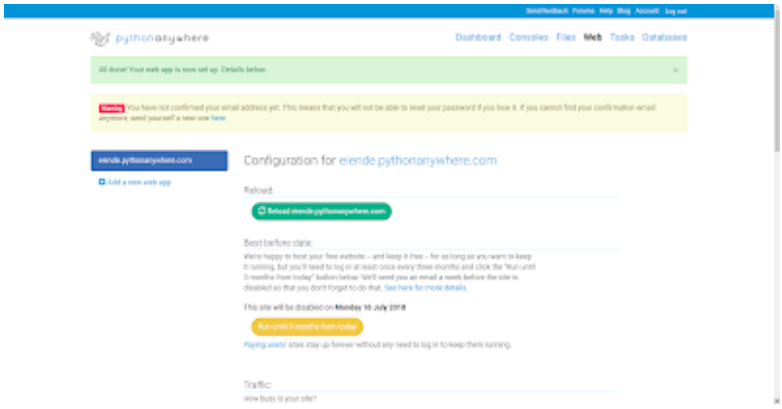
รูปที่ ก.17: หน้าการเลือก version ของ python

8. ตั้งชื่อ project และเลือก directory เก็บ file directory ที่เก็บไฟล์นี้จะถูกเก็บไว้ใน cloud ของ pythonanywhere ไม่เกี่ยวข้องกับพื้นที่เก็บข้อมูลในเครื่องแต่อย่างใด



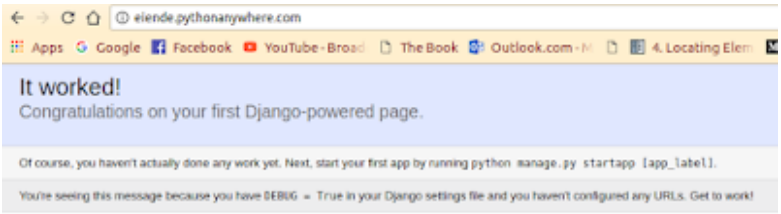
รูปที่ ก.18: หน้าการตั้งชื่อ project และเลือก directory เก็บ file directory

9. จะมีข้อความบ่งบอกว่าสร้าง web app สำเร็จแล้วพร้อมกับรายละเอียดของ web app ของเราด้านล่าง



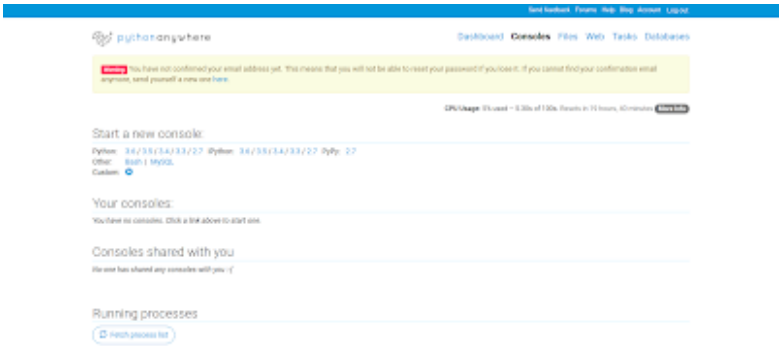
รูปที่ ก.19: หน้าการสร้าง web app สำเร็จ

10. เมื่อลองเข้าลิ้งค์เว็บของเราจะปรากฏข้อความว่าสามารใช้งาน web ได้ของ Django



รูปที่ ก.20: หน้าสามารใช้งาน web ได้ของ Django

11. หากไปที่หน้า Consoles และกด Bash จะเป็นการเปิด Console ขึ้นมา



รูปที่ ก.21: หน้า Consoles

12. Consoles มีหน้าที่คล้าย terminal สามารถลง modules เพิ่มเติมได้จากการพิมพ์คำสั่งในหน้านี้

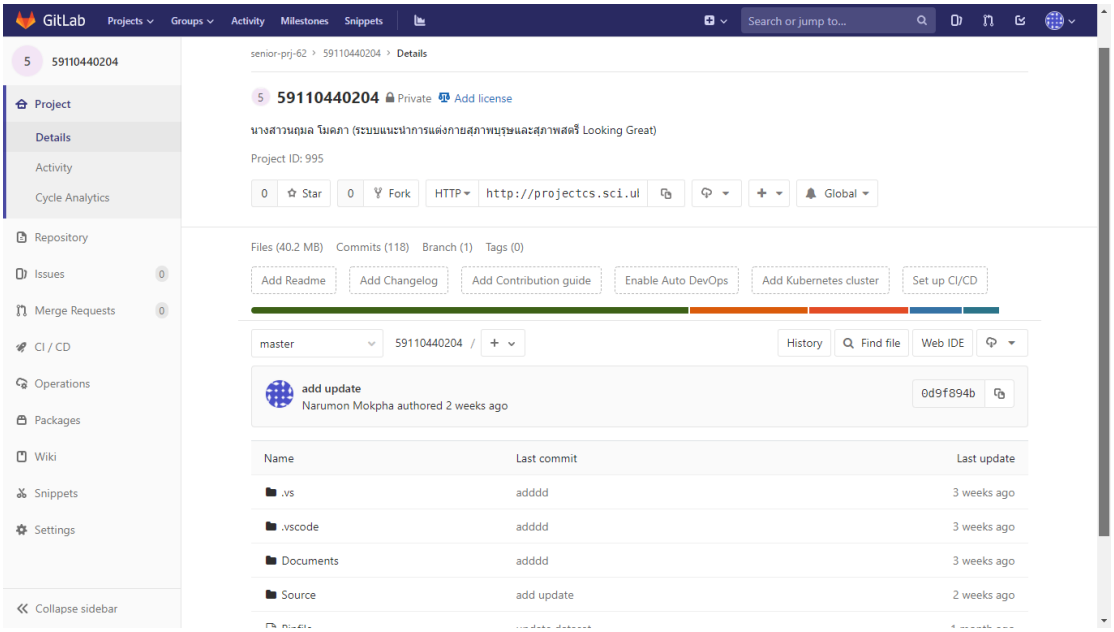
รูปที่ ก.22: หน้าการเปิด Console ขึ้นมา

ภาคผนวก ข

คู่มือการติดตั้งระบบ

คู่มือการติดตั้งระบบ ในการติดตั้งเพื่อใช้งานเว็บแอปพลิเคชันระบบแนะการแต่งกายสุภาพบุรุษและสุภาพสตรี สามารถทำได้โดยมีขั้นตอนดังนี้

1. สามารถ Clone Project ได้ที่ http://projectcs.sci.ubu.ac.th/senior-prj-62/59110440204.git



รูปที่ ข.1: หน้าเว็บ Reposity ของโปรเจ็ค

2. เข้าไปใน 59110440204แล้วใช้คำสั่ง migrate สร้างตารางตามลำดับดังต่อไปนี้

```
1
2
3
4
```

รูปที่ ข.2: คำสั่ง migrate สร้างตาราง

3. จากนั้นทำการโหลดข้อมูลลงในตารางดังต่อไปนี้

```
1
```

รูปที่ ข.3: คำสั่ง migrate สร้างตาราง

4. จากนั้นทำการสร้าง admin ดังต่อไปนี้

```
1
```

รูปที่ ข.4: คำสั่ง migrate สร้างตาราง

5. จากนั้นทำการ run โปรแกรมโดยใช้คำสั่งดังรูปที่ ข.4 website http://127.0.0.1:8000 บนเว็บ server http://lookinggreat.pythonanywhere.com/ ดังต่อไปนี้

```
1
```

รูปที่ ข.5: คำสั่ง migrate สร้างตาราง

ภาคผนวก ค

คู่มือการใช้งานระบบ

คู่มือการใช้งานทั้งหมดของระบบ สามารถแบ่งออกเป็น 3 ส่วน ดังนี้

1 ส่วนผู้ใช้งาน

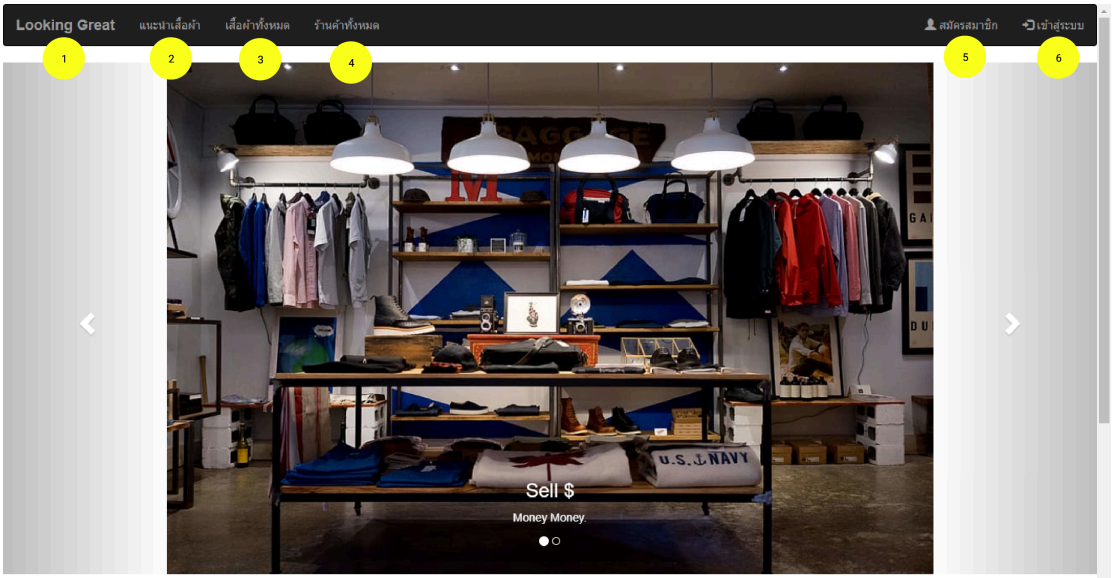
2 ส่วนร้านค้า

3 ส่วนผู้ดูแลระบบ

ค.1 ส่วนผู้ใช้งาน

1. ส่วนของหน้าหลัก

- เมื่อผู้ใช้เข้าเว็บระบบจะแสดงหน้าหลักดังต่อไปนี้



รูปที่ ค.1: หน้าจอหลักของระบบ

จากรูปที่ ค.13 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ปุ่มหน้าหลักของระบบ
- หมายเลขที่ 2 ปุ่มกดดูแนะนำเสื้อผ้า
- หมายเลขที่ 3 ปุ่มกดดูเสื้อผ้าทั้งหมดในระบบ
- หมายเลขที่ 4 ปุ่มกดดูร้านค้าทั้งหมดในระบบ
- หมายเลขที่ 5 ปุ่มสำหรับสมัครสมาชิก

