

โครงงาน

แอปพลิเคชันแสดงตารางเรียนและสถานที่สอนของอาจารย์  
สำหรับนักเรียน ม.602 โรงเรียนสวนกุหลาบวิทยาลัย

จัดทำโดย

นายพงษ์เทวิน นาคพงศ์พิมาน  
นายวศกร นพวรรณพร

ครูที่ปรึกษา

ครูปิยมาศ ศรีสมพันธ์  
ครูอัญชานา นิ่มอนุสสรณ์สกุล  
ครูเสาวลักษณ์ กังวานสกุลทอง

โครงงานนี้เป็นส่วนหนึ่งของรายวิชา ว30294 โครงงานวิทยาศาสตร์สร้างสรรค์ 1  
ภาคเรียนที่ 2 ปีการศึกษา 2561 และรายวิชา ว30295 โครงงานวิทยาศาสตร์สร้างสรรค์ 2  
ภาคเรียนที่ 1 ปีการศึกษา 2562 โรงเรียนสวนกุหลาบวิทยาลัย



โครงงาน

แอปพลิเคชันแสดงตารางเรียนและสถานที่สอนของอาจารย์  
สำหรับนักเรียน ม.602 โรงเรียนสวนกุหลาบวิทยาลัย

จัดทำโดย

นายพงษ์เทวิน นาคพงศ์พิมาน ม.602 เลขที่ 16  
นายวศกร นพวรรณพร ม.602 เลขที่ 31

ครูที่ปรึกษา

ครูปิยมาศ ศรีสมพันธ์  
ครูอัญชานา นิ่มอนุสสรณ์สกุล  
ครูเสาวลักษณ์ กังวานสกุลทอง

โครงงานนี้เป็นส่วนหนึ่งของรายวิชา ว30294 โครงงานวิทยาศาสตร์สร้างสรรค์ 1  
ภาคเรียนที่ 2 ปีการศึกษา 2561 และรายวิชา ว30295 โครงงานวิทยาศาสตร์สร้างสรรค์ 2  
ภาคเรียนที่ 1 ปีการศึกษา 2562 โรงเรียนสวนกุหลาบวิทยาลัย

**บทคัดย่อ**

โครงงานนี้มีวัตถุประสงค์เพื่อพัฒนาแอปพลิเคชัน “Teacher Finder” สำหรับการแสดงผลข้อมูลตารางเรียนและข้อมูลสถานที่สอน สำหรับนักเรียนห้อง ม.602 โรงเรียนสวนกุหลาบวิทยาลัย ที่มีความสามารถในการเพิ่ม แก้ไข ดัดแปลงข้อมูลต่าง ๆ และการส่งต่อข้อมูลระหว่างอุปกรณ์ ทำการศึกษาโดยทำการพัฒนาแอปพลิเคชันตามลักษณะ Waterfall Model ได้แอปพลิเคชันที่ทำงานได้บรรลุวัตถุประสงค์ และได้ทำการสอบถามความพึงพอใจ โดยแบ่งข้อคำถามเป็นความพึงพอใจในหลักการทำงานในแอปพลิเคชันจำนวน 10 ข้อ และความพึงพอใจในการออกแบบแอปพลิเคชันจำนวน 10 ข้อ กลุ่มตัวอย่างคือนักเรียนห้อง ม.602 โรงเรียนสวนกุหลาบวิทยาลัยจำนวน 29 คน ได้ผลความพึงพอใจในหลักการทำงานในแอปพลิเคชันโดยเฉลี่ย 79% และความพึงพอใจในการออกแบบแอปพลิเคชันโดยเฉลี่ย 82%

**กิตติกรรมประกาศ**

โครงงาน แอปพลิเคชันแสดงตารางเรียนและสถานที่สอนของอาจารย์สำหรับโรงเรียนสวนกุหลาบวิทยาลัย ได้รับการสนับสนุนเป็นอย่างดีจากคุณครูปิยมาศ ศรีสมพันธ์ คุณครูอัญชานา นิ่มอนุสสรณ์สกุล คุณครูเสาวลักษณ์ กังวาลสกุลทอง ครูที่ปรึกษา และคุณครูท่านอื่น ๆ ที่ได้ให้คำปรึกษาในการจัดทำโครงงาน และได้รับความอนุเคราะห์จากผู้ปกครองที่ได้ให้ข้อเสนอแนะต่างๆ ภายในโครงงาน รวมไปถึงเพื่อน ๆ นักเรียนชั้น ม.602 ทุกท่าน ในการทำแบบสอบถามความพึงพอใจ

คณะผู้จัดทำ ขอขอบพระคุณทุกท่านที่ได้กล่าวมาข้างต้นเป็นอย่างสูง

คณะผู้จัดทำ

**บทที่ 1  
บทนำ**

**1.1 ที่มาและความสำคัญ**

โรงเรียนสวนกุหลาบวิทยาลัย เป็นสถานศึกษาขนาดใหญ่พิเศษ มีห้องเรียน 79 ห้อง และมีนักเรียนประมาณ 3,200 คน การศึกษาในปัจจุบันของนักเรียนโรงเรียนสวนกุหลาบวิทยาลัย มีทั้งการเรียนปกติ กิจกรรมของโรงเรียน กิจกรรมชุมนุม และอื่น ๆ อีกมากมาย ทำให้การจัดสรรเวลาในแต่ละวันมีความจำกัดและมีความสำคัญ จากประสบการณ์ของผู้วิจัยพบว่า ปริมาณและเวลาของกิจกรรมและการเรียนอาจส่งผลให้นักเรียนส่งงานไม่ทันตามกำหนดเวลา จึงจำเป็นต้องมีการส่งงานนอกเหนือจากวันเวลาที่อาจารย์กำหนด ซึ่งมีความยุ่งยากในการตามหาสถานที่ที่อาจารย์อยู่ในขณะนั้น เนื่องจากอาจารย์อาจกำลังสอนอยู่ในห้องเรียนหนึ่ง เป็นต้น ปัจจุบันนักเรียนโรงเรียนสวนกุหลาบวิทยาลัยมีความนิยมและให้ความสำคัญกับเทคโนโลยี และมีการใช้งานโทรศัพท์เคลื่อนที่ในชีวิตประจำวันกันมากขึ้น และปัจจุบันก็มีการออกแบบแอปพลิเคชันเพื่อให้รองรับการใช้งานประเภทต่าง ๆ เช่น การแสดงตารางสอนภายในโทรศัพท์มือถือ เป็นต้น แต่ยังมีปัญหาที่สำคัญ คือ ข้อมูลต่าง ๆ ไม่สามารถแก้ไขได้ในแอปพลิเคชัน หรือถ้าสามารถแก้ไขได้ ก็ต้องใช้เวลานาน ไม่มีวิธีการส่งต่อข้อมูลระหว่างครูและนักเรียน คณะผู้วิจัยจึงได้พัฒนาแอปพลิเคชัน “Teacher Finder” เพื่อแก้ปัญหาเหล่านี้ และเป็นต้นแบบให้กับการพัฒนาแอปพลิเคชันที่เกี่ยวข้องกับการศึกษาในลักษณะต่าง ๆ เช่น การตามหาอาจารย์ และการแสดงผลข้อมูลที่เป็นระบบทั้งโรงเรียน เป็นต้น

**1.2 วัตถุประสงค์**

เพื่อพัฒนาแอปพลิเคชัน “Teacher Finder” ในการแสดงผลข้อมูลตารางเรียน และข้อมูลสถานที่สอน สำหรับนักเรียนห้อง ม.602 โรงเรียนสวนกุหลาบวิทยาลัย ที่มีความสามารถในการเพิ่ม แก้ไข ดัดแปลงข้อมูลต่าง ๆ และการส่งต่อข้อมูลระหว่างอุปกรณ์ได้

**1.3 สมมติฐาน**

นักเรียนห้อง ม.602 สามารถใช้แอปพลิเคชันที่พัฒนาขึ้นเพื่อทราบข้อมูลตารางเรียน และข้อมูลสถานที่สอนได้อย่างสะดวก รวดเร็ว ชัดเจน เข้าใจง่าย และสามารถเพิ่ม แก้ไข ดัดแปลงข้อมูล ส่งต่อข้อมูลระหว่างกันได้ตามวัตถุประสงค์

**1.4 ขอบเขตของการศึกษา**

1. แอปพลิเคชันที่ได้จากการพัฒนา สามารถใช้ได้ภายในระบบปฏิบัติการ Android ตั้งแต่รุ่น Android 5.0 (Lollipop) ขึ้นไป ทำการทดสอบโดยใช้โทรศัพท์มือถือจำนวน 4 เครื่อง ได้แก่
   * Oppo R7 Lite (Android 5.1, ก.ย. 2558)
   * Samsung Galaxy J7 Prime (Android 6.0.1, ต.ค. 2559)
   * Dtac Phone X3 (Android 7.0, เม.ย. 2560)
   * Vivo Y91i (Funtouch OS 4.5 มาจาก Android 8.1, ธ.ค. 2561)
2. ข้อมูลที่ใช้ในการทดสอบและวัดความพึงพอใจภายในแอปพลิเคชัน จะใช้ข้อมูลตารางเรียน­เฉพาะคาบเรียนในห้อง ม.602 และใช้ข้อมูลอาจารย์เฉพาะอาจารย์ที่มีรายชื่อสอนในระดับชั้น ม.6 ของโรงเรียนสวนกุหลาบวิทยาลัย ในภาคเรียนที่ 1 ปีการศึกษา 2562 โรงเรียนสวนกุหลาบวิทยาลัย โดยการกรอกข้อมูลจากฐานข้อมูลตารางเรียน และรายชื่อรายวิชาที่มีอยู่ในเว็บไซต์ของโรงเรียนสวนกุหลาบวิทยาลัย (http://www.sk.ac.th) สืบค้นข้อมูลครั้งสุดท้ายเมื่อวันที่ 31 พ.ค. 2562

