

โครงงานวิชา 01418211 Software Construction

ภาคต้น ปีการศึกษา 2564

อ่านรายละเอียดอย่างระมัดระวัง

ให้นิสิตสร้าง Desktop Application สำหรับการซื้อขายสินค้า (Marketplace E-commerce) ด้วย JavaFX (JavaSE 11 เท่านั้น) โดยต้องออกแบบและเขียนโปรแกรมที่ใช้หลักการโปรแกรมเชิงวัตถุ รายละเอียดและเกณฑ์การตรวจโปรแกรมมีดังนี้

รายละเอียดโครงงาน

1. โครงงานเป็นงานกลุ่ม กลุ่มละ 3-4 คน
2. โปรแกรมนี้เป็นโปรแกรมสำหรับการซื้อขายสินค้า (Marketplace E-commerce)
3. นิสิตมีอิสระในการกำหนดรูปแบบของสินค้า
4. โปรแกรมจะต้องมีหน้าจอหรือส่วนสำหรับผู้ขายให้นำสินค้าเข้ามาในระบบเพื่อวางขาย
5. โปรแกรมจะต้องมีหน้าจอหรือส่วนสำหรับผู้ซื้อให้เลือกซื้อสินค้า และชำระเงิน
6. ต้องใช้หลักการโปรแกรมเชิงวัตถุในการเขียนโปรแกรม
7. มี User Interface ที่ง่ายต่อการใช้งาน มีการแสดงข้อมูลที่ชัดเจน
ใช้ขนาดข้อความและขนาดภาพที่เหมาะสม สั่งเกตง่าย อ่านได้ง่าย เข้าใจง่าย

เกณฑ์การตรวจโครงงาน

นิสิตต้องทำโครงงานทั้งหมดด้วยตนเอง หากพบหรือสงสัย/ส่อว่า

(1) นิสิตคัดลอกงานมาจากอินเทอร์เน็ต ไม่ว่าส่วนใดส่วนหนึ่งหรือทั้งหมด นิสิตจะได้คะแนนโครงงานเป็น 0 และได้ระดับคะแนน I (ไม่สมบูรณ์)

(2) นิสิตคัดลอกงานจากเพื่อน ไม่ว่าส่วนใดส่วนหนึ่งหรือทั้งหมด
ทั้งผู้ลอกและผู้ให้ลอกจะได้คะแนนโครงงานเป็น 0 และได้ระดับคะแนน I (ไม่สมบูรณ์)

(3) หากนิสิตนำโครงชุดคำสั่งของเพื่อนมาปรับใช้ในงานของตน
คะแนนจะถูกหักตามจำนวนนิสิตที่ใช้โครงเดียวกัน หรืออาจถูกพิจารณาว่าคัดลอกงานจากเพื่อน

* นิสิตทุกคนในกลุ่มจะต้องเขียนโปรแกรมอย่างมีนัยสำคัญ โดยมีประวัติการ commit ในระบบ git

กำหนดการตรวจโครงงาน

นิสิตแต่ละกลุ่ม จะต้องติดต่ออาจารย์ผู้ตรวจ เพื่อแสดงความก้าวหน้าของการพัฒนาระบบอย่างน้อย 1 ครั้ง ในทุก 3 สัปดาห์ โดยนำเสนอต่อหน้าอาจารย์ผู้ตรวจ หรือผ่านระบบออนไลน์

การส่งโครงงานที่สมบูรณ์ ผู้ตรวจจะตรวจการทำงานของโปรแกรมผ่าน executable file เป็นหลัก โดยรันโปรแกรมผ่าน JRE (version JavaSE 11) และตรวจสอบการออกแบบโปรแกรมจากชุดคำสั่ง โดยทั้ง executable file และ source code จะต้องอยู่ใน github repository ที่กำหนด

กำหนดส่ง

ครั้งที่	วันที่	สิ่งที่ส่ง
1	26 ก.ค. - 13 ส.ค.	ความก้าวหน้าของระบบ (นำเสนอออนไลน์)
2	23 ส.ค. - 10 ก.ย.	ความก้าวหน้าของระบบ (นำเสนอออนไลน์)
3	13 ก.ย. - 1 ต.ค.	ความก้าวหน้าของระบบ (นำเสนอออนไลน์)
4	4 ต.ค. - 22 ต.ค.	โครงงานที่สมบูรณ์ (นำเสนอออนไลน์)