- หมายเลขที่ 6 ปุ่มสำหรับเข้าสู่ระบบ

2. ส่วนของหน้าสมัครสมาชิก

- เมื่อผู้ใช้กดปุ่มสมัครสมาชิกระบบจะแสดงหน้าให้กรอกข้อมูลสมัครสมาชิกดังต่อไปนี้

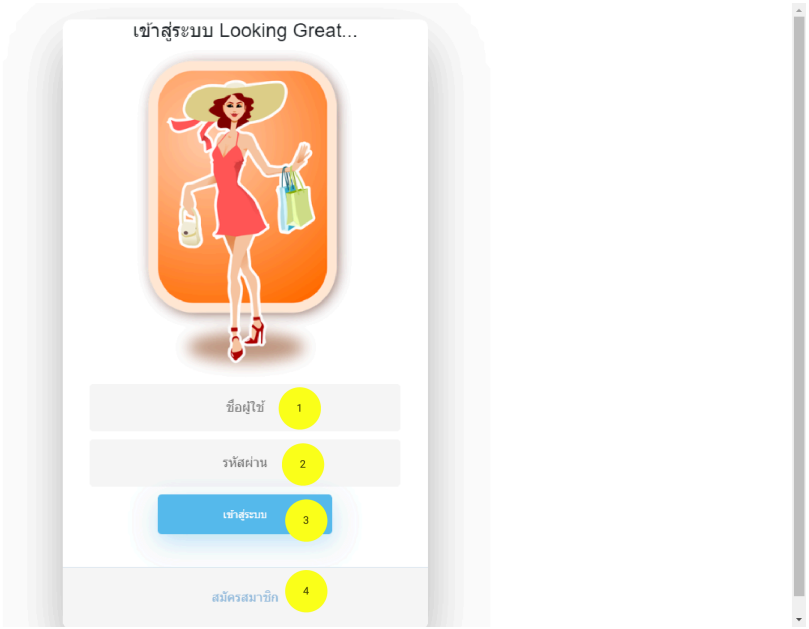
รูปที่ ค.2: หน้าจอสมัครสมาชิก

จากรูปที่ ค.14 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องกรอกชื่อผู้ใช้งาน
- หมายเลขที่ 2 ช่องกรอกรหัสผ่าน
- หมายเลขที่ 3 ช่องกรอก E-mail
- หมายเลขที่ 4 ช่องเลือกเพศ
- หมายเลขที่ 5 ช่องเลือกวัน/เดือน/ปีเกิด
- หมายเลขที่ 6 ช่องเลือกรูปร่าง
- หมายเลขที่ 7 ช่องเลือกสีผิว
- หมายเลขที่ 8 ช่องเลือกกรูปโปรไฟล์
- หมายเลขที่ 9 ปุ่มกดสมัครสมาชิก
- หมายเลขที่ 10 ปุ่มกดเข้าสู่ระบบหากทำการสมัครสมาชิกแล้ว
- หมายเลขที่ 11 ปุ่มกลับหน้าหลักของระบบ

3. ส่วนของหน้าเข้าสู่ระบบ

- เมื่อผู้ใช้สมัครสมาชิกสำเร็จแล้วกดปุ่มเข้าสู่ระบบ ระบบจะแสดงหน้าจอดังต่อไปนี้



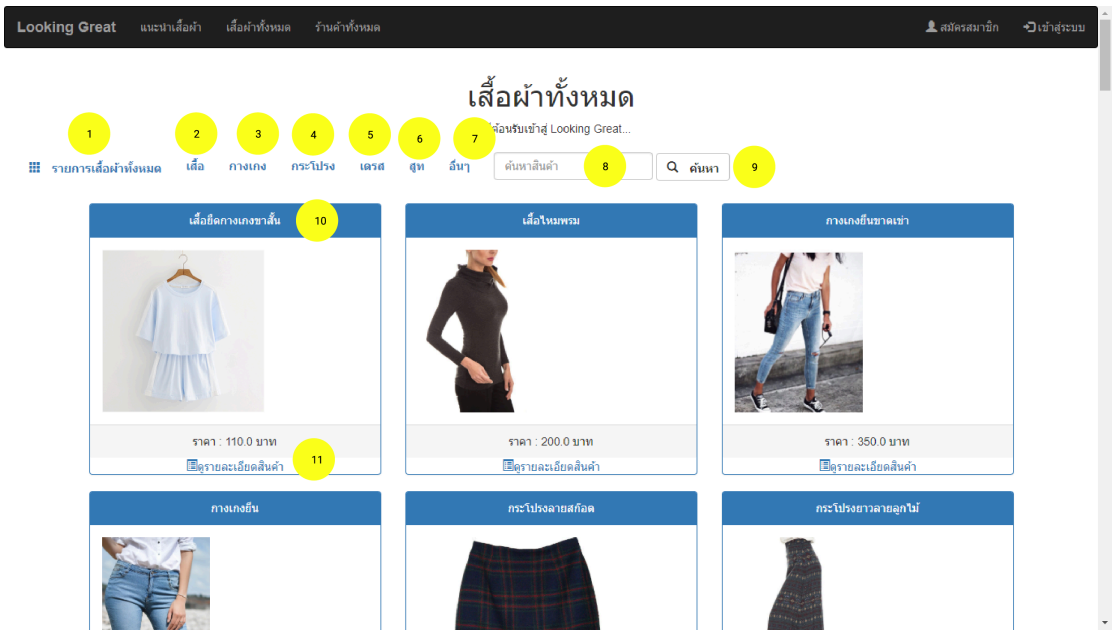
รูปที่ ค.3: หน้าจอเข้าสู่ระบบ

จากรูปที่ ค.15 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องกรอกชื่อผู้ใช้
- หมายเลขที่ 2 ช่องกรอกรหัสผ่าน
- หมายเลขที่ 3 ปุ่มกดเข้าสู่ระบบ
- หมายเลขที่ 4 ปุ่มสมัครสมาชิกสำหรับผู้ที่ยังไม่เป็นสมาชิก

4. ส่วนของหน้าเสื้อผ้าทั้งหมด

- เมื่อผู้ใช้กดปุ่มเสื้อผ้าทั้งหมดระบบจะแสดงหน้าจอดังต่อไปนี้



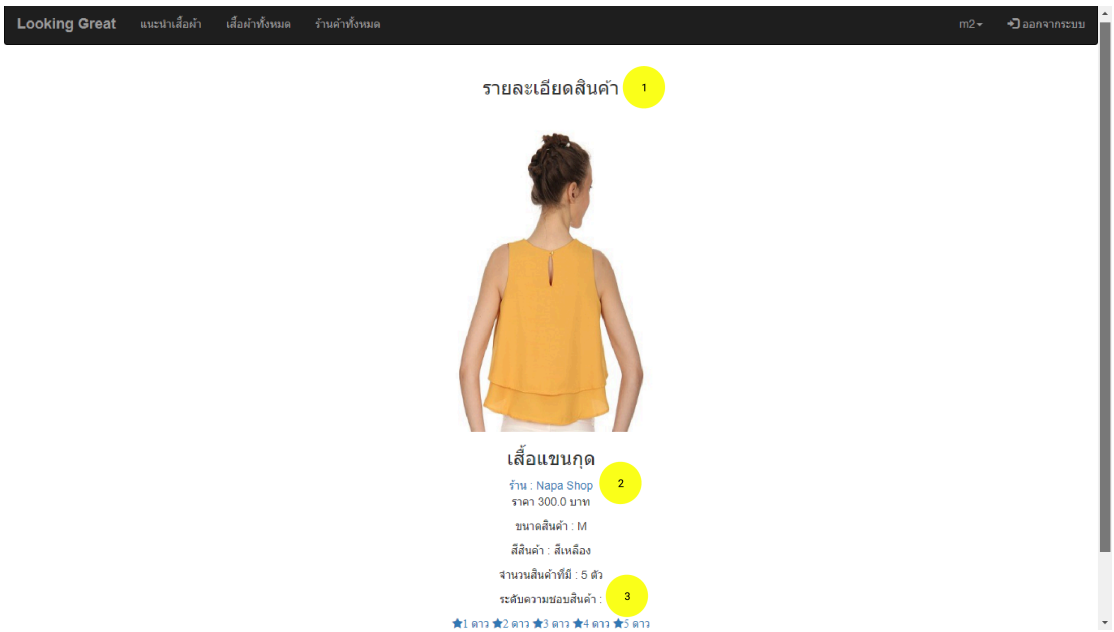
รูปที่ ค.4: หน้าจอเสื้อผ้าทั้งหมด

จากรูปที่ ค.4 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ปุ่มกดแสดงรายการเสื้อผ้าทั้งหมดรวมทั้ง 6 ประเภท
- หมายเลขที่ 2 ปุ่มกดดูเฉพาะเสื้อ
- หมายเลขที่ 3 ปุ่มกดดูเฉพาะกางเกง
- หมายเลขที่ 4 ปุ่มกดดูเฉพาะกระโปรง
- หมายเลขที่ 5 ปุ่มกดดูเฉพาะเดรส
- หมายเลขที่ 6 ปุ่มกดดูเฉพาะสูท
- หมายเลขที่ 7 ปุ่มกดดูรายการอื่นๆ
- หมายเลขที่ 8 ช่องกรอกชื่อเสื้อผ้าที่ต้องการค้นหา
- หมายเลขที่ 9 ปุ่มกดเพื่อค้นหา
- หมายเลขที่ 10 การ์ดแสดงรายการเสื้อที่แสดงจะประกอบไปด้วยชื่อและราคา
- หมายเลขที่ 11 ปุ่มกดเพื่อดูรายละเอียดสินค้า

5. ส่วนของหน้ารายละเอียดสินค้า

- เมื่อผู้ใช้กดปุ่มดูรายละเอียดสินค้าระบบจะแสดงหน้าจอดังต่อไปนี้



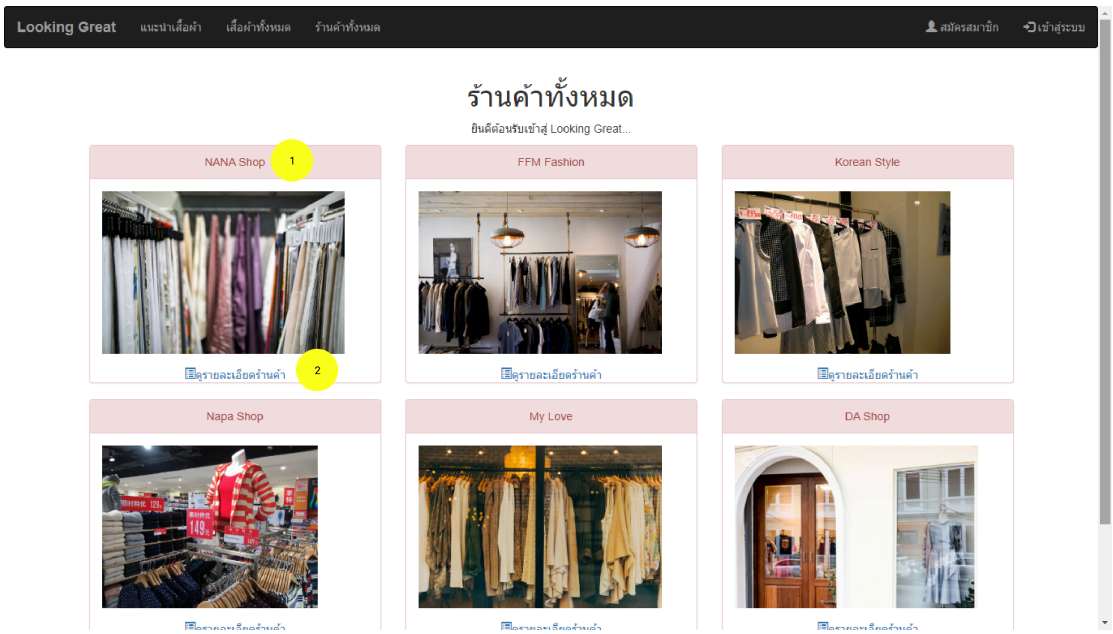
รูปที่ ค.5: หน้าจอรายละเอียดสินค้า

จากรูปที่ ค.5 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 แสดงรายละเอียดสินค้าทั้งหมด
- หมายเลขที่ 2 ปุ่มกดเพื่อดูร้านค้าของสินค้าชิ้นนี้
- หมายเลขที่ 3 ปุ่มเพิ่มเป็นสินค้าโปรดโดยสามารถกดเลือกระดับความชอบได้ 1-5 ซึ่งปุ่มกดเพิ่มเป็นสินค้าโปรดนี้จะแสดงต่อเมื่อเข้าสู่ระบบมาแล้วเท่านั้น