**1.5 ตัวแปรที่ศึกษา**

ตัวแปรต้น วิธีการได้มาซึ่งข้อมูลตารางเรียน และข้อมูลสถานที่สอน (ใช้และไม่ใช้แอปพลิเคชัน)

ตัวแปรตาม ความพึงพอใจของนักเรียน ห้อง ม.602 โรงเรียนสวนกุหลาบวิทยาลัย

ตัวแปรควบคุม ข้อมูลตารางเรียน และข้อมูลสถานที่สอน

**1.6 นิยามเชิงปฏิบัติการ**

1. ตารางเรียน คือ ตารางข้อมูลอธิบายกำหนดการเรียนการสอน หรือกิจกรรมต่าง ๆ ที่จะเกิดขึ้นภายในโรงเรียน โดยมีการแบ่งเป็นวันที่ ตามวันที่มีเรียน และคาบเรียนต่าง ๆ โดยโรงเรียนสวนกุหลาบวิทยาลัยมีคาบเรียนจำนวน 10 คาบ ยาวคาบละ 50 นาที โดยคาบเรียนที่ 1 เริ่มเรียนเวลา 8:30 น. คาบเรียนที่ 2 จะเริ่มเรียนต่อหลังจากคาบเรียนที่ 1 เสร็จสิ้น ไปจนถึงคาบเรียนที่ 10 ที่จะสิ้นสุดในเวลา 16:50 น. แต่การเรียนของนักเรียนจะเสร็จสิ้นเมื่อคาบสุดท้ายเสร็จสิ้น
2. วันที่มีเรียน คือ วันที่โรงเรียนกำหนดให้มีการเรียนการสอน ได้แก่ วันจันทร์ วันอังคาร วันพุธ วันพฤหัสบดี และวันศุกร์ ของทุกสัปดาห์ ยกเว้นวันที่โรงเรียนมีประกาศให้หยุดเรียน หรือมีกิจกรรมสำคัญทั้งวัน หรือในขณะปิดภาคเรียน
3. คาบเรียน คือการแบ่งเวลาเรียนภายในวันที่มีเรียน ออกเป็นส่วนต่าง ๆ เพื่อให้การจัดสรรเวลาเรียนของนักเรียนมีความชัดเจน และเป็นระเบียบ ภายในคาบเรียนจะระบุ ชื่อวิชา รหัสวิชา ห้องที่ใช้ในการเรียน นักเรียนที่เรียนในห้องนั้น ๆ รายชื่อของอาจารย์ต่าง ๆ ที่สอนในคาบนั้น ๆ เป็นต้น ยกเว้นบางคาบ ที่เป็นคาบของโรงเรียน เช่น คาบรับประทานอาหารซึ่งจะเป็นคาบเรียนที่ 4 หรือคาบเรียนที่ 5 หรือคาบประชุมในคาบเรียนที่ 1 หรือคาบเรียนสุดท้าย การเรียกคาบเรียนจะใช้หมายเลขคาบในการเรียก โดยหมายเลขคาบ 1 หมายถึงคาบเรียนที่ 1
4. การร่นคาบเรียน คือ การปรับเวลาของคาบเรียนให้น้อยลงชั่วคราว ให้เหลือคาบละ 40 นาที เป็นส่วนหนึ่งของคำสั่งของโรงเรียนซึ่งสามารถเรียกใช้ได้ในวันที่โรงเรียนมีกิจกรรมพิเศษ โดยมีสาเหตุให้นักเรียนได้เรียนในเวลาที่เหลือจากการทำกิจกรรมพิเศษนั้น ๆ โดยคาบเรียนที่ 1 จะเริ่มในเวลา 8:30 น. เช่นเดิมแต่คาบเรียนที่ 10 จะสิ้นสุดที่เวลา 15:10 น.

**บทที่ 2  
เอกสารและงานวิจัยที่เกี่ยวข้อง**

**2.1 แบบจำลองน้ำตก (Waterfall Model)**

แบบจำลองน้ำตก คือแบบจำลองการพัฒนาซอฟต์แวร์ (Software Development Lifecycle Model) แบบแรกๆ ที่ได้นำไปใช้กันอย่างแพร่หลาย โดยแบบจำลองจะสรุปขั้นตอนการทำงานในการพัฒนาซอฟต์แวร์ที่เหมาะสม ซึ่งมีทั้งหมด 6 ขั้นตอน



ภาพที่ 2.1 แบบจำลองน้ำตก  
ที่มา: TutorialsPoint (2019)

1. การกำหนดและวิเคราะห์ความต้องการ (Requirement Analysis) ให้ตรงกับที่ลูกค้าต้องการ

2. การออกแบบระบบ (System Design) หรือการวางโครงร่างของซอฟต์แวร์ก่อนการพัฒนาจริง

3. การดำเนินงานพัฒนา (Implementation) คือการพัฒนาซอฟต์แวร์ตามที่ได้ออกแบบไว้ โดยแบ่งซอฟต์แวร์เป็นส่วนต่าง ๆ ตามหน้าที่ และการใช้งานของมัน เรียกว่าหน่วย (Units)

4. การทดสอบระบบ และทดลองใช้ (Testing) คือการนำหน่วยต่าง ๆ มาเชื่อมโยงกันให้เป็นระบบที่สมบูรณ์ เพื่อให้ได้ผลิตภัณฑ์ที่พร้อมใช้งาน และทำการทดสอบหาปัญหาภายในซอฟต์แวร์ก่อนการแจกจ่าย

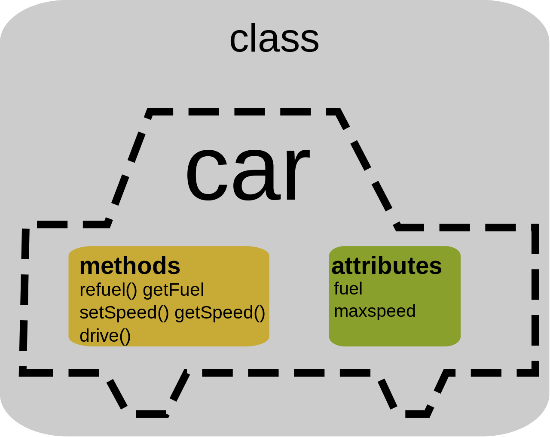
5. การนำไปใช้ (Deployment) หรือการแจกจ่ายซอฟต์แวร์ให้กับลูกค้า จะทำหลังจากการทดสอบเสร็จสิ้น

6. การดูแลรักษา (Maintenance) โดยการแก้ปัญหา หรือเพิ่มคุณค่าในผลิตภัณฑ์ภายหลังการแจกจ่าย ทำได้ด้วยการแจกจ่ายผลิตภัณฑ์ที่ถูกการปรับปรุงตามความต้องการต่าง ๆ ที่เปลี่ยนแปลงไป

**2.2 การเขียนโปรแกรมเชิงวัตถุ**

อานนท์ หลงหัน (2556) กล่าวว่า การเขียนโปรแกรมเชิงวัตถุ หรือ Object Oriented Programming (OOP) เป็นหลักการการเขียนโปรแกรม โดยอาศัยแนวคิดเกี่ยวกับการมองปัญหาต่าง ๆ ให้เป็นวัตถุ หลักการของ OOP ได้รับการยอมรับและพัฒนามาใช้ในภาษาโปรแกรมและระบบปฏิบัติการต่าง ๆ รวมไปถึงระบบปฏิบัติการ Android ด้วย

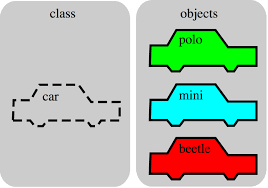
OOP มีหลักการมาจาก “ธรรมชาติของวัตถุ” หมายความว่า OOP จะมองสิ่งแต่ละสิ่งถือเป็น “วัตถุ” ไม่ว่าจะมีสีแดงหรือสีเขียว ขนาดใหญ่หรือเล็ก มีความกว้าง ความยาว ความสูงเท่าใด มันก็คือวัตถุชิ้นหนึ่งเหมือนกัน



ภาพที่ 2.2 คุณสมบัติ (Attribute) และขั้นตอนวิธี (Method) ของวัตถุ (รถยนต์)  
ที่มา: Friedland (2019)