* โครงงานที่สมบูรณ์ของนิสิตกลุ่มใดที่ไม่ผ่านเกณฑ์ที่กำหนด หรือไม่ส่งความก้าวหน้าของระบบครั้งใดก็ตาม (หรือส่งเลยกำหนดทุกกรณี) จะได้เกรด F (ไม่ผ่าน)

เกณฑ์การตรวจโครงงานและให้คะแนน

รายละเอียดด้านล่าง คือ เกณฑ์การตรวจโครงงาน **โดยนิสิตจะต้องทำข้อที่มีสีแดงให้ได้** หากนิสิตไม่สามารถทำข้อที่มีสีแดงได้ นิสิตจะได้ระดับคะแนน I (ไม่สมบูรณ์) จนกว่านิสิตจะทำได้ **แต่ไม่เกินวันที่กำหนดส่งโครงงานที่สมบูรณ์** ส่วนข้ออื่น นิสิตจะได้คะแนนตามผลงานที่นิสิตทำมา หากทำได้นิสิตจะได้คะแนนในส่วนนั้น หากไม่ได้ทำหรือทำแต่ไม่สมบูรณ์ นิสิตจะไม่ได้คะแนนส่วนนั้นหรือได้คะแนนแค่บางส่วน **ข้อที่มีสีน้ำเงิน** เป็นข้อพึงระวัง นิสิตอาจไม่ได้รับการตรวจหรือถูกหักคะแนนหรือมีผลให้ได้เกรด F คะแนนที่มีคำว่า extra นำหน้า เป็นคะแนนพิเศษ

คะแนนเต็ม 200 คะแนน (คิดเป็น 25% ของคะแนนรายวิชา) คะแนนพิเศษ +30 คะแนน (คิดเป็น +5% ของคะแนนรายวิชา)

นิสิตแต่ละคนในกลุ่ม อาจได้คะแนนไม่เท่ากัน ขึ้นกับสัดส่วนของความร่วมมือในการพัฒนาโปรแกรม

- ✓ 1. (5 คะแนน) นิสิตจะต้อง commit โครงงาน ไปไว้ที่ github classroom ที่กำหนด โดยแต่ละครั้งก็นำเสนอความก้าวหน้าต้องมียอย่างน้อย 1 commit จากนิสิตแต่ละคนในกลุ่ม

2. นิสิตที่ไม่เสนอความก้าวหน้าหรือไม่มีความก้าวหน้าหรือไม่ส่งโครงการที่สมบูรณ์ภายในวันที่กำหนดครั้งใดก็ตาม นิสิตทุกคนจะได้เกรด F
- การนำเสนอครั้งใดก็ตามหรือการส่งโครงการที่สมบูรณ์ นิสิตคนใดที่ไม่มี commit หรือ commit นั้นไม่มีการเขียนโปรแกรมอย่างมีนัยสำคัญ นิสิตคนนั้นจะได้เกรด F