6. ส่วนของหน้าร้านค้าทั้งหมด

- เมื่อผู้ใช้กดปุ่มร้านค้าทั้งหมด ระบบจะแสดงหน้าจอดังต่อไปนี้



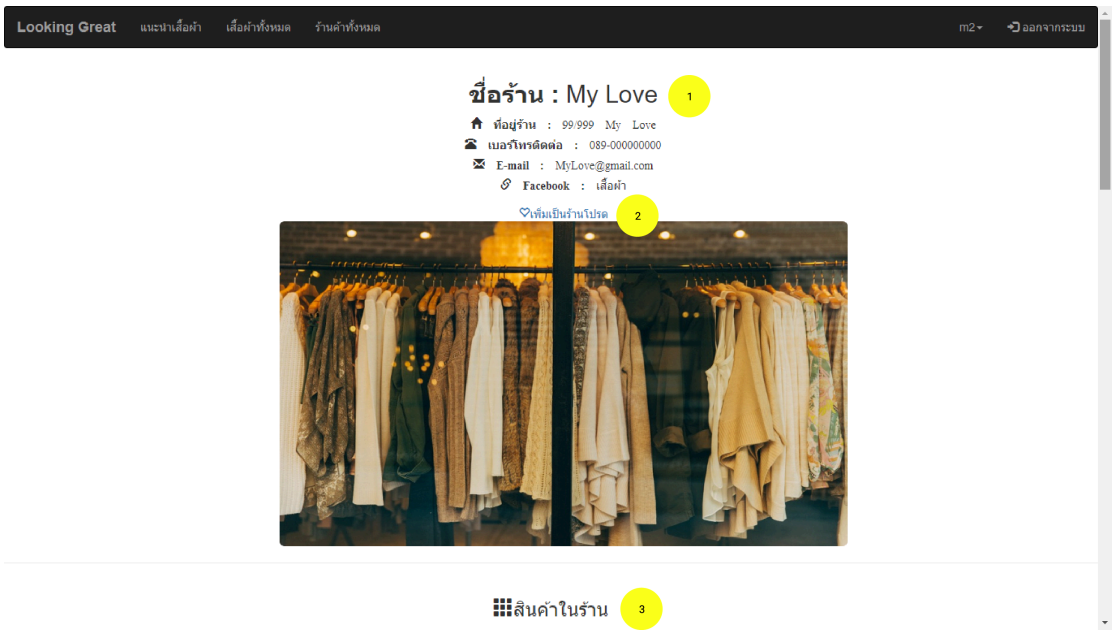
รูปที่ ค.6: หน้าจอร้านค้าทั้งหมด

จากรูปที่ ค.6 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 การ์ดแสดงรายการร้านค้าจะแสดงชื่อร้านค้า
- หมายเลขที่ 2 ปุ่มกดเพื่อดูรายละเอียดร้านค้า

7. ส่วนของหน้ารายละเอียดร้านค้า

- เมื่อผู้ใช้กดปุ่มดูรายละเอียดร้านค้า ระบบจะแสดงหน้าจอดังต่อไปนี้



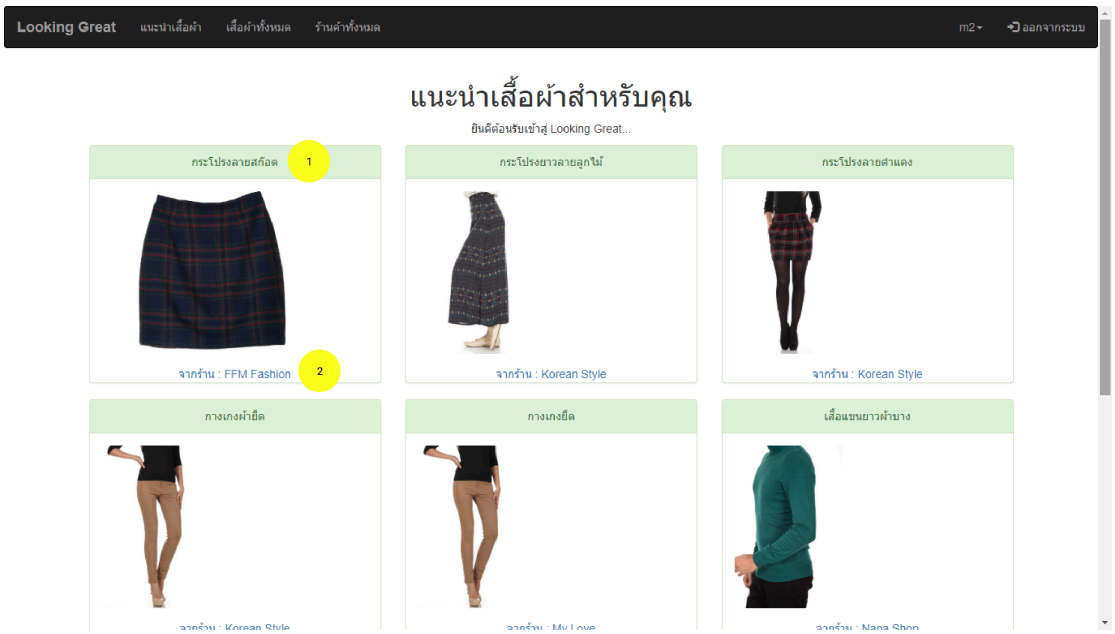
รูปที่ ค.7: หน้าจอรายละเอียดร้านค้า

จากรูปที่ ค.6 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 แสดงรายละเอียดร้านค้าทั้งหมด ซึ่งจะประกอบไปด้วย ชื่อร้าน ที่อยู่ร้าน เบอร์โทรติดต่อ E-mail และ Facebook
- หมายเลขที่ 2 ปุ่มกดเพิ่มเป็นร้านโปรด ซึ่งปุ่มกดเพิ่มเป็นร้านโปรดนี้จะแสดงต่อเมื่อเข้าสู่ระบบมาแล้วเท่านั้น
- หมายเลขที่ 3 จะแสดงรายการเสื้อผ้าทั้งหมดที่มีในร้าน

8. ส่วนของหน้าแนะนำเสื้อผ้า

- เมื่อผู้ใช้เข้าสู่ระบบเรียบร้อยแล้วทำการกดปุ่มแนะนำเสื้อผ้า ระบบจะแสดงหน้าจอดังต่อไปนี้



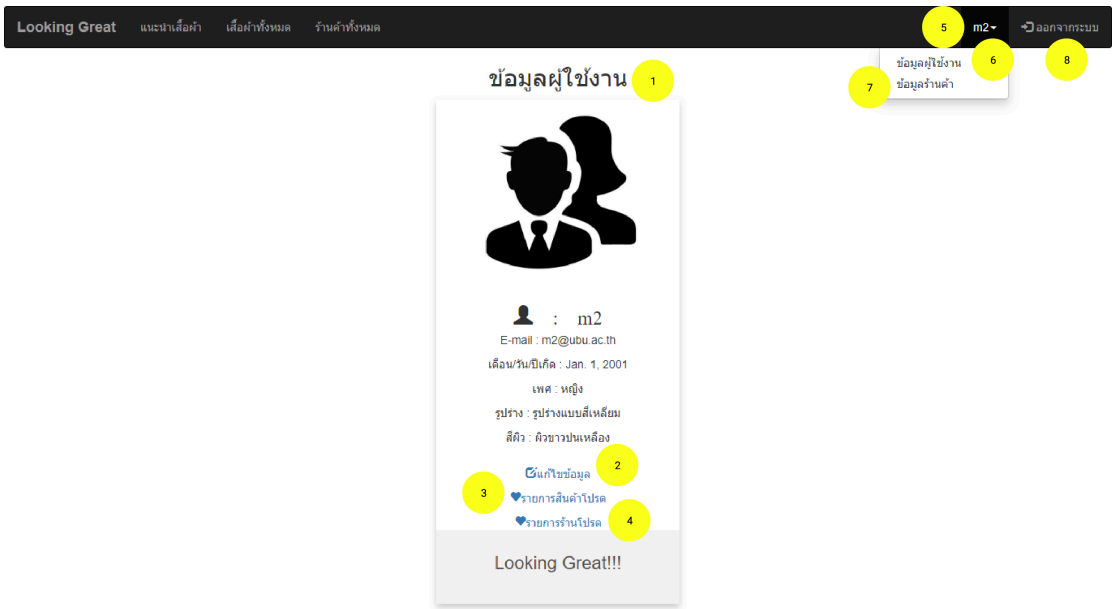
รูปที่ ค.8: หน้าจอแนะนำเสื้อผ้า

จากรูปที่ ค.8 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 แสดงรายละเอียดร้านค้าทั้งหมด ซึ่งจะประกอบไปด้วย ชื่อร้าน ที่อยู่ร้าน เบอร์โทรติดต่อ E-mail และ Facebookหมายเลขที่ 1 การ์ดแสดงรายการเสื้อผ้าแนะนำสำหรับผู้ใช้งาน
- หมายเลขที่ 2 ปุ่มกดดูรายละเอียดร้านค้าเพื่อดูสินค้าที่น่าสนใจเพิ่มเติม

9. ส่วนของหน้าข้อมูลผู้ใช้งาน

- เมื่อผู้ใช้ทำการเข้าสู่ระบบเรียบร้อยแล้วกดที่ปุ่มข้อมูลผู้มชัางาน ระบบจะแสดงหน้าจอดังต่อไปนี้