วัตถุแต่ละอย่างนั้น ต่างก็มีลักษณะและวิธีการใช้งานเป็นของตัวเอง คือมีรูปร่าง ลักษณะ และการใช้งานที่แตกต่างกันออกไป เราจะเรียกคุณลักษณะของวัตถุว่า Attribute (คุณสมบัติ) และจะเรียกวิธีการใช้งานวัตถุว่า Method (ขั้นตอนวิธี) ตัวอย่างเช่น “ขวดน้ำพลาสติก เป็นวัตถุทรงกระบอกกลวง ทำจากพลาสติก บรรจุน้ำสะอาดสำหรับดื่ม สามารถใช้ดื่มน้ำภายในขวดได้” จากข้อความข้างต้น สามารถเปรียบเทียบได้ว่า คุณสมบัติของวัตถุนี้คือ “วัตถุสามมิติ รูปร่างทรงกระบอกกลวง วัสดุเป็นพลาสติก สามารถบรรจุของเหลวได้ ขณะนี้บรรจุน้ำสะอาดสำหรับดื่ม” ส่วนขั้นตอนวิธีของขวดน้ำพลาสติกคือ “วัตถุนี้สามารถสั่งให้ดื่มน้ำภายในขวดได้” การสั่งให้ดื่มน้ำภายในขวดก็เปรียบเสมือนการทำงานของโปรแกรมโดยมีตัวแปรเป็นขวดน้ำพลาสติกนั่นเอง

นอกจากนี้ สามารถกำหนดขอบเขตให้กับคุณสมบัติ (Attribute) และขั้นตอนวิธี (Method) แต่ละอย่างของวัตถุได้ด้วย เช่น หนังสือที่มีคุณสมบัติคือมีข้อความอยู่ภายใน ไม่สามารถอ่านได้ทันที ต้องสั่งให้เปิดหนังสือก่อน จะแก้ไขข้อความก็ไม่สามารถทำได้ทันที ต้องสั่งให้เขียนข้อความภายในหนังสือก่อน เป็นต้น หรือตู้เอทีเอ็ม ที่ไม่สามารถสั่งให้ตู้นำเงินมาให้เราได้ทันที แต่ต้องป้อนบัตร ใส่รหัสผ่าน กดเลือกถอนเงิน ตรวจสอบจำนวนเงิน หักเงินในบัญชีให้เสร็จสิ้นก่อนจึงจะนำเงินออกมาได้ เป็นต้น



ภาพที่ 2.3 ประเภท (Class) และวัตถุ (Object) ที่สอดคล้องกัน   
ที่มา: Friedland (2019)

วัตถุแต่ละชิ้นยังสามารถแบ่งออกเป็นประเภท (Class) ที่ต่างกันได้ โดยวัตถุที่มีประเภท (Class) เดียวกัน จะทำหน้าที่ในลักษณะเดียวกันในเวลาทำงานจริง นอกจากนี้ ประเภทและคุณสมบัติของวัตถุยังสามารถสืบทอดต่อกันได้ เช่น วัตถุประเภทรถยนต์ สามารถขับได้ บีบแตรได้ หลังจากนั้น อาจจะมีวัตถุประเภทรถสปอร์ต ซึ่งสืบทอดมาจากประเภทรถยนต์ สามารถขับได้เช่นกันแต่อาจจะเร็วขึ้น บีบแตรได้เหมือนกัน เป็นต้น

แต่การสืบทอดวัตถุนั้นก็มีข้อจำกัดคือ สามารถสืบทอดได้เพียงประเภทเดียวเท่านั้น เช่น วัตถุประเภทนก สืบทอดมาจากสัตว์ปีก ส่วนวัตถุประเภทเครื่องบินสืบทอดมาจากเครื่องจักรเป็นต้น ถ้าเราต้องการโปรแกรมที่สามารถสั่งให้ทั้งนกและเครื่องบินออกตัวได้ มันก็จะทำได้ยากถ้าทั้งสองไม่ได้สืบทอดมาจากวัตถุต้นกำเนิดเดียวกัน วิธีการแก้ไขคือการให้วัตถุทั้งสองมีส่วนต่อประสานหรือ Interface ซึ่งจะบังคับให้ทั้งสองวัตถุต้องมีวิธีการหรือ Method ของตัว Interface นั้น เพื่อทำให้การเรียกใช้งานสามารถทำได้ผ่าน Method ของตัว Interface นี้ได้เลย

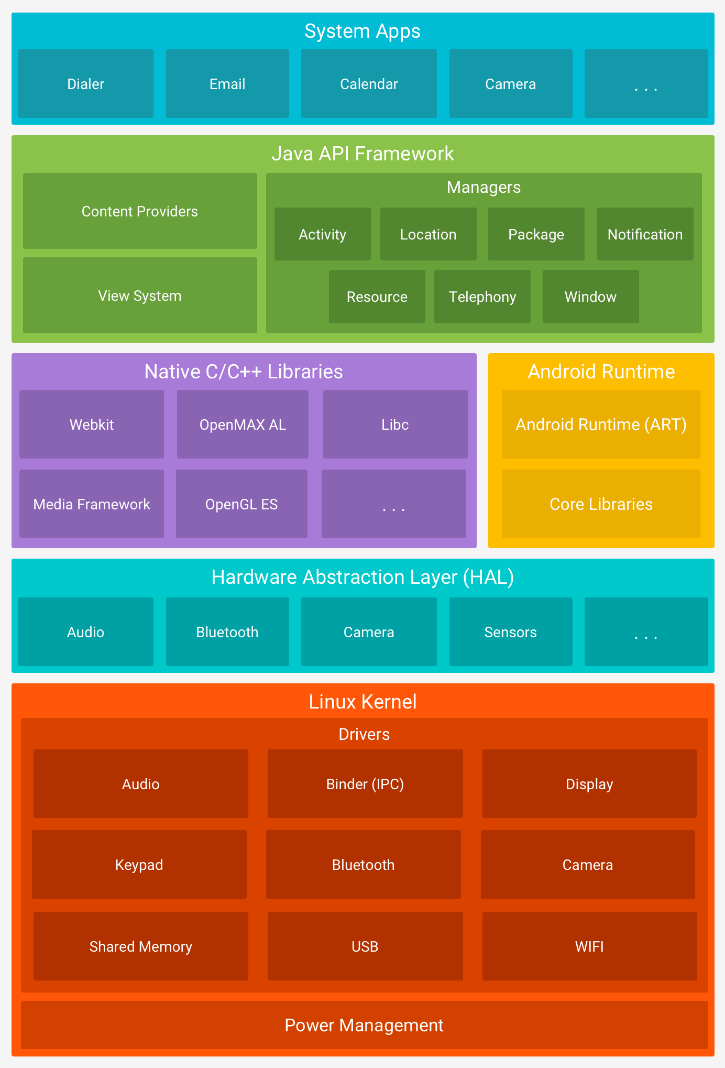
นอกจากนี้ Interface ยังสามารถเรียกใช้งานได้ง่ายและสามารถใช้ได้พร้อมกันหลาย Interface ในวัตถุเดียว เช่น รถยนต์สามารถเร่งเครื่องได้ สามารถเบรกได้ สามารถปัดน้ำฝนได้ สามารถเลี้ยวได้ และสามารถทำได้อีกหลายหน้าที่ ถ้าต้องการจะสั่งให้ทำหน้าที่ใดหน้าที่หนึ่งได้ในหลายวัตถุพร้อมกัน ก็สามารถให้วัตถุเหล่านั้นใช้ Interface ร่วมกันได้ ในกรณีนี้คือ รถยนต์มี Interface สำหรับการเร่งเครื่อง การเบรก การปัดน้ำฝน การเลี้ยว ทำให้สามารถสั่งการร่วมกับวัตถุอื่น เช่น เรือ ที่เร่งเครื่องได้ เลี้ยวได้ แต่วิธีการอาจจะไม่เหมือนกัน เป็นต้น

ประโยชน์ในการเขียนโปรแกรมโดยอาศัยหลักการเชิงวัตถุนั้น จะมีอยู่ที่การออกแบบโปรแกรม ด้วยคุณลักษณะของโปรแกรมที่ใช้วัตถุเป็นเครื่องมือในการทำงาน ทำให้การใช้งานโค้ดซ้ำ ๆ มีความง่าย รวดเร็ว และจัดระเบียบได้ง่ายในฐานะของการเป็น “วัตถุ” เมื่อโปรแกรมทำงาน โปรแกรมก็เพียงแค่สั่งระบบให้ทำการเรียกใช้ สร้าง แก้ไข ลบ ดัดแปลงวัตถุต่าง ๆ ตามลักษณะการใช้งานที่โปรแกรมออกแบบไว้ ทำให้การเขียนโปรแกรมเชิงวัตถุ เป็นหลักการที่จำเป็นต่อการพัฒนาโปรแกรมสำหรับการใช้ในโปรแกรมสำหรับผู้ใช้โดยเสมอ