- ✓ 3. ใน repository จะต้องมีการโปรแกรม executable file (.jar หรือ .sh) และไฟล์ README.md ที่เขียนเรียบเรียงแสดงถึงความตั้งใจเขียน และมีเนื้อหาที่อธิบาย วิธีการติดตั้งหรือรันโปรแกรม การวางโครงสร้างไฟล์ (บอกว่า folder ใน repository เก็บอะไร) และสรุปสิ่งที่พัฒนาในแต่ละครั้งที่น่าเสนอความก้าวหน้าของระบบ หากผู้ตรวจไม่สามารถหา executable file ได้จาก repository จะไม่ได้รับการตรวจ
4. นิสิตต้องตั้งชื่อไฟล์ และ directory ให้ตรงกับบริบทของโครงการ หากไม่ตั้งชื่อตามบริบทของงาน จะถูกหักคะแนนอย่างน้อยจุดละ 10 คะแนน
- ✓ 5. (5 คะแนน) ต้องมีไฟล์ executable file ที่สามารถดับเบิลคลิกแล้วรันได้เลย หรือหากต้องอาศัย command ช่วยในการรัน ต้องระบุ command ในส่วนการติดตั้งโปรแกรมของไฟล์ README.md ด้วย
6. *** ต้องใช้หลักการโปรแกรมเชิงวัตถุพื้นฐาน
- ✓ • (5 คะแนน) ออกแบบคลาสให้สามารถสร้าง object ที่มีข้อมูลต่างกันผ่านการส่งค่าให้ parameter ใน constructor ได้ ดังนั้น นิสิตควรสร้าง object ที่เก็บข้อมูลต่างกัน โดยใช้คลาสเดียวกันได้
 - ✓ • (5 คะแนน) สร้าง attributes ได้ถูกต้อง ไม่ใช่ local variable มาเป็น attributes
7. ต้องเขียนโปรแกรมในรูปแบบของ Model View Controller (MVC) Architecture
- ✓ • (5 คะแนน) Model Class เก็บข้อมูลและ method ที่ทำงานในการแก้ไขข้อมูลของคลาส และ/หรือคำนวณ/ประมวลผลกับข้อมูลของคลาส มี Collection Class เพื่อจัดการ collection ของ model objects และจะต้องไม่มีส่วนของ Controller หรือ View ใน Model Class
 - ✓ • (5 คะแนน) Controller Class เป็นส่วนจัดการการรับข้อมูลจากผู้ใช้ และการส่งค่าไปแสดงผลใน View จะต้องไม่มีการคำนวณเกิดขึ้นใน Controller Class
 - ✓ • View (JavaFX) เป็นส่วนการแสดงผลข้อมูลของโปรแกรม นิสิตจะเขียนไฟล์ .fxml หรือจะสร้าง component สำหรับการแสดงผลจากใน Controller Class ก็ได้
8. ต้องใช้หลักการโปรแกรมเชิงวัตถุขั้นสูง รวมถึง การใช้ inheritance, interface, polymorphism ด้วย โดยนิสิตต้องออกแบบให้เหมาะสม
- เช่น ควรพิจารณาสถานการณ์ที่เหมาะสมในการใช้ inheritance ไม่ใช่สร้าง superclass แต่ไม่มีการใช้ประโยชน์จาก method ของ superclass หากนิสิตทำแบบนี้ ก็จะได้คะแนนในส่วนการใช้ inheritance เป็นต้น นิสิตควรออกแบบให้ดังนี้
- (5 คะแนน) มี superclass/subclass ที่เหมาะสม หมายความว่า ควรตรงกับคุณสมบัติ IS-A เช่น Car เป็น subclass ของ Vehicle นั้นเหมาะสม แต่ Book เป็น subclass ของ Library นั้นไม่เหมาะสม
 - ✓ • อาจมีหรือไม่มี interface ก็ได้ให้นิสิตพิจารณาตามความเหมาะสม

- (5 คะแนน) superclass และ subclass ต้องมีการทำงานบางอย่างที่เหมือนกัน และบางอย่างที่แตกต่างกัน โดย superclass ควรระบุ method ทั้งหมดที่ subclass ใช้ร่วมกัน (เพื่อให้เกิด polymorphism ได้) และ subclass ควร override บาง method ของ superclass ให้ตรงกับความต้องการของ subclass ดังนั้น นิสิตควรออกแบบให้ดีเพื่อให้ตรงกับคุณสมบัติดังกล่าว
- หาก subclass ต่างจาก superclass แค่ constructor แสดงว่ายังออกแบบไม่ดี
- หรือหาก superclass ไม่มี method ใดเลย แต่ subclass มี method ที่แตกต่างกันไปหมด แสดงว่ายังออกแบบไม่ดี
- ✓ • (5 คะแนน) เขียนโค้ดให้มี polymorphism ที่เหมาะสม โดยควรออกแบบคลาสที่ใช้ object ของ super type/sub type ให้สามารถใช้ object ของ sub type/super type เหล่านี้แทนได้

- ✓ 9. (5 คะแนน) ระบุ access modifier ที่เหมาะสม โค้ดอยู่ใน package ที่เหมาะสม และไม่ใช่ static หากไม่จำเป็น

- ✓ 10. (5 คะแนน) ตั้งชื่อ class, method, ตัวแปร ให้สื่อถึงบริบทหรือการใช้งาน ถูกต้องตามหลักการตั้งชื่อ

11. (5 คะแนน) ต้องสร้าง UML class diagram ที่แสดงถึงคลาสในโปรแกรม โดยเก็บเป็นไฟล์รูป เช่น .png, .jpg