รูปที่ ค.9: หน้าจอข้อมูลผู้ใช้งาน

จากรูปที่ ค.9 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 การ์ดแสดงข้อมูลผู้ใช้งานโดยจะประกอบไปด้วย ชื่อผู้ใช้ E-mail วัน/เดือน/ปีเกิด เพศ รูปร่าง สีผิว และรูปโปร์ไฟล์
- หมายเลขที่ 2 ปุ่มกดแก้ไขข้อมูลผู้ใช้
- หมายเลขที่ 3 ปุ่มกดดูรายการสินค้าโปรด
- หมายเลขที่ 4 ปุ่มกดดูรายการ้านโปรด
- หมายเลขที่ 5 ปุ่มแสดงชื่อผู้ใช้จะประกอบไปด้วย ข้อมูลผู้ใช้งาน และข้อมูลร้านค้าในกรณีที่มีร้านค้าแล้ว แต่หากยังไม่มีร้านค้าจะแสดงรายการเป็นเพิ่มร้านค้า
- หมายเลขที่ 6 ปุ่มกดดูข้อมูลผู้ใช้งาน
- หมายเลขที่ 7 ปุ่มกดดูข้อมูลร้านค้า
- หมายเลขที่ 8 ปุ่มกดออกจากระบบ

10. ส่วนของหน้าแก้ไขข้อมูลผู้ใช้งาน

- เมื่อผู้ใช้กดปุ่มแก้ไขข้อมูล ระบบจะแสดงหน้าจอดังต่อไปนี้

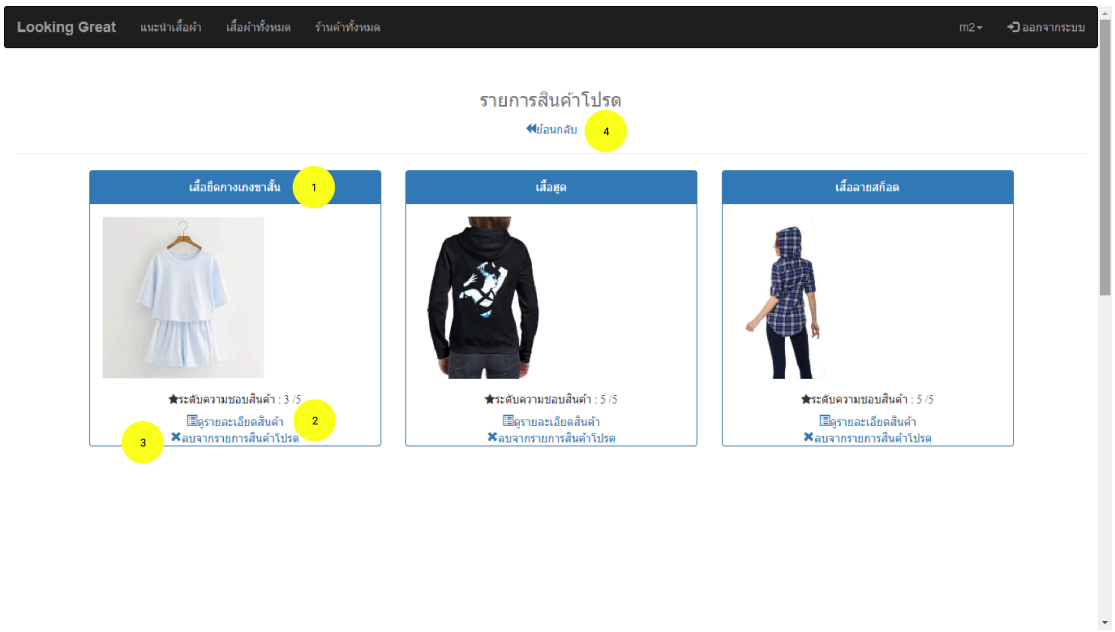
รูปที่ ค.10: หน้าจอแก้ไขข้อมูลผู้ใช้งาน

จากรูปที่ ค.10 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องกรอกชื่อผู้ใช้งาน ซึ่งจะไม่สามารถแก้ไขได้
- หมายเลขที่ 2 ช่องกรอกรหัสผ่าน ซึ่งจะไม่สามารถแก้ไขได้
- หมายเลขที่ 3 ช่องแก้ไขกรอก E-mail
- หมายเลขที่ 4 ช่องแก้ไขเลือกเพศ
- หมายเลขที่ 5 ช่องแก้ไขเลือกวัน/เดือน/ปีเกิด
- หมายเลขที่ 6 ช่องแก้ไขเลือกรูปร่าง
- หมายเลขที่ 7 ช่องแก้ไขเลือกสีผิว
- หมายเลขที่ 8 ช่องแก้ไขเลือกกรูปโปร์ไฟล์
- หมายเลขที่ 9 ปุ่มกดแก้ไขข้อมูลผู้ใช้
- หมายเลขที่ 10 ปุ่มกลับหน้าหลักของระบบ

11. ส่วนของหน้ารายการสินค้าโปรด

- เมื่อผู้ใช้กดปุ่มรายการสินค้าโปรด ระบบจะแสดงหน้าจอดังต่อไปนี้



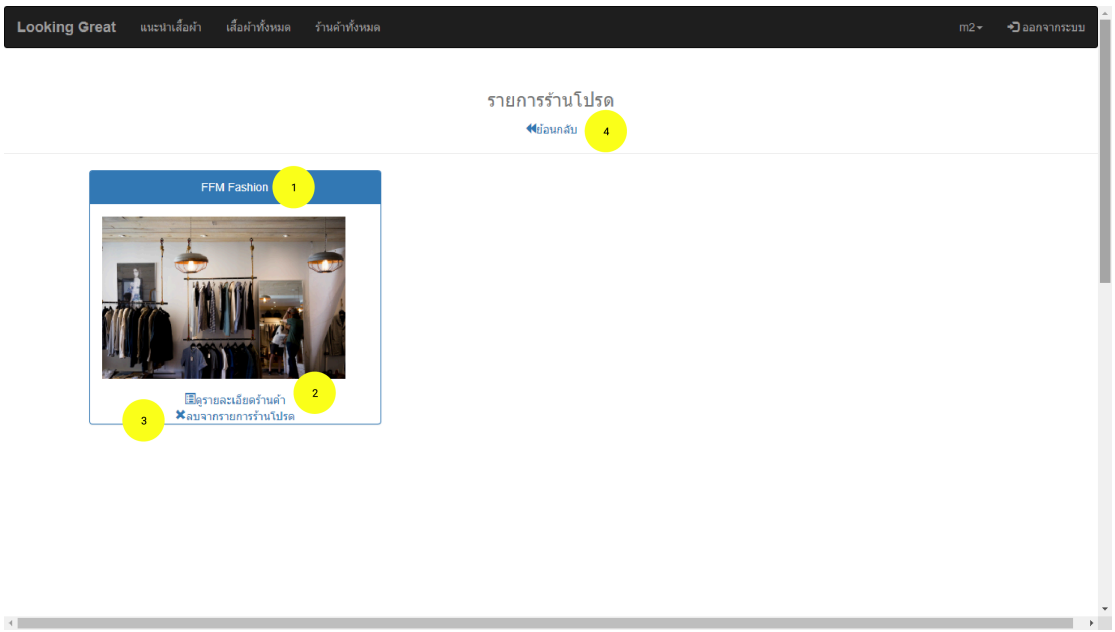
รูปที่ ค.11: หน้าจอรายการสินค้าโปรด

จากรูปที่ ค.11 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 การ์ดแสดงรายการสินค้าโปรดโดยจะประกอบไปด้วย ชื่อสินค้าและระดับความชอบสินค้าชิ้นนั้น
- หมายเลขที่ 2 ปุ่มกดเพื่อดูรายละเอียดสินค้า
- หมายเลขที่ 3 ปุ่มลบรายการสินค้าโปรด
- หมายเลขที่ 4 ปุ่มย้อนกลับไปหน้าข้อมูลผู้ใช้งาน

12. ส่วนของหน้ารายการร้านโปรด

- เมื่อผู้ใช้กดปุ่มรายการร้านโปรด ระบบจะแสดงหน้าจอดังต่อไปนี้



รูปที่ ค.12: หน้าจอรายการร้านโปรด

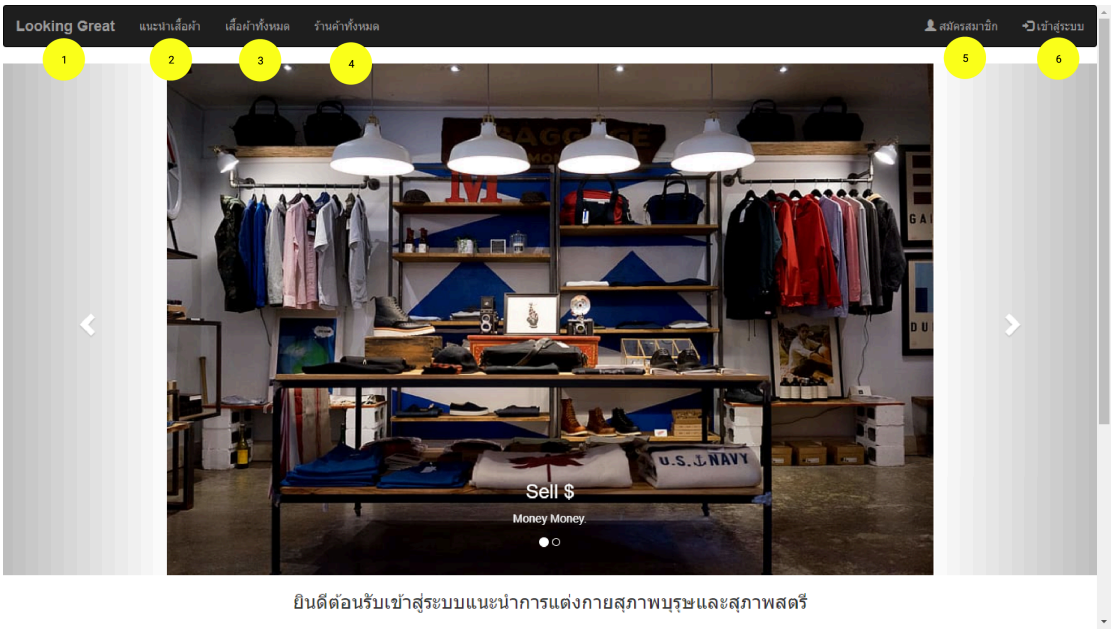
จากรูปที่ ค.12 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 การ์ดแสดงรายการร้านโปรดโดยจะประกอบไปด้วย ชื่อร้านค้า
- หมายเลขที่ 2 ปุ่มกดเพื่อดูรายละเอียดร้านค้า
- หมายเลขที่ 3 ปุ่มลบรายการร้านโปรด
- หมายเลขที่ 4 ปุ่มย้อนกลับไปหน้าข้อมูลผู้ใช้งาน