**2.3 ภาษา Java**

ภาษา Java เป็นภาษาโปรแกรมที่พัฒนาโดยใช้หลักการเขียนโปรแกรมเชิงวัตถุ พัฒนาโดยบริษัท Sun Microsystems ในปี ค.ศ. 1995 ซึ่งต่อมาถูกซื้อกิจการโดยบริษัท Oracle ในปี ค.ศ. 2010 (The Guardian, 2009) โดยถูกออกแบบมาให้เป็นภาษาที่เมื่ออ่านแล้วเข้าใจได้ง่าย บวกกับการออกแบบให้มีความคล้ายคลึงกับภาษา C++ ที่ใช้กันอย่างกว้างขวาง ทำให้ภาษา Java กลายเป็นภาษาที่ใช้กันแพร่หลายภายในเวลาอันรวดเร็ว (Oracle, 2019) แอปพลิเคชันบนระบบปฏิบัติการ Android ถูกพัฒนาโดยการใช้ภาษา Java เป็นหลัก ทำให้ภาษา Java เป็นพื้นฐานความรู้ที่จำเป็นในการศึกษาก่อนเริ่มการพัฒนาแอปพลิเคชันเสมอ

**2.4 ระบบปฏิบัติการ Android**



ภาพที่ 2.4 สถาปัตยกรรมภายในของ Android  
ที่มา: Android Developers (2019)

จากบทความของ Android Developers (2019) ที่ได้กล่าวถึงสถาปัตยกรรมหลักของระบบปฏิบัติการ Android ได้ทำการจำแนกออกไว้เป็น 6 ส่วนหลัก ๆ ตามภาพดังนี้

1. Linux Kernel เคอร์เนลระบบปฏิบัติการ Linux การนำมาใช้จึงทำให้ระบบปฏิบัติการมีความเสถียรและความปลอดภัยเทียบเท่ากับระบบปฏิบัติการ Linux ที่ได้รับการพัฒนา และดูแลรักษาหลายสิบปี จึงมั่นใจได้ในด้านเสถียรภาพและความปลอดภัยของ Android

2. Hardware Abstraction Layer คือส่วนที่ทำการมอบการใช้งาน Hardware ต่าง ๆ เช่น กล้อง ลำโพง หรือเซนเซอร์ต่าง ๆ โดยทำการควบคุมการใช้งาน Hardware ผ่านสิทธิ์ต่าง ๆ ที่แอปพลิเคชันมีอยู่

3. Android Runtime คือ ส่วนที่คอยจัดการการทำงานของแอปพลิเคชัน โดยจะทำการแบ่งพื้นที่ให้แอปพลิเคชันทำงานแยกกัน และทำการควบคุมความปลอดภัยไม่ให้แอปพลิเคชันเข้าไปในกระบวนการอื่น ๆ

4. Native C++ Library คือกลุ่มของ Library ในภาษา C++ ที่ใช้ในการพัฒนาแอปพลิเคชันในส่วนเชิงลึก ซึ่งรวมไปถึง Android Runtime ด้วย การใช้งาน Libraries เหล่านี้จึงมีประโยชน์อย่างมากหากต้องการประสิทธิภาพที่สูง เช่น เกมแข่งรถที่ต้องประมวลผลกราฟฟิกในประมาณมาก เป็นต้น ในกรณีเหล่านี้การนำภาษา C++ มาใช้ควบคุมโดยตรง (หรือเพียงบางส่วน) จะได้ผลดีกว่าการใช้ Java API Framework

5. Java API Framework คือส่วนของ API ที่พัฒนาโดย Android ในภาษา Java ที่เป็นส่วนของการทำงานหลักในแอปพลิเคชัน ยกตัวอย่างเช่น View System สำหรับการแสดงผลภายในแอปพลิเคชัน และ Resource Manager สำหรับการเก็บข้อมูลภายในแอปพลิเคชัน เช่น สี รูปภาพ ผังการแสดงผล เป็นต้น

6. System Apps หรือแอปพลิเคชันระบบพื้นฐานที่ติดตั้งมาในอุปกรณ์ Android ทุกเครื่อง โดยปกติแล้วแอปพลิเคชันเหล่านี้ เช่น แอปเครื่องคิดเลข (Calculator) หรือแอปข้อความ (Messages) มักจะเป็นการทำงานปกติในโทรศัพท์เคลื่อนที่ ประโยชน์ของการมี System Apps คือ เราไม่จำเป็นที่จะต้องใช้วิธีอะไรที่ยุ่งยากเพื่อทำสิ่งที่โทรศัพท์ทั่วไปทำได้อยู่แล้ว เช่น โทรศัพท์ หรือ รับส่งข้อความ SMS แค่เพียงเราส่งข้อมูลไปให้แอปพลิเคชันต่าง ๆ เหล่านี้ทำให้แทน ยกตัวอย่างการอ่านข้อมูล SMS ของแอปพลิเคชัน แทนที่จะไปตรวจสอบข้อมูลต่าง ๆ ให้ยุ่งยาก การอ่านข้อมูลผ่านแอปข้อความของ Android ที่สามารถส่ง / รับ SMS ได้ มักจะเป็นวิธีการที่สะดวกกว่า

**2.5 โครงสร้างข้อมูล**

โครงสร้างข้อมูล คือคำนิยามของการจัดการข้อมูลในรูปแบบต่าง ๆ โดยทำการจัดเป็นกลุ่ม เป็นชนิด อย่างมีกฎและรูปแบบ เพื่อให้การเก็บข้อมูล การนำข้อมูลกลับมาใช้ และการประมวลผลทำได้อย่างมีประสิทธิภาพ ซึ่งโดยปกติแล้ว ข้อมูลในภาษาเครื่องจะมีแค่สัญญาณ 0 และ 1 เท่านั้น จึงเป็นเรื่องที่แทบจะเป็นไปไม่ได้เลยถ้าต้องการสร้างโปรแกรมให้ผู้ใช้อ่านได้ ถ้าไม่มีการจัดข้อมูลให้เป็นระเบียบและเป็นมาตรฐาน นอกจากนี้ การที่ไม่จัดรูปแบบข้อมูลนั้น การเขียนโปรแกรมก็จะเป็นสิ่งที่แทบจะเป็นไปไม่ได้เช่นกัน การมีโครงสร้างข้อมูลจึงเป็นสิ่งที่จำเป็นสำหรับภาษาโปรแกรมส่วนใหญ่ในปัจจุบัน (ภาชญา เชี่ยวชาญ, 2562)

การแบ่งประเภทของโครงสร้างข้อมูลมักจะแบ่งตามลักษณะการจัดเรียงข้อมูล ซึ่งแบ่งได้ดังนี้

1. โครงสร้างข้อมูลประเภทพื้นฐาน คือ ประเภทของโครงสร้างข้อมูลที่มีการจัดข้อมูลเป็นแบบพื้นฐาน โครงสร้างข้อมูลประเภทนี้มักจะเป็นการทำให้ข้อมูลทั่วไป เช่น ตัวเลข ตัวอักษร สามารถบันทึกลงบนสัญญาณ 0 / 1 ได้ และเป็นพื้นฐานให้กับโครงสร้างข้อมูลประเภทอื่น ๆ

|  |  |  |  |
| --- | --- | --- | --- |
| **ชื่อเรียก** | **ชนิด** | **ขนาด (ใน Java)** | **ค่าที่เก็บได้ (ใน Java)** |
| boolean | ค่าความจริง | ≤ 1 B (≤ 8 bit) | true (จริง) / false (เท็จ) |
| char | ตัวอักษร | 2 B (16 bit) | ตัวอักษร 0-255 (ASCII) |
| byte | จำนวนเต็ม | 1 B (8 bit) | -128 ถึง 127 |
| short | 2 B (16 bit) | -32,768 ถึง 32,767 |
| int | 4 B (32 bit) | -2,147,483,648 ถึง 2,147,483,647 |
| long | 8 B (64 bit) | ± 9.223372 × 1018 |
| float | จำนวนจริง (มีทศนิยม) | 4 B (32 bit) | ± 3.402823 × 1038 |
| double | 8 B (64 bit) | ± 1.797693 × 10308 |

ภาพที่ 2.5 โครงสร้างข้อมูลประเภทพื้นฐาน ในภาษา Java  
ที่มา: Kumar (2019)