- ✓ 12. (5 คะแนน) ต้องมีไฟล์ pdf เพื่ออธิบายรายละเอียดและการใช้งานโปรแกรมโดยมีหัวข้อดังต่อไปนี้
- "วิธีการใช้งาน<ชื่อ application ของนิสิต>"
เพื่ออธิบายวิธีการใช้งานโปรแกรมของนิสิต ซึ่งต้องมีการอธิบายอย่างเป็นขั้นตอน ใช้ภาษาที่เข้าใจง่าย และมีการแสดงรูปภาพประกอบ (นำเนื้อหาจาก README.md มาประกอบได้)
 - "ตัวอย่างข้อมูลผู้ใช้ระบบ"
เพื่อระบุตัวอย่างของ username และ password ในการเข้าใช้ระบบ ของผู้ใช้แต่ละ role
 - "รายละเอียดของไฟล์ CSV"
เพื่ออธิบายข้อมูลในไฟล์ .csv ว่า ไฟล์จะถูกสร้างไว้ที่ใด ข้อมูลที่เก็บในแต่ละ column/row ของไฟล์หมายถึงอะไร และ โปรแกรมของนิสิตอ่านข้อมูลจากไฟล์นี้เมื่อใด อ่านอย่างไร
 - "Extra Features"
หากนิสิตทำข้อ extra มาด้วย ให้ระบุด้วยว่าทำข้อใดบ้าง อยู่ส่วนใดของโปรแกรม หากผู้ตรวจอ่านวิธีการใช้งานที่นิสิตเขียนแล้วไม่เข้าใจได้โดยง่าย จะหักคะแนนอย่างน้อย 30 คะแนน

Features ของ Application มีดังนี้

- ✓ 13. (5 คะแนน) ต้องมีส่วนของ GUI ที่แสดงถึงข้อมูลต่อไปนี้ โดยอาจเข้าถึงข้อมูลนี้ผ่านเมนู หรือมีปุ่มกดที่หน้าแรก
- ข้อมูลนิสิตผู้จัดทำโปรแกรม ได้แก่
รูปที่แสดงหน้าตา (หน้ายิ้ม) ที่ชัดเจน ชื่อ นามสกุล ชื่อเล่น รหัสนิสิต
 - ข้อมูลคำสั่ง หรือคำแนะนำ ในการใช้โปรแกรมที่นิสิตสร้างขึ้นมา

- ✓ 14. ระบบของผู้ดูแลระบบ ระบบส่วนนี้ถูกจำกัดสิทธิ์สำหรับผู้ดูแลระบบเท่านั้น
ผู้ใช้ที่ไม่ใช่ผู้ดูแลระบบต้องไม่สามารถเข้าใช้งานส่วนนี้ได้
* Application ไม่ต้องมีระบบการสร้างบัญชีสำหรับผู้ดูแลระบบ ให้นิสิตรระบุ username และ password
ของผู้ดูแลระบบในไฟล์ pdf (ข้อ 11 หัวข้อ "ตัวอย่างข้อมูลผู้ใช้ระบบ")
- ✓ 14.1 (5 คะแนน) ผู้ดูแลระบบสามารถเปลี่ยนรหัสผ่านของตนเองได้ และรหัสผ่านใหม่ต้องใช้ได้
- ✓ 14.2 (5 คะแนน) มีหน้าแสดงรายชื่อของผู้ใช้ระบบ โดยต้องแสดงภาพผู้ใช้ username ชื่อของผู้ใช้ระบบ
ชื่อร้านค้า (กรณีผู้ใช้ระบบเปิดร้านค้าของตนเอง) และวันเวลาที่เข้าใช้ล่าสุดของผู้ใช้ระบบคนนั้น
เรียงลำดับตามวันเวลาที่ผู้ใช้ระบบเข้าใช้งานล่าสุดก่อน
- 14.3 (5 คะแนน) มีหน้าแสดงรายการของรายงานความไม่เหมาะสมของผู้ใช้ระบบ (เช่น เนื้อหาไม่เหมาะสม
ความคิดเห็นไม่เหมาะสม ข่าวดปลอม เนื้อหาล่อแหลม เนื้อหาอันตรายมีความรุนแรง ฯลฯ)
ผู้ดูแลระบบสามารถระงับสิทธิ์การเข้าใช้งานของผู้ใช้ระบบได้
ผู้ใช้ระบบที่ถูกระงับสิทธิ์จะไม่สามารถเข้าใช้งานโปรแกรมได้
โดยมีข้อความแจ้งเตือนการถูกระงับสิทธิ์
และผู้ดูแลระบบสามารถทราบถึงจำนวนครั้งในการพยายามเข้าใช้โปรแกรมในขณะที่ผู้ใช้ระบบถูกระงับ
สิทธิ์ และผู้ดูแลระบบสามารถคืนสิทธิ์การเข้าใช้งานของผู้ใช้ระบบได้
- 14.4 (extra 5 คะแนน) มีเมนูจัดการหมวดหมู่สินค้า
สามารถสร้างหมวดหมู่ของสินค้าที่แต่ละร้านค้าจะนำมาขายได้
โดยสามารถกำหนดให้หมวดหมู่ที่ต่างกัน มีคุณลักษณะของสินค้าที่ต่างกัน เช่น หมวดเสื้อผ้า
จะให้เพิ่มข้อมูลสี ขนาด แต่หมวดโทรศัพท์มือถือ จะให้เพิ่มข้อมูลรุ่น และความจุ
และจะต้องนำไปใช้ในข้อ 16.2 ได้ด้วย หากการจัดการหมวดหมู่สินค้าเป็นแบบ hard code
หรือไม่สามารถเพิ่มหมวดหมู่ใหม่ได้จากใน application จะไม่ได้คะแนนในส่วนนี้