ค.2 ส่วนร้านค้า

1. ส่วนของหน้าหลัก

- เมื่อผู้ใช้เข้าเว็บระบบจะแสดงหน้าหลักดังต่อไปนี้



รูปที่ ค.13: หน้าจอหลักของระบบ

จากรูปที่ ค.13 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ปุ่มหน้าหลักของระบบ
- หมายเลขที่ 2 ปุ่มกดดูแนะนำเสื้อผ้า
- หมายเลขที่ 3 ปุ่มกดดูเสื้อผ้าทั้งหมดในระบบ
- หมายเลขที่ 4 ปุ่มกดดูร้านค้าทั้งหมดในระบบ
- หมายเลขที่ 5 ปุ่มสำหรับสมัครสมาชิก
- หมายเลขที่ 6 ปุ่มสำหรับเข้าสู่ระบบ

2. ส่วนของหน้าสมัครสมาชิก

- เมื่อผู้ใช้กดปุ่มสมัครสมาชิกระบบจะแสดงหน้าให้กรอกข้อมูลสมัครสมาชิกดังต่อไปนี้

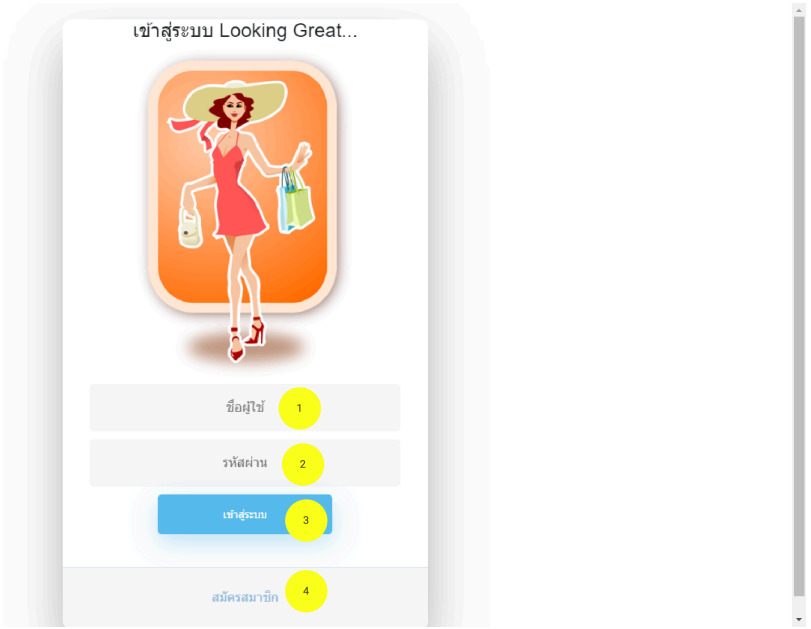
รูปที่ ค.14: หน้าจอสมัครสมาชิก

จากรูปที่ ค.14 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องกรอกชื่อผู้ใช้งาน
- หมายเลขที่ 2 ช่องกรอกรหัสผ่าน
- หมายเลขที่ 3 ช่องกรอกอีเมล์
- หมายเลขที่ 4 ช่องเลือกเพศ
- หมายเลขที่ 5 ช่องเลือกวัน/เดือน/ปีเกิด
- หมายเลขที่ 6 ช่องเลือกรูปร่าง
- หมายเลขที่ 7 ช่องเลือกสีผิว
- หมายเลขที่ 8 ช่องเลือกกรูปโปร์ไฟล์
- หมายเลขที่ 9 ปุ่มกดสมัครสมาชิก
- หมายเลขที่ 10 ปุ่มกดเข้าสู่ระบบหากทำการสมัครสมาชิกแล้ว
- หมายเลขที่ 11 ปุ่มกลับหน้าหลักของระบบ

3. ส่วนของหน้าเข้าสู่ระบบ

- เมื่อผู้ใช้สมัครสมาชิกสำเร็จแล้วกดปุ่มเข้าสู่ระบบ ระบบจะแสดงหน้าจอดังต่อไปนี้



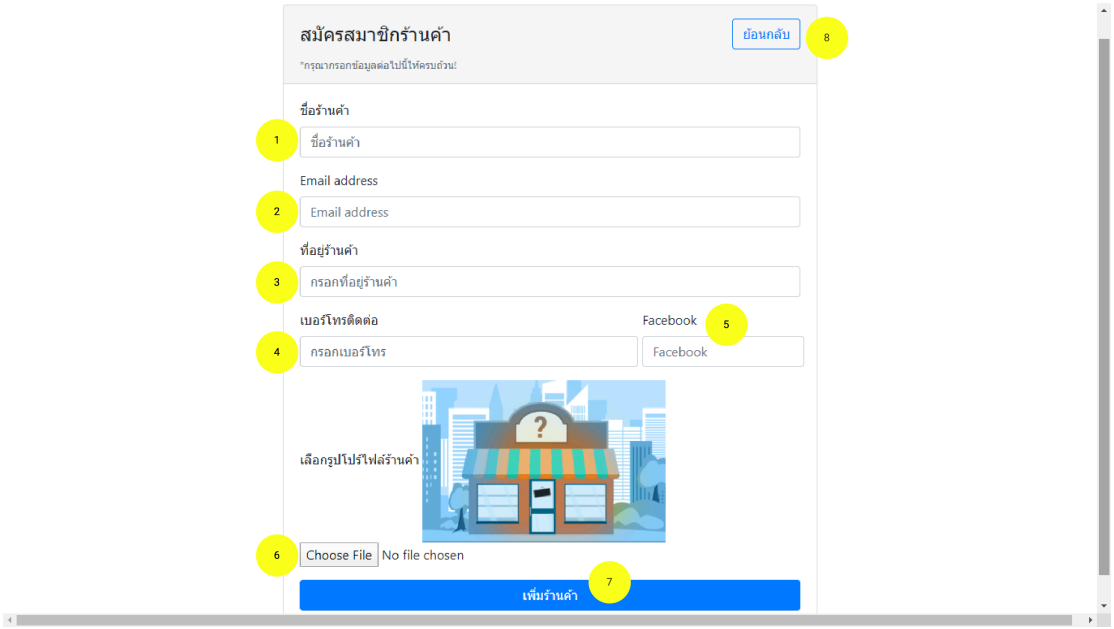
รูปที่ ค.15: หน้าจอเข้าสู่ระบบ

จากรูปที่ ค.15 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องกรอกชื่อผู้ใช้
- หมายเลขที่ 2 ช่องกรอกรหัสผ่าน
- หมายเลขที่ 3 ปุ่มกดเข้าสู่ระบบ
- หมายเลขที่ 4 ปุ่มสมัครสมาชิกสำหรับผู้ที่ยังไม่เป็นสมาชิก

4. ส่วนของหน้าเพิ่มร้านค้า

- เมื่อร้านค้าเข้าสู่ระบบเรียบร้อยแล้วกดปุ่มเพิ่มร้านค้าตรงชื่อผู้ใช้งาน ระบบจะแสดงหน้าจอดังต่อไปนี้



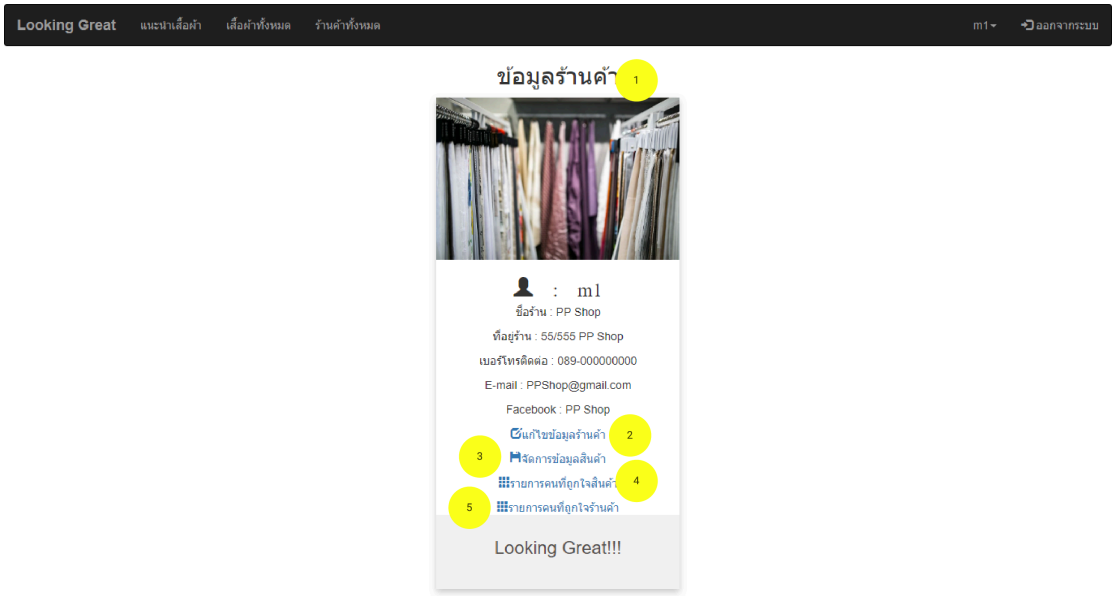
รูปที่ ค.16: หน้าจอเพิ่มร้านค้า

จากรูปที่ ค.16 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องกรอกชื่อร้านค้า
- หมายเลขที่ 2 ช่องกรอกที่อยู่ E-mail
- หมายเลขที่ 3 ช่องกรอกที่อยู่ร้านค้า
- หมายเลขที่ 4 ช่องกรอกเบอร์โทรติดต่อ
- หมายเลขที่ 5 ช่องกรอก Facebook
- หมายเลขที่ 6 เลือกรูปโปร์ไฟล์ร้านค้า
- หมายเลขที่ 7 ปุ่มกดเพิ่มร้านค้า ซึ่ง 1 ผู้ใช้จะสามารถเพิ่มร้านค้าได้แค่ 1 ร้าน
- หมายเลขที่ 8 ปุ่มกดกลับไปหน้าข้อมูลร้านค้า

5. ส่วนของหน้าข้อมูลร้านค้า

- เมื่อร้านค้าทำการเพิ่มข้อมูลร้านค้าสำเร็จแล้วทำการกดที่ชื่อผู้ใช้กดเลือกที่ข้อมูลร้านค้า ระบบจะแสดงหน้าจอดังต่อไปนี้