1. โครงสร้างข้อมูลประเภทผสม คือ การนำโครงสร้างข้อมูลประเภทต่าง ๆ มาจัดเรียงให้เป็นระเบียบ หรือเป็นลักษณะที่กำหนดไว้ เพื่อทำให้การบันทึกข้อมูลทำได้ง่ายขึ้นและเป็นหมวดหมู่ ยกตัวอย่างเช่น
   1. แถวลำดับ (Array) เป็นการจัดเรียงข้อมูลให้เรียงกันเป็นเส้นตรง โดยส่วนใหญ่ข้อมูลภายในแถวลำดับมักเป็นชนิดเดียวกัน ข้อมูลภายในทุกข้อมูลจะมีจำนวนเต็มกำกับเป็นตำแหน่ง ขนาดของแถวลำดับมักจะมีขนาดคงที่ แต่สามารถปรับให้ขยายขนาดได้ด้วยวิธีต่าง ๆ ตัวอย่างการใช้งานเช่น สายอักขระ (String) ที่เป็นแถวลำดับของตัวอักษร ใช้สำหรับการเก็บข้อมูลในโปรแกรมเอกสาร เช่น Microsoft Word เป็นต้น
   2. ระเบียน (Record) เป็นโครงสร้างข้อมูลแบบหนึ่ง ที่ประกอบไปด้วยโครงสร้างข้อมูลอื่น ๆ เรียกว่า เขตข้อมูล ส่วนใหญ่ระเบียนจะมีจำนวนเขตข้อมูลและลำดับที่คงที่ การเข้าถึงข้อมูลภายในจะเข้าถึงได้ผ่านการกำหนดชื่อของเขตข้อมูล ยกตัวอย่างเช่น โครงสร้างข้อมูลชื่อและนามสกุล ประกอบไปด้วย ชื่อที่เป็นสายอักขระ และนามสกุลที่เป็นสายอักขระเช่นกัน เมื่อเรียกใช้งานก็สามารถรับค่าชื่อจากโครงสร้างของชื่อและนามสกุลได้ สามารถแก้ไขได้เช่นกัน เป็นต้น ระเบียนเป็นโครงสร้างข้อมูลอย่างง่ายชนิดหนึ่งที่ใช้กันอย่างกว้างขวาง เนื่องจากมีประโยชน์ในการเขียนโปรแกรมเชิงวัตถุอย่างมาก โดยส่วนใหญ่ การออกแบบวัตถุในภาษาดังกล่าว จะอาศัยหลักการมองว่าคุณสมบัติ (Attribute) ของวัตถุคือระเบียนชนิดหนึ่ง ส่วนขั้นตอนวิธี (Method) ของวัตถุคือคำสั่งในการจัดการระเบียน เป็นต้น
   3. ยูเนียน (Union) เป็นโครงสร้างข้อมูลประเภทหนึ่ง สามารถเก็บค่าได้หลายประเภทตามแต่จะกำหนดไว้ ซึ่งสามารถเปลี่ยนแปลงประเภทที่จะเก็บได้ในเวลาทำงานจริง เช่น ยูเนียนของตัวอักษรและตัวเลข สามารถเก็บตัวเลขก็ได้ เก็บตัวอักษรก็ได้ แต่ไม่สามารถเก็บได้ทั้งตัวเลขและตัวอักษรพร้อมกัน การใช้ยูเนียนจะมีประโยชน์ในด้านพื้นที่ ที่ใช้พื้นที่ในการเก็บข้อมูลน้อยกว่า แต่จะแลกมาด้วยความยุ่งยากในการตรวจสอบประเภทของค่าที่เก็บไว้ จึงมีการพัฒนายูเนียนพร้อมป้ายระบุ (Tagged Union) สำหรับการตรวจสอบว่าในขณะนั้น ยูเนียนเก็บข้อมูลประเภทใดอยู่นั่นเอง
2. โครงสร้างข้อมูลประเภทนามธรรม คือ โครงสร้างข้อมูลที่จำเป็นที่จะต้องระบุประเภทให้แน่ชัดก่อนใช้งาน เนื่องจากโครงสร้างข้อมูลประเภทนี้ถูกออกแบบให้สามารถเก็บข้อมูลได้จากทุกประเภท และออกแบบมาเพื่อช่วยเหลือการทำงานของโปรแกรม ที่ไม่สามารถทำได้ หรือทำได้แต่ประสิทธิภาพต่ำกว่า ถ้าหากใช้โครงสร้างข้อมูลประเภทผสม ในเอกสารนี้จะยกตัวอย่างเฉพาะแถวลำดับเชื่อมโยงที่ใช้ในงานวิจัยเท่านั้น
   1. แถวลำดับเชื่อมโยง (Associative Array) คือแถวลำดับประเภทหนึ่งที่เปลี่ยนจากการใช้จำนวนเต็มในการกำกับตำแหน่ง เป็นข้อมูลประเภทใด ๆ แทน ซึ่งการค้นหาจะเริ่มจากการใส่ค่ากำกับ แล้วแถวลำดับเชื่อมโยงจะคืนค่าที่ได้กลับมา หรือคืนค่าว่าง (Null Value) ถ้าไม่มีค่าที่กำกับในแถวลำดับเชื่อมโยงนั้น แถวลำดับเชื่อมโยงนั้นสามารถออกแบบได้หลากหลายวิธี แต่ขอไม่กล่าวถึงเพราะเนื้อหาในการออกแบบและประสิทธิภาพนั้นไม่ใช่สาระสำคัญในการพัฒนา  
      แอปพลิเคชัน

**2.6 งานวิจัยที่เกี่ยวข้อง**

สุปรียา เดชรักษา และ ทิพวิมล ชมพูคำ (2560) ได้ศึกษาและทำการพัฒนาแอปพลิเคชันบัญชีรายรับ-รายจ่ายบนระบบปฏิบัติการ Android โดยผลการศึกษาได้ผลการประเมินคุณภาพจากผู้เชี่ยวชาญด้านการพัฒนาแอปพลิเคชันจำนวน 3 คน อยู่ในระดับมาก (Avg. 4.39, S.D. 0.49) และผลการสอบถามความพึงพอใจจากนักศึกษาสาขาวิชาเทคโนโลยีคอมพิวเตอร์และการสื่อสารชั้นปีที่ 4 จำนวน 30 คน มหาวิทยาลัยราชภัฏมหาสารคาม อยู่ในระดับมากที่สุด (Avg. 4.56, S.D. 0.53)

ชัชฎาพร ปุกแก้ว, สุพงศ์ แดงสุริยศรี, เทวา พรหมนุชานนท์ และ ณพัชธา โฉมศรี (2562) ได้ศึกษาและทำการพัฒนาแอปพลิเคชันบัญชีรายรับ-รายจ่ายบนระบบปฏิบัติการ Android ชื่อว่า “จัดการเงินของฉัน” โดยได้มีการกล่าวถึงการศึกษาความเป็นไปได้ การกำหนดความต้องการ การออกแบบระบบงานที่มีความน่าสนใจ และการนำแอปพลิเคชันที่พัฒนาขึ้นแล้วลงจำหน่ายใน Google Play Store ได้ผลการศึกษาว่า แอปพลิเคชันมีการติดตั้งเป็นจำนวนมากกว่า 500 ครั้ง และการให้คะแนนมากกว่า 3 ดาว ตรงตามเป้าหมายที่ตั้งไว้

**บทที่ 3  
วิธีการดำเนินการวิจัย**

เนื่องจากโครงงานนี้มีวัตถุประสงค์เพื่อพัฒนาแอปพลิเคชันซึ่งเป็นซอฟต์แวร์ ภายในโครงงานจะใช้ขั้นตอนการพัฒนาซอฟต์แวร์มาใช้ในการดำเนินงาน ในโครงงานนี้จะใช้แบบจำลองน้ำตก (Waterfall Model) เป็นขั้นตอนดำเนินงาน วิธีดำเนินงาน มีดังนี้

**3.1 การกำหนดและวิเคราะห์ความต้องการ**

ตามวัตถุประสงค์ของโครงงาน คณะผู้วิจัยต้องการพัฒนาแอปพลิเคชันสำหรับ

* 1. การแสดงผลข้อมูลตารางเรียน
  2. การแสดงผลข้อมูลสถานที่สอน

ซึ่งได้ทำการกำหนดและวิเคราะห์รายละเอียดต่าง ๆ ดังนี้

* 1. ข้อมูลของตารางเรียน หมายถึง การกำหนดห้องเรียนของนักเรียน (ม.602) ไว้ในชุดข้อมูลล่วงหน้า ประเภทของข้อมูลที่บรรจุในแต่ละคาบเรียนที่นำมาประกอบเป็นตารางเรียน ประกอบด้วย หมายเลขคาบ ชื่อวิชา รหัสวิชา รายชื่อของอาจารย์ผู้สอน และสถานที่เรียน สำหรับคาบอื่น ๆ ที่ไม่เกี่ยวข้องกับการเรียนนั้น จะมีระบุไว้ด้วย
  2. ข้อมูลของสถานที่สอน หมายถึง การจัดเรียงข้อมูลกลุ่มของอาจารย์ โดยข้อมูลของอาจารย์แต่ละท่านแบ่งเก็บไว้เป็น 50 ส่วน แต่ละส่วนจะระบุวันที่และคาบไว้ (ลักษณะคล้ายข้อมูลตารางเรียน) ซึ่งจะเก็บข้อมูลว่า ในคาบนั้น ๆ อาจารย์ท่านดังกล่าวสอนอยู่หรือไม่ หากกำลังสอนอยู่ โปรแกรมจะเก็บข้อมูลไว้ด้วยว่า อาจารย์กำลังสอนนักเรียนห้องใด สถานที่ใด เป็นต้น

นอกจากนี้ แอปพลิเคชันต้องมีความสามารถในการเพิ่ม แก้ไข ดัดแปลงข้อมูล และการส่งต่อข้อมูลระหว่างอุปกรณ์ โดยไม่ทำระบบให้มีขนาดใหญ่เพื่อประหยัดค่าบำรุงรักษาแอปพลิเคชัน

**3.2 การออกแบบระบบ**

คณะผู้วิจัยได้ทำการออกแบบระบบ ลักษณะการทำงานภายในแอปพลิเคชันออกเป็น 3 ส่วนด้วยกันตามหน้าที่การใช้งานต่าง ๆ ดังนี้