15. การสร้างบัญชีของผู้ใช้ระบบ

- ✓ 15.1 (5 คะแนน) มีระบบลงทะเบียน (registration) สำหรับผู้ใช้ระบบ
โดยผู้ใช้ระบบที่ลงทะเบียนนี้จะเป็นผู้ซื้อเท่านั้น ข้อมูลที่ใช้ในการลงทะเบียน ได้แก่ ชื่อของผู้ใช้ระบบ
ชื่อสำหรับเข้าสู่ระบบ (username) รหัสผ่าน ยืนยันรหัสผ่าน โดยมีการตรวจสอบ username
จะต้องไม่ซ้ำกับผู้ใช้ระบบที่มีอยู่แล้ว
- ✓ 15.2 (5 คะแนน) ผู้ใช้ระบบสามารถเปลี่ยนรหัสผ่านของตนเองได้ และรหัสผ่านใหม่ต้องใช้ได้
- ✓ 15.3 (5 คะแนน) ผู้ใช้ระบบ upload รูปภาพ เพื่อใช้เป็นภาพของผู้ใช้ระบบ และสามารถเปลี่ยนรูปได้
หากผู้ใช้ระบบยังไม่กำหนดภาพ ระบบจะให้ใช้ภาพ default แทนไปก่อน

16. ระบบสำหรับผู้ขายสินค้า

- ✓ 16.1 (5 คะแนน) มีเมนูเปิดร้านค้า ให้ผู้ใช้ระบบสามารถสร้างร้านค้าของตนเอง
ซึ่งจะทำให้ผู้ใช้ระบบมีสถานะเป็นผู้ขายด้วย โดยการเปิดร้านค้าจะใช้ข้อมูลชื่อร้าน
- ✓ 16.2 (5 คะแนน) มีเมนูเพิ่มสินค้า เพื่อลงขายสินค้าโดยใช้ข้อมูลชื่อสินค้า ภาพสินค้า รายละเอียดสินค้า
ราคาสินค้า จำนวนสินค้าในคลัง หมวดหมู่สินค้า คุณลักษณะสินค้าเพิ่มเติมตามหมวดหมู่
ซึ่งตั้งค่าจากเมนูจัดการหมวดหมู่สินค้าข้อ 14.4
มีการตรวจสอบรูปแบบของข้อมูลก่อนจะบันทึกการลงขายสินค้า
- ✓ 16.3 (5 คะแนน) มีเมนูดูรายการสินค้าทั้งหมด ที่แสดงรายการสินค้าทั้งหมด แสดงภาพสินค้า ชื่อสินค้า
ราคาสินค้า และจำนวนสินค้าในคลัง

โดยมีสัญลักษณ์และสีที่ทำให้สังเกตได้ว่าสินค้าใดมีจำนวนน้อยในคลัง ซึ่งระบุจากเมนูตั้งค่าร้านค้า ข้อ 16.6