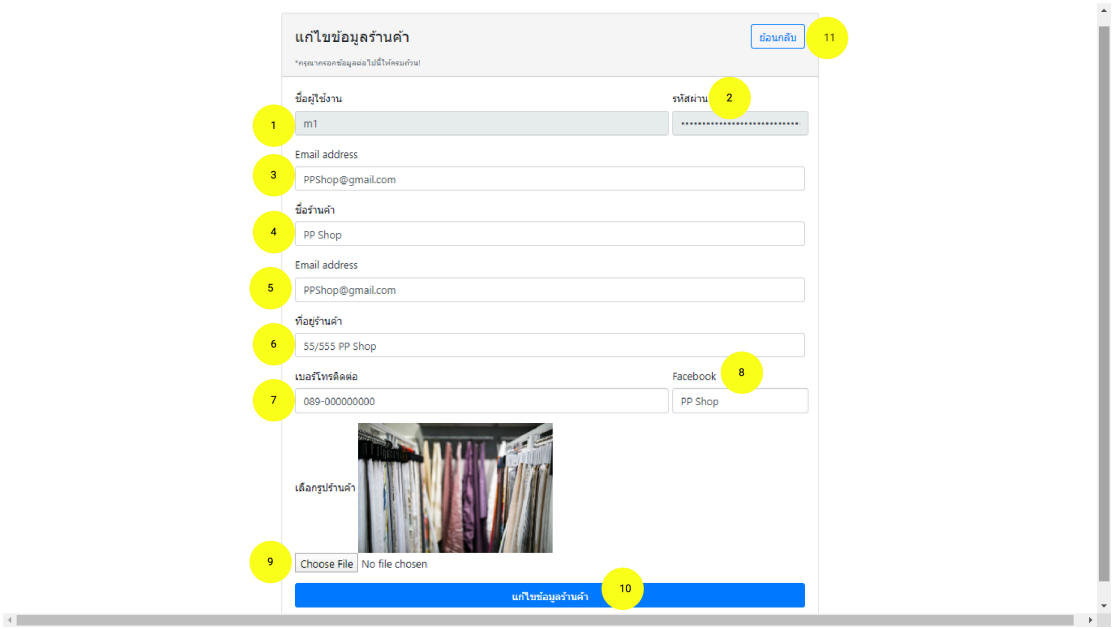
รูปที่ ค.17: หน้าจอข้อมูลร้านค้า

จากรูปที่ ค.17 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 การ์ดแสดงข้อมูลร้านค้าจะประกอบไปด้วย ชื่อผู้ใช้งาน ชื่อร้าน ที่อยู่ร้าน เบอร์โทรติดต่อ E-mail Facebook และรูปโปรไฟล์ร้านค้า
- หมายเลขที่ 2 ปุ่มกดแก้ไขข้อมูลร้านค้า
- หมายเลขที่ 3 ปุ่มกดจัดการข้อมูลสินค้าในร้าน
- หมายเลขที่ 4 ปุ่มกดดูรายการคนที่ถูกใจสินค้า
- หมายเลขที่ 5 ปุ่มกดดูรายการคนที่ถูกใจร้านค้า

6. ส่วนของหน้าแก้ไขข้อมูลร้านค้า

- เมื่อร้านค้ากดปุ่มแก้ไขข้อมูลร้านค้า ระบบจะแสดงหน้าจอดังต่อไปนี้



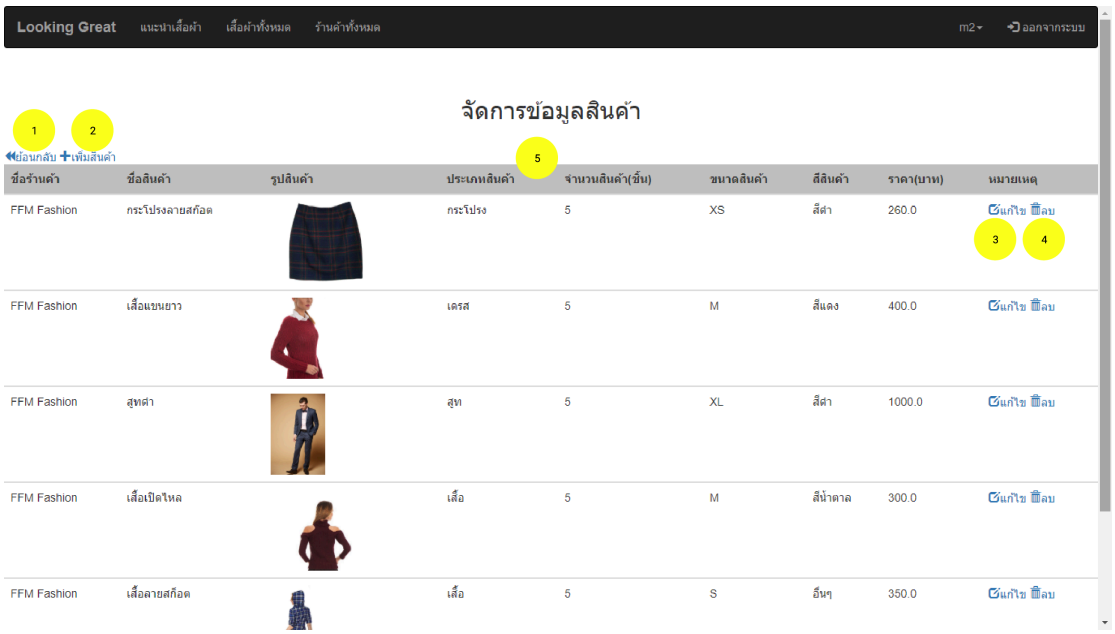
รูปที่ ค.18: หน้าจอแก้ไขข้อมูลร้านค้า

จากรูปที่ ค.18 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องกรอกชื่อผู้ใช้งาน ซึ่งจะไม่สามารถแก้ไขได้
- หมายเลขที่ 2 ช่องกรอกรหัสผ่าน ซึ่งจะไม่สามารถแก้ไขได้
- หมายเลขที่ 3 ช่องแก้ไขชื่อร้านค้า
- หมายเลขที่ 4 ช่องแก้ไขที่อยู่ E-mail
- หมายเลขที่ 5 ช่องแก้ไขที่อยู่ร้านค้า
- หมายเลขที่ 6 ช่องแก้ไขเบอร์โทรติดต่อ
- หมายเลขที่ 7 ช่องแก้ไข Facebook
- หมายเลขที่ 8 ช่องแก้ไขเลือกรูปร้านค้า
- หมายเลขที่ 9 ปุ่มกดบันทึกการแก้ไขข้อมูลร้านค้า
- หมายเลขที่ 10 ปุ่มย้อนกลับไปหน้าข้อมูลร้านค้า

7. ส่วนของหน้าจัดการข้อมูลสินค้า

- เมื่อร้านค้ากดปุ่มจัดการข้อมูลสินค้า ระบบจะแสดงหน้าจอดังต่อไปนี้



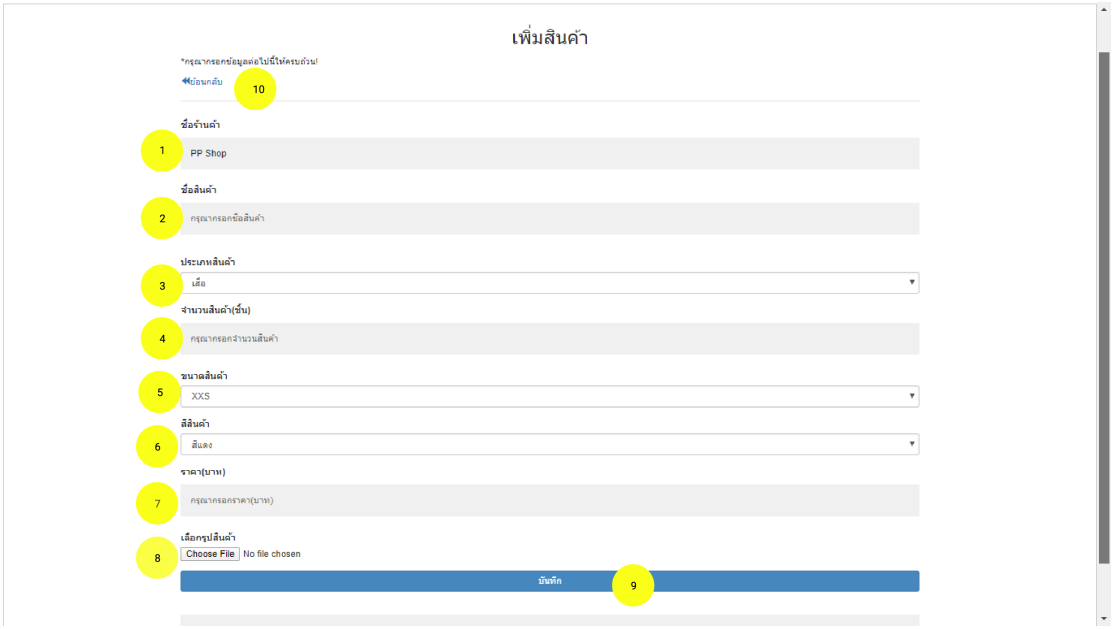
รูปที่ ค.19: หน้าจอจัดการข้อมูลสินค้า

จากรูปที่ ค.19 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ปุ่มย้อนกลับไปหน้าข้อมูลร้านค้า
- หมายเลขที่ 2 ปุ่มกดเพิ่มสินค้า
- หมายเลขที่ 3 ปุ่มกดแก้ไขสินค้า
- หมายเลขที่ 4 ปุ่มกดลบสินค้า
- หมายเลขที่ 5 ตารางแสดงรายการสินค้า ซึ่งจะประกอบไปด้วย ชื่อร้านค้า ชื่อสินค้า รูปสินค้า ประเภทสินค้า จำนวนสินค้า ขนาดสินค้า สีสินค้า และราคา

8. ส่วนของหน้าเพิ่มสินค้า

- เมื่อร้านค้ากดปุ่มเพิ่มสินค้า ระบบจะแสดงหน้าจอดังต่อไปนี้



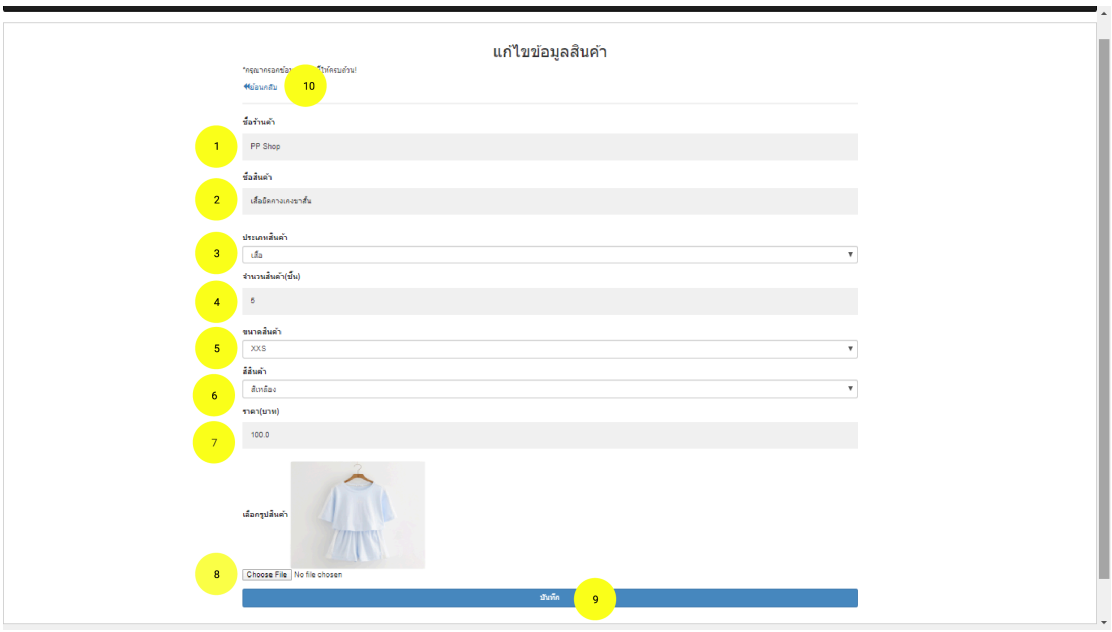
รูปที่ ค.20: หน้าจอเพิ่มสินค้า

จากรูปที่ ค.20 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องชื่อร้านค้า ซึ่งไม่สามารถแก้ไขได้
- หมายเลขที่ 2 ช่องกรอกชื่อสินค้า
- หมายเลขที่ 3 ช่องเลือกประเภทสินค้า
- หมายเลขที่ 4 ช่องกรอกจำนวนสินค้า
- หมายเลขที่ 5 ช่องเลือกขนาดสินค้า
- หมายเลขที่ 6 ช่องเลือกสีสินค้า
- หมายเลขที่ 7 ช่องกรอกราคาสินค้า
- หมายเลขที่ 8 ช่องเลือกรูปสินค้า
- หมายเลขที่ 9 ปุ่มบันทึกรายการสินค้า
- หมายเลขที่ 10 ปุ่มย้อนกลับไปหน้าจัดการข้อมูลสินค้า