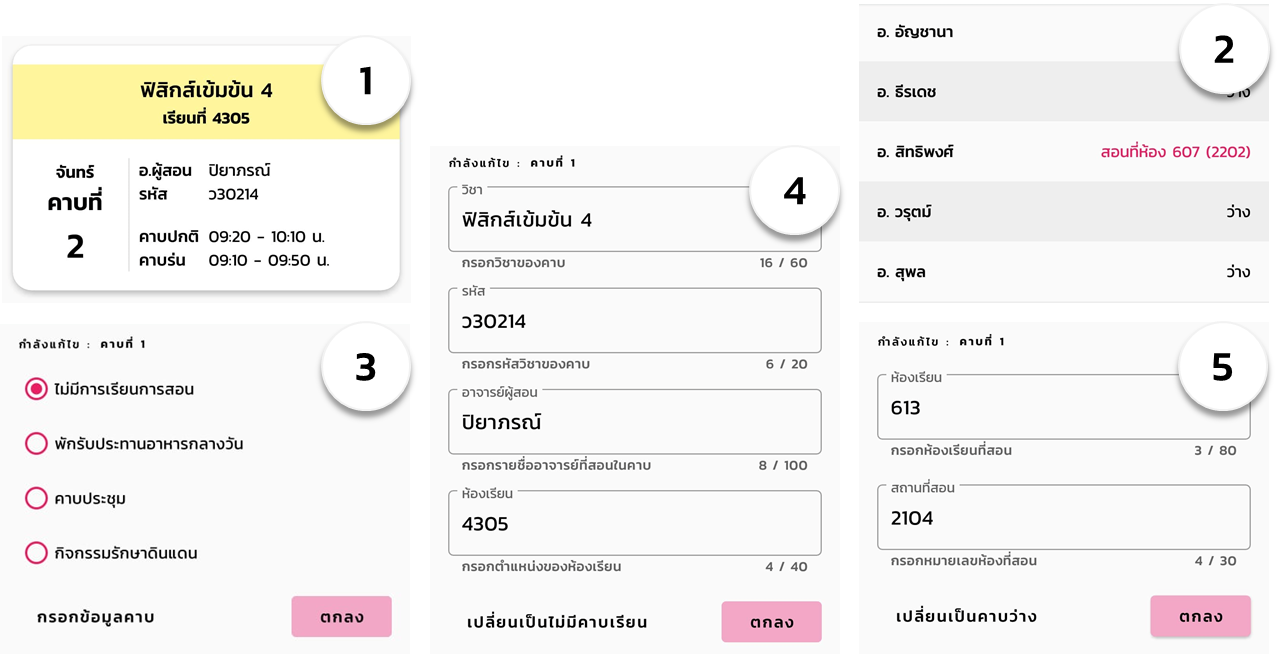
* 1. ระบบการบันทึกข้อมูล จะนำข้อมูลภายในแอปพลิเคชัน (ข้อมูลตารางเรียนและข้อมูลสถานที่สอน) แปลงเป็นสตริง เพื่อนำไปเก็บไว้ใน Shared Preferences หรือส่วนการตั้งค่าภายในแอปพลิเคชัน (โดยปกติแล้ว Shared Preferences มักจะถูกใช้ในการเก็บข้อมูลที่เล็กเพียงพอ แอปพลิเคชันที่พัฒนาขึ้นเก็บข้อมูลในปริมาณน้อยจึงสามารถใช้ได้โดยไม่มีปัญหา)
  2. ระบบการจัดเรียงและประมวลผลข้อมูล ทุกครั้งที่เปิดแอปพลิเคชัน ระบบจะดึงข้อมูลจาก Shared Preferences (แบบสตริง) แปลงกลับให้เข้าไปในระบบข้อมูลในแอปพลิเคชัน ข้อมูลที่ดึงมาจะถูกนำไปเรียงเป็นแถวลำดับเชื่อมโยงตามระบบที่ได้นำไว้เพื่อเตรียมข้อมูลสำหรับแสดงผล
  3. ระบบการแสดงผล จะใช้ระบบ View (ระบบออกแบบ User Interface ของ Google) วิธีการใช้คือให้นำข้อมูลที่ประมวลผลและจัดเรียงแล้วไปเรียงบน View (ในตำแหน่งที่ต้องการ ตามที่ได้ออกแบบไว้) แล้ว View จะทำหน้าที่ติดต่อกับ Android ให้แสดงผลออกมาเป็น User Interface หรือหน้าตาของแอปพลิเคชัน โดยเราสามารถกำหนดคำสั่งที่จะทำงานเมื่อผู้ใช้มีการตอบสนองกับ View เช่น กด กดค้าง ลาก ปัด View ไปมาแล้ว จะเกิดสิ่งใดสิ่งหนึ่งขึ้น เช่น ไปหน้าใหม่ เป็นต้น

ในการจัดเก็บข้อมูล ได้ทำการออกแบบโครงสร้างข้อมูลไว้ ดังนี้

* + ข้อมูลตารางเรียน จะแบ่งออกเป็นข้อมูลคาบเรียนจำนวน 50 คาบ โดยจะถูกจัดเรียงตามเวลาที่เรียน เช่น ข้อมูลแรกจะเก็บคาบเรียนวันจันทร์ คาบที่ 1 ภายในข้อมูลนั้นจะประกอบไปด้วย ชื่อวิชา รหัสวิชา ห้องที่เรียน รายชื่ออาจารย์ผู้สอน (สตริง) หมายเลขระบุประเภท (มีการเรียนการสอนหรือไม่ ถ้าไม่มีเพราะอะไร เป็นต้น) และหมายเลขระบุคาบเรียนดังภาพที่ 3.1 [A] และภาพที่ 3.2 [1] ซึ่งเราสามารถแก้ไขข้อมูลต่างๆ ได้โดยการเลือกคาบเรียนที่ต้องการในหน้าของภาพที่ 3.1 [C] เมื่อกดเลือกแล้ว ถ้ามีการเรียนการสอนจะแสดงดังภาพที่ 3.2 [4] แต่ถ้าหากไม่มีจะแสดงดังภาพที่ 3.2 [3] ซึ่งผู้ใช้สามารถสลับระหว่างคาบที่มีการเรียนการสอน (3.2 [4]) และไม่มี (3.2 [3]) ได้โดยการเลือก “เปลี่ยนเป็นไม่มีคาบเรียน” หรือ “กรอกข้อมูลคาบตามที่ต้องการ” ในการยืนยันการกรอกข้อมูลให้เลือก “ตกลง” เป็นต้น



ภาพที่ 3.1 หน้าแสดงผลและหน้าแก้ไขข้อมูลภายในแอปพลิเคชัน



ภาพที่ 3.2 ส่วนประกอบต่าง ๆ ภายในหน้าแสดงผลและหน้าแก้ไขข้อมูล

* + ข้อมูลสถานที่สอน จะแบ่งออกเป็นข้อมูลเป็นกลุ่มตามอาจารย์แต่ละท่าน สำหรับอาจารย์แต่ละท่านจะมีการบันทึกสถานะการสอน คือ สอนอยู่ หรือว่างจากการสอน เป็นจำนวนเท่ากับคาบเรียนคือ 50 ชุด สถานการณ์สอนต่าง ๆ จะเก็บรหัสอาจารย์ ห้องเรียน สถานที่สอน และหมายเลขกำกับคาบเรียน ในการแสดงผลเมื่อผู้ใช้ได้ใช้งานดังภาพที่ 3.1 [A] เมื่อเลือกที่บัตรคาบเรียนดังภาพที่ 3.2 [1] จะเปลี่ยนแปลงลักษณะกลายเป็นดังภาพที่ 3.1 [B] / 3.2 [2] ในหน้านี้จะมีรายชื่ออาจารย์ทั้งหมด และจะบอกข้อมูลว่า ในคาบนี้นั้น อาจารย์แต่ละคนว่างอยู่ หรือกำลังสอนอยู่ที่ห้องใด เป็นต้น การแก้ไขข้อมูลเหล่านี้สามารถทำได้โดยการเลือกครูจากภาพที่ 3.1 [D] แล้วแก้ไขข้อความดังภาพที่ 3.1 [E] เมื่อกดปุ่มแก้ไขแล้วจะขึ้นกล่องดังภาพที่ 3.2 [5]

**3.3 การดำเนินงานพัฒนา**

คณะผู้จัดทำได้ทำการพัฒนาแอปพลิเคชันตามที่ได้ออกแบบไว้ข้างต้น ตั้งแต่วันที่ 20 มีนาคม พ.ศ. 2562 ถึงวันที่ 23 เมษายน พ.ศ. 2562 เป็นเวลา 33 วัน โดยจะแบ่งเป็นขั้นตอนต่าง ๆ ดังนี้