- ✓ 16.4 (5 คะแนน) ในหน้ารายการสินค้าทั้งหมด สามารถเลือกสินค้าเพื่อแก้ไขข้อมูลสินค้าได้ โดยสามารถแก้ไขภาพสินค้า ชื่อสินค้า ราคาสินค้า ได้ถูกต้อง มีการตรวจสอบรูปแบบของข้อมูลก่อนจะบันทึกการแก้ไข
- ✓ 16.5 (5 คะแนน) ในหน้ารายการสินค้าทั้งหมด สามารถเลือกสินค้าเพื่อเพิ่มจำนวนสินค้าได้ จำนวนสินค้าจะต้องเพิ่มขึ้นอย่างถูกต้อง
- ✓ 16.6 (5 คะแนน) มีเมนูตั้งค่าร้านค้า กำหนดจำนวนที่ถือว่าสินค้ามีจำนวนน้อยในคลัง เพื่อใช้ในการกำหนดสัญลักษณ์หรือสีในข้อ 16.3
- 16.7 (5 คะแนน) มีเมนูดูรายการสั่งซื้อ แสดงรายการสินค้าที่มีผู้ซื้อสั่งซื้อ แยกรายการสั่งซื้อใหม่ และรายการสั่งซื้อที่จัดส่งแล้ว โดยแสดงภาพสินค้า ชื่อสินค้า จำนวนสินค้า ยอดซื้อ และมีส่วนของการจัดส่งสินค้า ซึ่งจะสมมติว่าผู้ขายส่งสินค้าเรียบร้อยแล้ว โดยต้องระบุหมายเลขติดตาม (tracking number) ในรายการสั่งซื้อ จากนั้นต้องลดจำนวนสินค้าได้ถูกต้อง และรายการสั่งซื้อจะเปลี่ยนสถานะเป็นจัดส่งแล้ว
- 16.8 (extra 5 คะแนน) มีเมนูจัดการโปรโมชั่น เพื่อใช้เป็นส่วนลดในการซื้อสินค้าในร้านค้า โดยให้ระบุเป็นโค้ดส่วนลด ซึ่งโค้ดจะต้องไม่ซ้ำกับโปรโมชั่นอื่นในร้านค้าเดียวกัน และไม่ซ้ำกับโค้ดของโปรโมชั่นของร้านค้าอื่นด้วย ผู้ขายสามารถสร้างโปรโมชั่นได้หลายรูปแบบ เช่น โปรโมชั่นส่วนลด X เปอร์เซ็นต์เมื่อซื้อตั้งแต่ Y บาทขึ้นไป โปรโมชั่นส่วนลด X บาท เมื่อซื้อสินค้า Y ชิ้นขึ้นไป เป็นต้น (ค่าไป ออกแบบคลาสโปรโมชั่นโดยอาศัยคุณสมบัติ polymorphism)

17. ระบบสำหรับผู้ซื้อสินค้า

ผู้ใช้ระบบที่ลงทะเบียนตามข้อ 15.1 จะเป็นผู้ซื้อสินค้าโดยปริยาย ซึ่งจะเข้าดูหน้า marketplace และเลือกซื้อสินค้าได้

- ✓ 17.1 (5 คะแนน) มีหน้า marketplace ที่แสดงสินค้าทั้งหมดจากทุกร้าน เรียงลำดับโดยให้สินค้าที่วางขายล่าสุดขึ้นแสดงก่อน โดยแสดงภาพสินค้า ชื่อสินค้า และราคาสินค้า
- ✓ 17.2 (5 คะแนน) ในหน้า marketplace ที่แสดงสินค้าทั้งหมดจากทุกร้าน มีส่วนให้ผู้ซื้อเลือกเรียงลำดับสินค้าจากราคามากที่สุดขึ้นแสดงก่อน หรือเลือกเรียงลำดับสินค้าจากราคาน้อยที่สุดขึ้นแสดงก่อน
- 17.3 (5 คะแนน) ในหน้า marketplace ที่แสดงสินค้าทั้งหมดจากทุกร้าน มีส่วนให้ผู้ซื้อเลือกการแสดงผลเฉพาะสินค้าตามหมวดหมู่ที่สนใจ โดยยังคงเลือกเรียงลำดับการแสดงผลสินค้าจากราคาข้อ 17.2 ได้
- 17.4 (5 คะแนน) ในหน้า marketplace ที่แสดงสินค้าทั้งหมดจากทุกร้าน มีส่วนให้ผู้ซื้อเลือกการแสดงผลเฉพาะสินค้าที่มีช่วงราคาที่คุณกำหนดได้เอง โดยยังคงเลือกเรียงลำดับการแสดงผลสินค้าจากราคาข้อ 17.2 ได้
- ✓ 17.5 (5 คะแนน) เมื่อผู้ซื้อกดเลือกสินค้าจากหน้า marketplace แล้ว ให้แสดงหน้ารายละเอียดสินค้า โดยแสดงข้อมูลสินค้าที่ผู้ขายกรอกข้อมูลการวางขายสินค้าข้อ 16.2 และแสดงชื่อร้านค้าที่สามารถเชื่อมโยงไปยังหน้าร้านค้าได้