9. ส่วนของหน้าแก้ไขข้อมูลสินค้า

- เมื่อร้านค้ากดปุ่มแก้ไขในตารางจัดการข้อมูลสินค้า ระบบจะแสดงหน้าจอดังต่อไปนี้



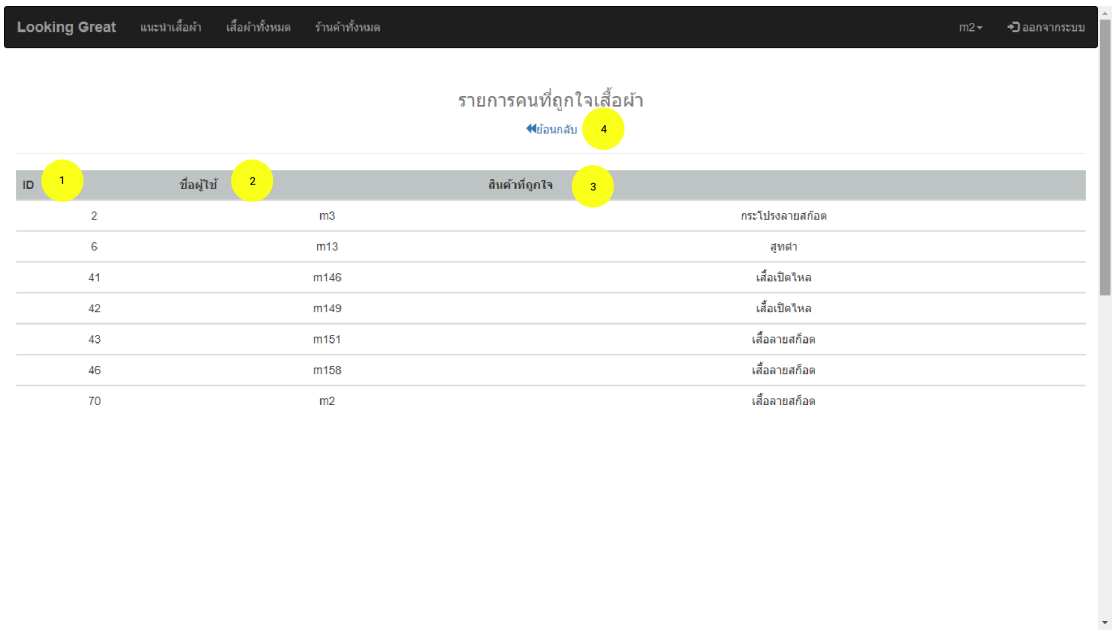
รูปที่ ค.21: หน้าจอแก้ไขข้อมูลสินค้า

จากรูปที่ ค.21 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องชื่อร้านค้า ซึ่งไม่สามารถแก้ไขได้
- หมายเลขที่ 2 ช่องกรอกชื่อสินค้า
- หมายเลขที่ 3 ช่องเลือกประเภทสินค้า
- หมายเลขที่ 4 ช่องกรอกจำนวนสินค้า
- หมายเลขที่ 5 ช่องเลือกขนาดสินค้า
- หมายเลขที่ 6 ช่องเลือกสีสินค้า
- หมายเลขที่ 7 ช่องกรอกราคาสินค้า
- หมายเลขที่ 8 ช่องเลือกรูปสินค้า
- หมายเลขที่ 9 ปุ่มบันทึกรายการสินค้า
- หมายเลขที่ 10 ปุ่มย้อนกลับไปหน้าจัดการข้อมูลสินค้า

10. ส่วนของหน้ารายการคนที่ถูกใจเสื้อผ้า

- เมื่อร้านค้ากดปุ่มดูรายการคนที่ถูกใจเสื้อผ้า ระบบจะแสดงหน้าจอดังต่อไปนี้



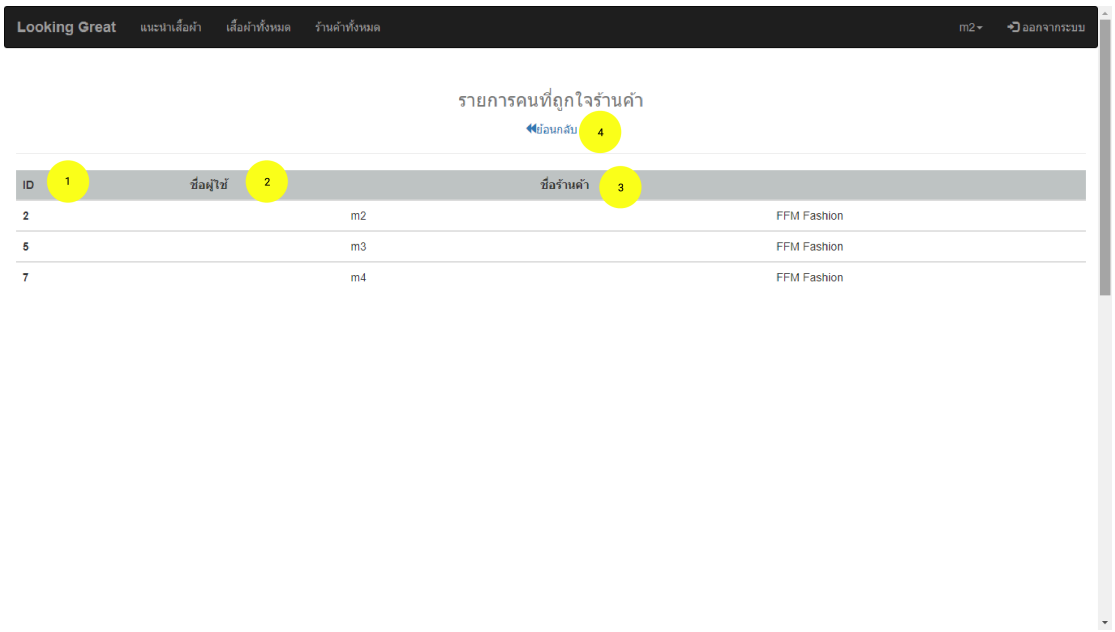
รูปที่ ค.22: หน้าจอรายการคนที่ถูกใจเสื้อผ้า

จากรูปที่ ค.22 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 รหัสผู้ใช้งานที่ถูกใจเสื้อผ้า
- หมายเลขที่ 2 ชื่อผู้ใช้งานที่ถูกใจเสื้อผ้า
- หมายเลขที่ 3 สินค้าที่ผู้ใช้งานถูกใจ
- หมายเลขที่ 4 ปุ่มย้อนกลับไปหน้าข้อมูลร้านค้า

11. ส่วนของหน้ารายการคนที่ถูกใจร้านค้า

- เมื่อร้านค้ากดปุ่มดูรายการคนที่ถูกใจร้านค้า ระบบจะแสดงหน้าจอดังต่อไปนี้



รูปที่ ค.23: หน้าจอรายการคนที่ถูกใจร้านค้า

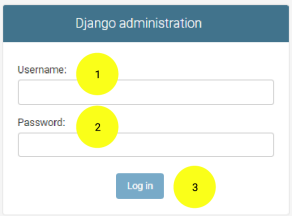
จากรูปที่ ค.23 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 รหัสผู้ใช้งานที่ถูกใจร้านค้า
- หมายเลขที่ 2 ชื่อผู้ใช้งานที่ถูกใจร้านค้า
- หมายเลขที่ 3 ชื่อร้านค้าที่ผู้ใช้งานถูกใจ
- หมายเลขที่ 4 ปุ่มย้อนกลับไปหน้าข้อมูลร้านค้า

ค.3 ส่วนผู้ดูแลระบบ

1. ส่วนของหน้าเข้าสู่ระบบสำหรับผู้ดูแลระบบในฐานข้อมูล

    - เมื่อผู้ดูแลทำการสมัครสมาชิกเสร็จเรียบร้อยแล้วทำการเข้าสู่ระบบ ระบบจะแสดงหน้าจอดังต่อไปนี้



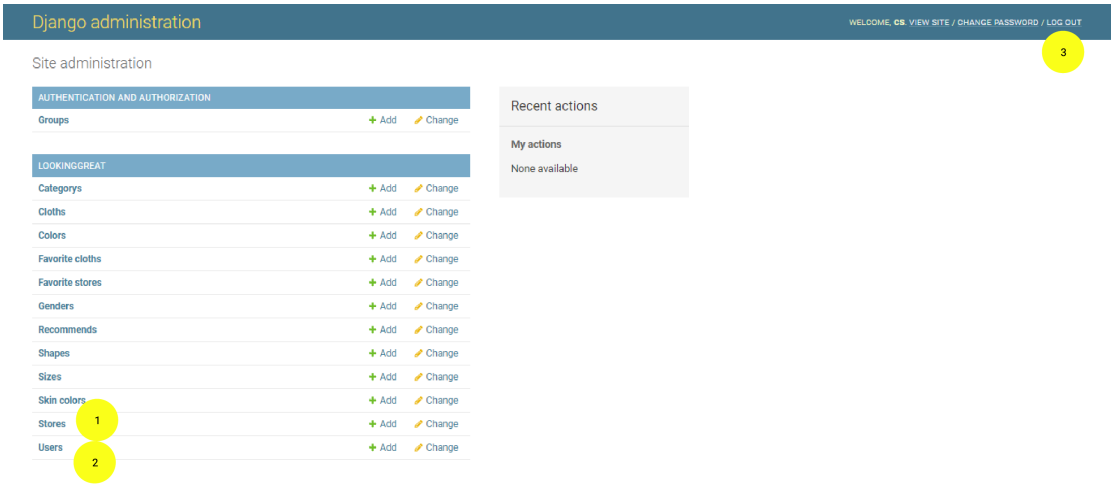
รูปที่ ค.24: หน้าจอเข้าสู่ระบบสำหรับผู้ดูแลระบบในฐานข้อมูล