* 1. ทำการเขียนโครงสร้างข้อมูลต่าง ๆ ตั้งแต่ข้อมูลคาบเรียน ไปจนครบข้อมูลที่ต้องการทั้งหมด เช่น ข้อมูลคาบเรียน มารวมกันเป็นข้อมูลตารางเรียน ข้อมูลสถานที่สอน มารวมกันเป็นข้อมูลอาจารย์ แล้วนำรายชื่ออาจารย์ทั้งหมดมาเรียงกันเป็นต้น เพื่อเป็นพื้นฐานให้กับการนำข้อมูลมาจัดเรียง
  2. ทำการวางระบบแปลงข้อมูลแอปพลิเคชันทั้งหมดในรูปแบบสตริง สำหรับการนำข้อมูลระหว่างการทำงานบันทึกลงบนหน่วยความจำ หรือเป็นข้อมูลที่ใช้ส่งออกให้โทรศัพท์เครื่องอื่น ๆ นอกจากนี้ การวางระบบจะรวมถึงการแปลงข้อมูลจากสตริงกลับเป็นข้อมูลแอปพลิเคชันด้วย สำหรับการรับข้อมูลจากหน่วยความจำและข้อมูลที่นำเข้ามาจากโทรศัพท์เครื่องอื่น ๆ
  3. ทำการแปลงส่วนประกอบการแสดงผลต่าง ๆ ให้อยู่ในระบบ View เตรียมไว้ใช้แสดงผล โดยจะเก็บในลักษณะไฟล์ XML เตรียมพร้อมสำหรับการเชื่อมต่อกับระบบข้อมูลที่ได้ทำการพัฒนาไว้
  4. ทำการร่างโครงร่างการทำงานสำหรับหน้าต่าง ๆ ในแอปพลิเคชัน และลำดับการทำงานภายในแอปพลิเคชัน หลังจากนั้นจึงเขียนคำสั่งควบคุมการทำงานของส่วนต่าง ๆ ของการแสดงผลให้ทำหน้าที่เป็นปุ่ม เช่น เมื่อกดที่ใบคาบเรียน (ภาพที่ 3.2 [1]) จะแสดงผลหน้าที่มีข้อมูลสถานที่สอนของอาจารย์แต่ละท่านในคาบเรียนนั้น ๆ (ภาพที่ 3.1 [B]) เป็นต้น
  5. ทำการเชื่อมโยงข้อมูลต่าง ๆ ภายในระบบให้ส่วนประกอบการแสดงผลทำงานตรงกันกับข้อมูล โดยการกระทำเหล่านี้จะเกิดขึ้นก่อนที่ภาพภายในหน้านั้น ๆ จะแสดงผล ตัวอย่างเช่น สำหรับการแสดงผลคาบเรียน ให้นำข้อมูลคาบเรียนที่บันทึกไว้ มาใส่ในใบคาบเรียน (ภาพที่ 3.2 [1]) ให้ตรงตามข้อมูลคาบเรียนแต่ละส่วนที่มีอยู่ในแอปพลิเคชัน ณ ขณะนั้น สำหรับการแสดงผลสถานที่สอนก็จะค้นหาข้อมูลสถานที่สอนของอาจารย์ทุกท่านในคาบเรียนเดียวกันกับที่เลือกไว้ และแสดงผลออกมาเป็นคู่ ๆ (ตามภาพที่ 3.1 [B]) ถ้าคาบใดอาจารย์ว่างจะแสดงตัวอักษรสีเทา หากอาจารย์กำลังสอนอยู่ก็จะแสดงตัวอักษรสีชมพู เป็นต้น
  6. ทำการเพิ่มคุณสมบัติการทำงานต่าง ๆ เช่น การล้างข้อมูลภายในการกด 3 ครั้ง, การส่งออกข้อมูลเป็นรูปแบบตัวอักษร และการนำเข้าข้อมูลที่มีกล่องให้ใส่วางข้อมูล รวมถึงการแสดงผลตัวอย่างข้อมูลเพื่อตรวจสอบว่ามั่นใจในการแทนที่ข้อมูลด้วย รวมถึงการตกแต่งรูปลักษณ์ภายในแอปพลิเคชัน สีภายในแอปพลิเคชัน ขนาดและฟอนต์ตัวอักษร และหน้าปกแอปพลิเคชัน

**3.4 การทดสอบระบบ และทดลองใช้**

หลังจากที่ได้แอปพลิเคชันจากการพัฒนาเรียบร้อยแล้ว จึงนำมาทดสอบในระหว่างวันที่ 25 เมษายน พ.ศ. 2562 ถึงวันที่ 9 พฤษภาคม พ.ศ. 2562 เป็นระยะเวลา 15 วัน การทดสอบระบบนี้มีวัตถุประสงค์เพื่อที่จะแก้ไขข้อผิดพลาดต่าง ๆ (Bug) ในการทดสอบจะใช้โทรศัพท์มือถือจำนวน 4 เครื่อง รายละเอียดดังนี้

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| แบรนด์ | ชื่อรุ่น | ระบบปฏิบัติการ | วันที่วางขาย | RAM | จอภาพ | |
| ขนาด | ความละเอียด |
| Oppo | R7 Lite | Android 5.1 Lollipop | กันยายน 2558 | 2GB | 5 นิ้ว | 720 x 1080 px |
| Samsung | Galaxy J7 Prime | Android 6.0.1 Marshmallow | ตุลาคม 2559 | 3GB | 5.5 นิ้ว | 1080 x 1920 px |
| Dtac | Phone X3 | Android 7.0 Nougat | เมษายน 2560 | 3GB | 5.5 นิ้ว | 1080 x 1920 px |
| Vivo | Y91i | Android 8.1 Oreo (Funtouch OS 4.5) | ธันวาคม 2561 | 2GB | 6.22 นิ้ว | 720 x 1520 px |

ตารางที่ 3.1 รายการโทรศัพท์มือถือที่ใช้ในการทดสอบ และข้อมูลที่สำคัญ

**3.5 การนำไปใช้**

คณะผู้จัดทำได้อัปโหลดแอปพลิเคชันที่ได้พัฒนาแล้วไปยังระบบ GitHub ซึ่งเป็นเว็บไซต์สำหรับเก็บรักษาและส่งต่อโค้ดต่าง ๆ แบบสาธารณะได้โดยไม่มีค่าใช้จ่าย ในวันที่ 10 พฤษภาคม พ.ศ. 2562 โดยทุกท่านสามารถดาวน์โหลดไฟล์แอปพลิเคชันได้ที่ https://github.com/pongtaewin/teacher-finder/releases

**3.6 การดูแลรักษา**

ในการดูแลรักษาแอปพลิเคชัน จะทำการสำรวจความพึงพอใจในแอปพลิเคชันเพื่อเป็นการยืนยันและ

าวน์โหลดไฟล์แอปพลิเคชันได้ที่ https://github.com/pongtaewin/teacher-finder/releases

**3.7 เครื่องมือที่ใช้ในการวิจัย**

การวิจัยครั้งนี้เป็นการวิจัยเชิงสำรวจ (Servey Research) ความคิดเห็นเกี่ยวกับผลิตภัณฑ์โคมไฟเครื่องแขวนคริสตัล มีวิธีการสร้างเครื่องมือดังนี้

1. ศึกษาค้นคว้าเอกสารต่าง ๆ ที่เกี่ยวข้อง

2. กำหนดข้อคำถามที่เกี่ยวข้องกับผลิตภัณฑ์

3. สร้างเครื่องมือเป็นแบบสอบถามที่ใช้ในการเก็บรวบรวมข้อมูล เป็นแบบมาตราส่วนประมาณค่า(Rating Scale) 5 ระดับโดยแบ่งเป็น 2 ตอนคือ

ตอนที่ 1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม ได้แก่ข้อมูลเกี่ยวกับเพศ อายุ และสถานภาพ

ตอนที่ 2 แบบสอบถามเกี่ยวกับผลิตภัณฑ์โคมไฟเครื่องแขวนคริสตัล โดยคณะผู้จัดทำได้สร้างแบบสอบถามเกี่ยวกับผลิตภัณฑ์โคมไฟเครื่องแขวนคริสตัล มี 4 ด้าน ด้านความคิดสร้างสรรค์ ด้านความคงทน ด้านความประณีตและสวยงาม และด้านการนำไปใช้ จำนวน 14 ข้อ เป็นแบบมาตรฐานส่วนประมาณค่า( (Likert, 1932)) 5 ระดับ ความคิดเห็น คือ มากที่สุด มาก ปานกลาง พอใช้ และปรับปรุง โดยคำถามจะมีค่าคะแนนตั้งแต่ 5 4 3 2 และ1 ตามลำดับ

4. นำเครื่องมือไปให้ผู้ทรงคุณวุฒิตรวจสอบเครื่องมือ เพื่อกลับมาปรับปรุงแก้ไขตามคำแนะนำ

5.ปรับปรุงแก้ไขเครื่องมือ เพื่อนำไปใช้เก็บข้อมูลต่อไป

**3.2 ประชากรและกลุ่มตัวอย่าง**

2.1 ประชากร (Population) ที่ใช้ในการศึกษาครั้งนี้เป็นบุคลาการของวิทยาลัยอาชีวศึกษานครราชสีมา จำนวน 3,675 คน

2.2 กลุ่มตัวอย่าง ((Sample) เป็นนักเรียน นักศึกษา ครู เจ้าหน้าที่และนักการภารโรงที่เลือกมาเป็นกลุ่มตัวอย่าง เลือกโดยการสุ่มอย่างง่าย (Simple Random Sampling) ตามตารางของเคอร์ซีและมอแกนท์ (Kerjcle and Morgan) จำนวน 100 คน

**3.3 การดำเนินการทดลอง**

3.3.1ออกแบบผลิตภัณฑ์ที่ประดิษฐ์

3.3.2 จัดเตรียมวัสดุและอุปกรณ์ในการทำ

3.3.3 ขั้นตอนการทำผลิตภัณฑ์

3.3.4 นำผลงานไปใช้ผู้เชี่ยวชาญประเมินและแก้ไขผลิตภัณฑ์

3.3.5 นำผลงานสำเร็จไปเก็บแบบสอบถามเพื่อประเมินผลตามวัตถุประสงค์

**3.4 การเก็บรวบรวมข้อมูล**

ขั้นตอนการเก็บรวบรวมข้อมูลมีดังนี้

3.4.1 การเก็บรวบรวมข้อมูลโดยทำหนังสือขอความอนุเคราะห์เก็บข้อมูลต่อผู้อำนวยการวิทยาลัยอาชีวศึกษานครราชสีมา

3.4.2 แจกแบบสอบถามด้วยตนเองจำนวน 100 คน แล้วเก็บข้อมูลกลับคืนทันที

3.4.3 นำข้อมูลที่ได้มาอย่างครบถ้วนและสมบูรณ์มาวิเคราะห์ข้อมูล โดยใช้โปรแกรม Excel เพื่อหาค่าเฉลี่ยเลขคณิต(AVERAGE) และส่วนเบี่ยงเบนมาตรฐาน(Standard Deviation)

**3.5 วิเคราะห์และสรุปผล**

การวิเคราะห์ข้อมูลสถิตที่ใช้ในการวิเคราะห์ข้อมูล เป็นค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานโดยใช้โปรแกรม Excel แล้วเขียนพรรณนาหรือบรรยายเพื่อแปลผลของข้อมูลที่หาได้จาก เกณฑ์การประเมินความพึงพอใจของผู้ตอบแบบสอบถาม ดังนี้ (สิทธิ์ ธีรสรณ์, 2550: 141)

4.51-5.00 หมายถึง พึงพอใจมากที่สุด

3.51-4.50 หมายถึง พึงพอใจมาก

2.51-3.50 หมายถึง พึงพอใจปานกลาง

1.51-2.50 หมายถึง พึงพอใจพอใช้

1.00-1.50 หมายถึง พึงพอใจปรับปรุง

**บทที่ 4  
ผลการวิจัย**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ข้อ** | **คำถาม** | **จำนวนการให้คะแนน** | | | | | **อัตราความพึงพอใจ** |
| **1** | **2** | **3** | **4** | **5** |
| 1 | ปริมาณข้อมูลในการกรอกข้อมูลคาบเรียน | 0 | 0 | 1 | 16 | 12 | 84% |
| 2 | ปริมาณข้อมูลในการกรอกข้อมูลสถานที่สอน | 0 | 0 | 3 | 15 | 11 | 82% |
| 3 | การแสดงผลข้อมูลตารางเรียน | 0 | 0 | 3 | 14 | 12 | 83% |
| 4 | การแสดงผลข้อมูลสถานที่สอน | 0 | 0 | 6 | 12 | 11 | 79% |
| 5 | การแก้ไขข้อมูลคาบเรียนและตารางเรียน | 0 | 0 | 8 | 9 | 12 | 78% |
| 6 | การแก้ไขข้อมูลสถานที่สอนของอาจารย์ | 0 | 0 | 9 | 11 | 9 | 75% |
| 7 | การจัดการกับรายชื่ออาจารย์ | 0 | 0 | 6 | 13 | 10 | 78% |
| 8 | คำสั่ง [ส่งออกข้อมูล] | 0 | 0 | 7 | 12 | 10 | 78% |
| 9 | คำสั่ง [นำเข้าข้อมูล] | 0 | 0 | 7 | 13 | 9 | 77% |
| 10 | คำสั่ง [ล้างข้อมูล] | 0 | 0 | 4 | 16 | 9 | 79% |
| **ความพึงพอใจในส่วนที่ 1 โดยเฉลี่ย** | | | | | | | **79%** |

ตารางที่ 4.1 ผลการสอบถามความพึงพอใจจากคำถาม  
ในส่วนที่ 1 (หลักการภายในแอปพลิเคชัน) ของแบบสอบถาม

แผนภูมิที่ 4.1 แผนภูมิแท่งแสดงอัตราความพึงพอใจ  
ในส่วนที่ 1 (หลักการภายในแอปพลิเคชัน) ของแบบสอบถาม

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ข้อ** | **คำถาม** | **จำนวนการให้คะแนน** | | | | | **อัตราความพึงพอใจ** |
| **1** | **2** | **3** | **4** | **5** |
| 11 | การอ่านข้อมูลภายในแอปพลิเคชัน | 0 | 0 | 2 | 9 | 18 | 89% |
| 12 | การเพิ่มและการแก้ไขข้อมูลภายในแอปพลิเคชัน | 0 | 1 | 6 | 9 | 13 | 79% |
| 13 | การโอนย้ายข้อมูลระหว่างแอปพลิเคชัน | 0 | 0 | 6 | 11 | 12 | 80% |
| 14 | ความครบถ้วนของข้อมูลที่ใช้แสดงผลภายในแอปพลิเคชัน | 0 | 0 | 6 | 12 | 11 | 79% |
| 15 | ความรวดเร็วในการเรียกใช้แอปพลิเคชัน | 0 | 0 | 5 | 8 | 16 | 84% |
| 16 | การควบคุมแอปพลิเคชัน | 0 | 0 | 6 | 10 | 13 | 81% |
| 17 | ความเสถียรภายในแอปพลิเคชัน | 0 | 1 | 5 | 13 | 10 | 78% |
| 18 | การออกแบบภายในแอปพลิเคชัน | 0 | 0 | 3 | 11 | 15 | 85% |
| 19 | ขนาดของตัวอักษรภายในแอปพลิเคชัน | 0 | 0 | 3 | 12 | 14 | 84% |
| 20 | ความพึงพอใจแอปพลิเคชันในภาพรวม | 0 | 0 | 2 | 16 | 11 | 83% |
| **ความพึงพอใจในส่วนที่ 2 โดยเฉลี่ย** | | | | | | | **82%** |

ตารางที่ 4.2 ผลการสอบถามความพึงพอใจจากคำถาม  
ในส่วนที่ 2 (การออกแบบแอปพลิเคชัน) ของแบบสอบถาม

แผนภูมิที่ 4.2 แผนภูมิแท่งแสดงอัตราความพึงพอใจ  
ในส่วนที่ 2 (การออกแบบแอปพลิเคชัน) ของแบบสอบถาม

บรรณานุกรม

# บรรณานุกรม

Andrew Powell-Morse. (8 December 2016). *Waterfall Model: What Is It and When Should You Use It?* เข้าถึงได้จาก https://airbrake.io/blog/sdlc/waterfall-model

Android Developers. (2019, April 26). *Platform Architecture*. Retrieved from https://developer.android.com/guide/platform

ConceptDraw. (2019). *Flowchart Components*. Retrieved April 26, 2019, from https://www.conceptdraw.com/How-To-Guide/diagram-software-flowchart-component

Friedland, G. (2019, April 26). *Lecture #09: Object-Oriented Programming.* Retrieved from https://cs88-website.github.io/assets/slides/08-Object\_Oriented\_Programming\_1pp.pdf

Jeffrey S. Carroll. (26 April 2019). *How Java Works*. เข้าถึงได้จาก http://www.cs.cmu.edu/~jcarroll/15-100-s05/supps/basics/history.html

Krishan Kumar. (26 April 2019). *Java Primitive Data Types*. เข้าถึงได้จาก http://cs-fundamentals.com/java-programming/java-primitive-data-types.php

Oracle. (2019, April 26). *The Java Language Environment*. Retrieved from https://www.oracle.com/technetwork/java/intro-141325.html

Rensis Likert. (1932). A technique for the measurement of altitudes. *Archive of Psychology, 22*(140), 5-55.

Stefan, R. (2003, July 23). *Dr. Alan Kay on the meaning of “object-oriented programming”*. Retrieved April 26, 2019, from http://www.purl.org/stefan\_ram/pub/doc\_kay\_oop\_de

The Guardian. (2009, April 20). *Oracle's takeover of Sun Microsystems comes as surprise to software industry.* Retrieved from https://www.theguardian.com/business/2009/apr/20/sun-microsystems-oracle-takeover

TutorialsPoint. (26 April 2019). *SDLC - Waterfall Model*. เข้าถึงได้จาก https://www.tutorialspoint.com/sdlc/sdlc\_waterfall\_model.htm

ภาชญา เชี่ยวชาญ. (26 เมษายน 2562). *ความรู้เบื้องต้นเกี่ยวโครงสร้างข้อมูล*. เข้าถึงได้จาก http://www.elfhs.ssru.ac.th/wipada\_ch/pluginfile.php/893/course/summary/Chapter%201.pdf

สุพรรษา ยวงทอง. (2557). *ความรู้เบื้องต้นเกี่ยวกับคอมพิวเตอร์เเละเทคโนโลยีสารสนเทศ.* กรุงเทพฯ: โปรวิชั่น.

อานนท์ หลงหัน. (25 ตุลาคม 2556). *การเขียนโปรแกรมเชิงวัตถุ OOP (Object Oriented Programming)*. เรียกใช้เมื่อ 26 เมษายน 2562 จาก https://arit.rmutsv.ac.th/th/blogs/80-การเขียนโปรแกรมเชิงวัตถุ-oop-object-oriented-programming-537