- ✓ 17.6 (5 คะแนน) ผู้ซื้อเลือกซื้อสินค้าจากหน้า marketplace แล้วจากข้อ 17.5 สามารถสร้างรายการสั่งซื้อได้เลย (ไม่ต้องทำระบบตะกร้าสินค้า) โดยให้ระบุจำนวน และมีการสุ่มราคาให้ก่อนยืนยันการสร้างรายการสั่งซื้อ ซึ่งจะมีสถานะเป็นรายการสั่งซื้อใหม่ หากจำนวนสินค้ามีไม่เพียงพอ จะต้องมียอดความแจ้งเตือนเพื่อให้ผู้ซื้อทราบอย่างเหมาะสม
- 17.7 (extra 5 คะแนน) กรณีที่ทำเมนูจัดการโปรโมชั่น ข้อ 16.8 จะต้องให้ผู้ซื้อกรอกโค้ดโปรโมชั่นเพื่อใช้เป็นส่วนลด และมีการสุ่มราคาให้ก่อนยืนยันการสร้างรายการสั่งซื้อ หากโปรโมชั่นที่ระบุไม่สอดคล้องกับการเงื่อนไขในการใช้โปรโมชั่นจะต้องมียอดความแจ้งเตือนที่เหมาะสม และเข้าใจได้ว่าทำไมจึงใช้โปรโมชั่นที่กรอกโค้ดนี้ไม่ได้
- 17.8 (5 คะแนน) ในหน้ารายละเอียดสินค้า ข้อ 17.5 มีส่วนที่ให้ผู้ซื้อแจ้งรายงานความไม่เหมาะสมของสินค้า ซึ่งจะไปแสดงผลในข้อ 14.3 โดยต้องระบุประเภทของความไม่เหมาะสม และข้อความเพิ่มเติม ซึ่งเจ้าของร้านอาจถูกระงับสิทธิ์การเข้าใช้ระบบจากรายงานนี้เมื่อผู้ดูแลระบบเห็นว่าไม่เหมาะสมจริง
- 17.9 (5 คะแนน) ในหน้ารายละเอียดสินค้า ข้อ 17.5 มีส่วนที่ให้ผู้ซื้อเขียนรีวิวสินค้า และให้คะแนนความนิยมเป็นคะแนนเต็มในช่วง 1 ถึง 5 คะแนน ดังนั้นหน้าแสดงรายละเอียดสินค้าจะต้องแสดงคะแนนความนิยมเฉลี่ย และจำนวนผู้ให้คะแนนด้วย
- 17.10 (5 คะแนน) ในหน้ารายละเอียดสินค้า ข้อ 17.5 มีส่วนที่แสดงความคิดเห็นของการรีวิวสินค้า และคะแนนความนิยมที่ได้จากข้อ 17.9 และมีส่วนให้ผู้ซื้อแจ้งรายงานความไม่เหมาะสมของรีวิวสินค้า ซึ่งจะไปแสดงผลในข้อ 14.3 โดยต้องระบุประเภทของความไม่เหมาะสมและข้อความเพิ่มเติม ซึ่งเจ้าของรีวิวอาจถูกระงับสิทธิ์การเข้าใช้ระบบจากรายงานนี้เมื่อผู้ดูแลระบบเห็นว่าไม่เหมาะสมจริง
18. ความสวยงามและประสบการณ์ของผู้ใช้งาน
- ✓ 18.1 (5 คะแนน) ใช้โทนสีและองค์ประกอบต่าง ๆ ของ GUI ที่ดูได้ชัดเจน น่าใช้ ไม่แสบตา ไม่ลวงตา โดยลองนึกถึงว่าหากเราทำโปรแกรมนี้ให้คนทั่วไปใช้จริง ควรแสดงองค์ประกอบเหล่านี้อย่างไร
- ✓ 18.2 โปรดตรวจสอบความถูกต้องของข้อความที่ปรากฏในส่วนต่าง ๆ ของโปรแกรม หรือเอกสารประกอบ หากพบว่ามีคำผิด หักค่าที่ผิดจุดละ 5 คะแนน
- ✓ 18.3 (10 คะแนน) Graphic User Interface (GUI) ควรมีรูปแบบที่เข้าใจง่าย มีการแสดงข้อมูลที่ชัดเจน มี Label กำกับข้อมูลที่ชัดเจน เพื่อให้เข้าใจว่าค่าที่แสดงนั้นคือค่าอะไร มีการแสดงองค์ประกอบต่าง ๆ ในขนาดที่เหมาะสม ไม่เล็กหรือไม่ใหญ่จนเกินไป ขนาดหน้าจอของโปรแกรมต้องมีความสูง ไม่เกิน 1024 pixel
- ✓ 18.4 (extra 5 คะแนน) GUI มี effect ที่น่าสนใจ เมื่อมี action ต่าง ๆ เช่น มีการ highlight button เมื่อคลิกปุ่ม มี effect การ transition เมื่อเปลี่ยน scene เป็นต้น
- 18.5 (extra 10 คะแนน) ผู้ใช้โปรแกรมสามารถปรับเปลี่ยน Theme ของ Application ได้ เช่น เปลี่ยนโทนสี เปลี่ยนขนาดอักษร หรือเปลี่ยนฟอนต์ของตัวอักษรได้ โดยจะต้องเปลี่ยนให้สอดคล้องกันทุก Scene
19. โปรแกรมต้องสามารถบันทึกสถานะของข้อมูลต่าง ๆ ในรูปแบบของไฟล์ csv (comma-separated values) และสามารถโหลดไฟล์ที่บันทึกไว้นั้นมาแสดงผลได้อย่างถูกต้อง
- ✓ 19.1 ต้องมีการบันทึกสถานะและข้อมูลที่จำเป็นในรูปแบบไฟล์ csv

✓ 19.2 (5 คะแนน) เมื่อเปิดโปรแกรมใหม่ หรือเมื่อโปรแกรมโหลดไฟล์ csv
จะต้องแสดงข้อมูลที่ได้นับที่กไว้อย่างถูกต้อง

✓ 20. ต้องมีข้อมูลเริ่มต้นในการทดสอบระบบ เสมือนว่าระบบถูกใช้งานมาประมาณ 1 เดือน เป็นอย่างน้อย
เช่น มีข้อมูลผู้ดูแลระบบ ข้อมูลบัญชีผู้ใช้งานอย่างน้อย 5 บัญชี มีร้านค้าอย่างน้อย 3 ร้านค้า
มีสินค้าอย่างน้อย 10 รายการ เป็นต้น
หากมีข้อมูลเริ่มต้นระบบน้อยกว่าที่กำหนด จะไม่ตรวจให้

ขั้นตอนการเริ่มต้นโครงการ

1. ตัวแทนสมาชิกในกลุ่ม 1 คน กรอกข้อมูลในฟอร์ม <https://forms.gle/iNZkz73LWMj1LsAG6> เพื่อระบุรายชื่อสมาชิกในกลุ่ม และชื่อกลุ่ม พร้อมรับทราบเงื่อนไขการได้เกรด F ภายในวันอาทิตย์ที่ 25 กรกฎาคม 2564 เวลา 17.00 น. นิสิตที่ไม่กรอกข้อมูลภายในเวลาที่กำหนด ให้ถอนการลงทะเบียน หรือได้เกรด F ไปโดยปริยาย
2. ตัวแทนสมาชิกในกลุ่ม 1 คน สร้าง repository เริ่มต้น ที่ <https://classroom.github.com/g/Yqn18E0a> โดยระบุชื่อกลุ่ม และสมาชิกที่เหลือเข้าไปที่เดียวกัน (<https://classroom.github.com/g/Yqn18E0a>) เลือกกลุ่มตามชื่อที่ตกลงกันไว้
3. clone repository ลงมาในเครื่อง แล้วเริ่มเขียนโปรแกรมได้

การส่งงานเพื่อนำเสนอความก้าวหน้า

1. สมาชิกทุกคนในทีม ต้องมี commit งานในส่วนของตนเองอย่างมีนัยสำคัญ ซึ่งหมายถึงการพัฒนาบาง feature ที่ทำได้อย่างสมบูรณ์ด้วยตนเอง และเข้าใจในสิ่งที่ตนเองเขียน
2. แยกส่วนของ executable files ออกจาก directory target และสร้าง directory ใหม่ เพื่อใช้ในการส่ง executable files
3. push ทุกอย่างที่จะส่งผ่านระบบ git เข้าไปที่ GitHub Classroom Repository ของกลุ่มตนเอง ก่อนเวลานำเสนอ

การติดต่ออาจารย์ผู้ตรวจเพื่อนำเสนอความก้าวหน้า

จะแจ้งให้ทราบชื่ออาจารย์ผู้ตรวจโครงการของนิสิตแต่ละกลุ่ม และวิธีการติดต่อภายหลัง