จากรูปที่ ค.24 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลขที่ 1 ช่องกรอกชื่อผู้ใช้
- หมายเลขที่ 2 ช่องกรอกรหัสผ่าน
- หมายเลขที่ 3 ปุ่มกดเข้าสู่ระบบ

2. ส่วนของหน้าตารางฐานข้อมูลทั้งหมด

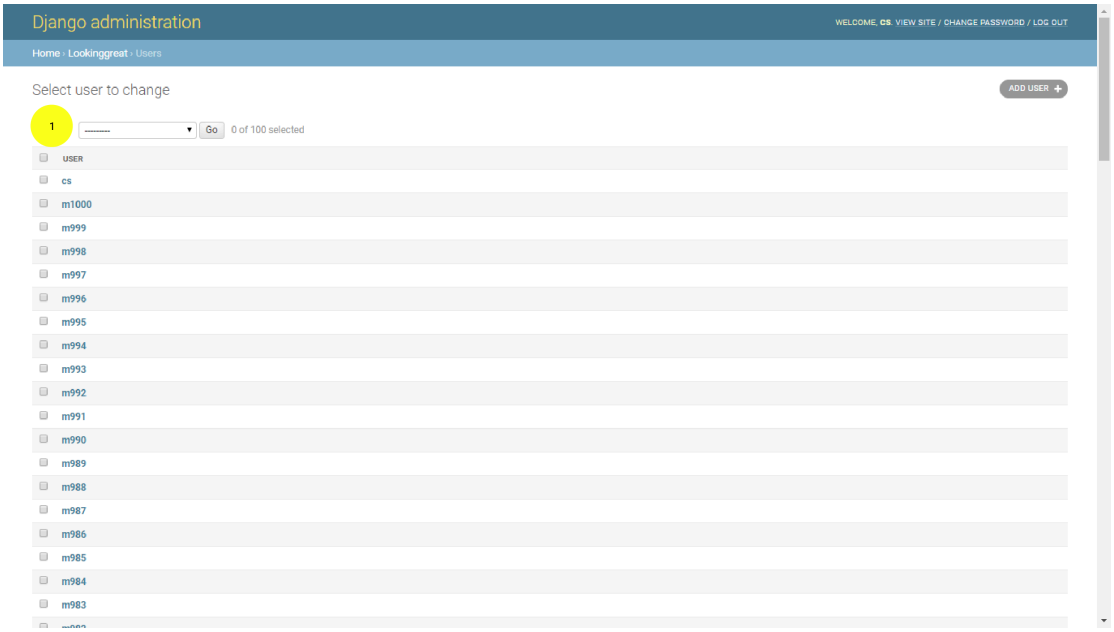
- เมื่อผู้ดูแลระบบเข้าสู่ระบบเรียบร้อยแล้ว โดยในที่นี้จะยกตัวอย่างข้อมูลเพียงแค่ในส่วนของ ผู้ใช้งานและร้านค้า ระบบจะแสดงหน้าจอดังต่อไปนี้



รูปที่ ค.25: หน้าจอหน้าตารางฐานข้อมูลทั้งหมด

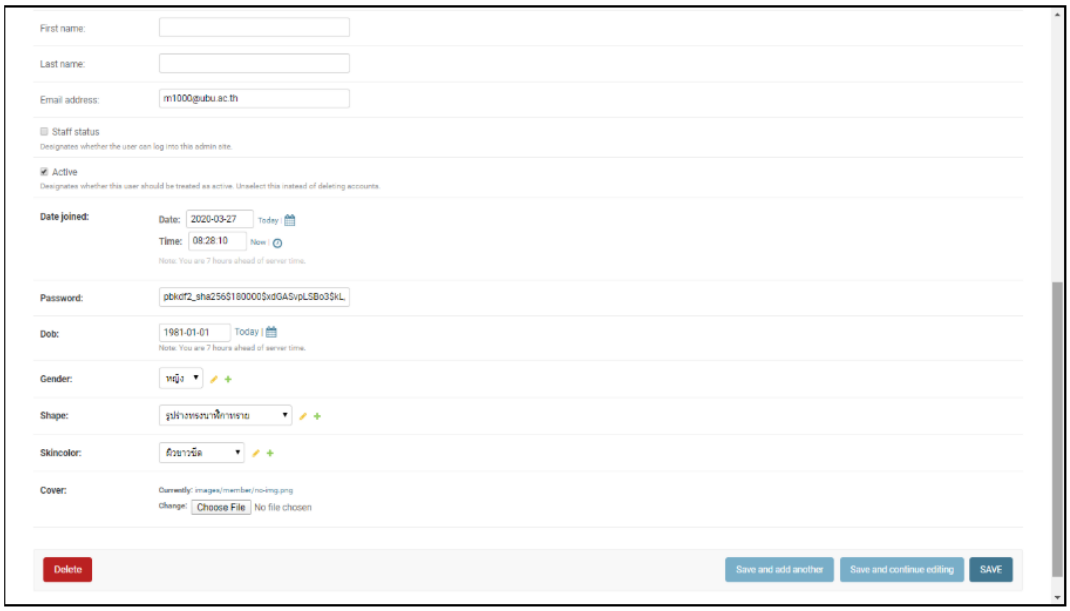
3. ส่วนของหน้าจอตารางข้อมูลผู้ใช้งานทั้งหมด

- เมื่อผู้ดูแลระบบกดปุ่ม Users ระบบจะแสดงหน้าจอดังต่อไปนี้



รูปที่ ค.26: หน้าจอตารางข้อมูลผู้ใช้งานทั้งหมด

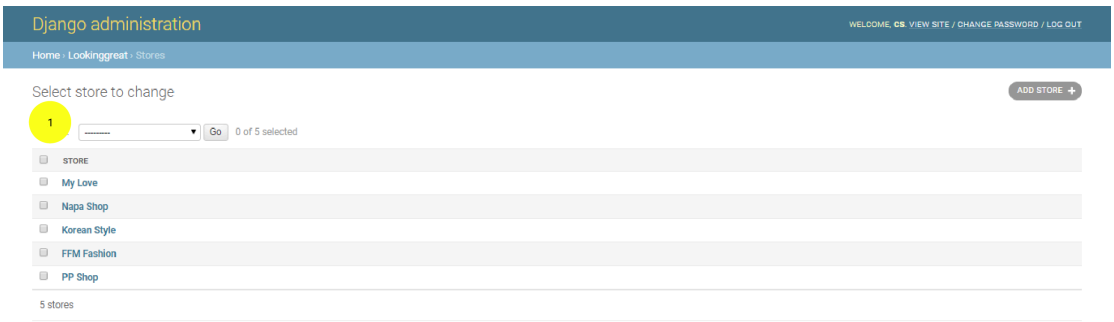
- เมื่อผู้ใช้กดที่ชื่อผู้ใช้งานระบบจะแสดงหน้าจอดังต่อไปนี้



รูปที่ ค.27: หน้าจอแสดงข้อมูลผู้ใช้งาน ซึ่งผู้ดูแลระบบจะสามารถเพิ่ม ลบ และแก้ไขผู้ใช้งานได้

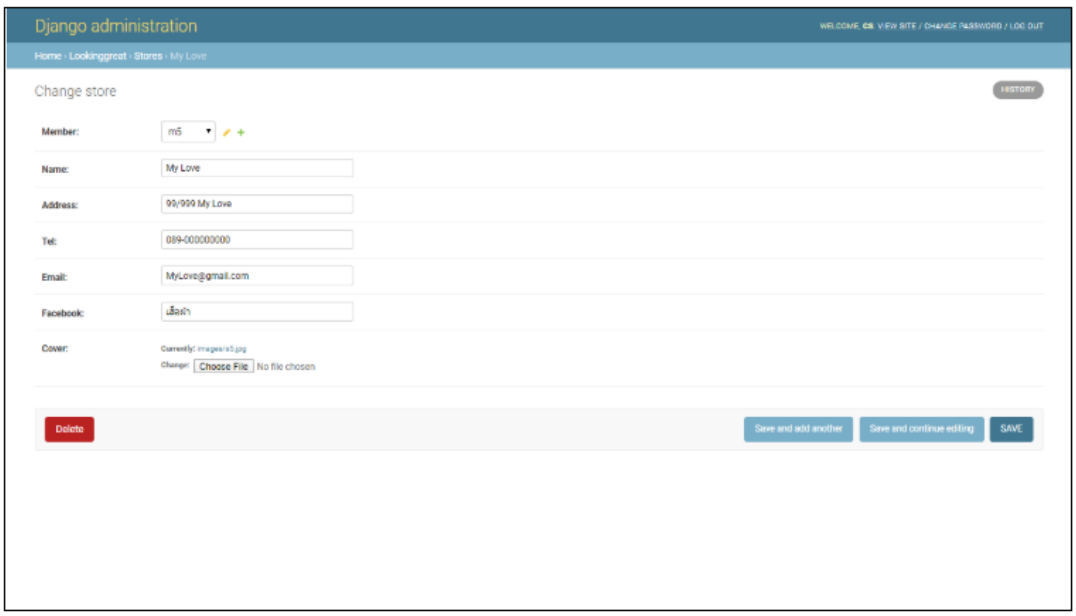
4. ส่วนของหน้าจอตารางข้อมูลร้านค้าทั้งหมด

- เมื่อผู้ดูแลระบบกดปุ่ม Stores ระบบจะแสดงหน้าจอดังต่อไปนี้



รูปที่ ค.28: หน้าจอแสดงข้อมูลร้านค้าทั้งหมด

- เมื่อผู้ใช้กดที่ชื่อร้านค้าระบบจะแสดงหน้าจอดังต่อไปนี้



รูปที่ ค.29: หน้าจอแสดงข้อมูลร้านค้า ซึ่งผู้ดูแลระบบจะสามารถเพิ่ม ลบ และแก้ไขร้านค้าได้

ประวัติผู้พัฒนา

ชื่อ-สกุล: นางสาวนฤมล โมคภา

รหัสประจำตัวนักศึกษา: 59110440204

วัดเกิด: 2 พฤศจิกายน 2540

ที่อยู่ที่สามารถติดต่อได้: 49 ม.1 ต.หัวช้าง อ.สุวรรณภูมิ จ.ร้อยเอ็ด 45130

เบอร์โทรศัพท์: (+66) 88 026 8694

อิเมลล์: narumon.mo.59@ubu.ac.th

ระดับมัธยมต้น: โรงเรียนบ้านห้างหว้า คุรุรัฐประชาสรรค์ จังหวัด ร้อยเอ็ด

ระดับมัธยมปลาย: โรงเรียนสุวรรณภูมิวิทยาลัย จังหวัด ร้อยเอ็ด

ระดับอุดมศึกษา: ภาควิชาคณิตศาสตร์ สถิติ และคอมพิวเตอร์ สาขาวิทยาการ คอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี