

# บทที่ 1

## บทนำ

### 1.1 ที่มาและความสำคัญ

โรงเรียนสวนกุหลาบวิทยาลัย เป็นสถานศึกษาขนาดใหญ่พิเศษ มีห้องเรียน 79 ห้อง และมีนักเรียนประมาณ 3,200 คน การศึกษาในปัจจุบันของนักเรียนโรงเรียนสวนกุหลาบวิทยาลัย มีทั้งการเรียนปกติ กิจกรรมของโรงเรียน กิจกรรมชุมนุม และอื่น ๆ อีกมากมาย ทำให้การจัดสรรเวลาในแต่ละวันมีความจำกัด และมีความสำคัญ

จากประสบการณ์ของผู้วิจัยพบว่า ปริมาณและเวลาของกิจกรรมและการเรียนอาจส่งผลให้นักเรียนส่งงานไม่ทันตามกำหนดเวลา จึงจำเป็นต้องมีการส่งงานนอกเหนือจากวันเวลาที่อาจารย์กำหนด ซึ่งมีความยุ่งยากในการตามหาสถานที่ที่อาจารย์อยู่ในขณะนั้น เนื่องจากอาจารย์อาจกำลังสอนอยู่ในห้องเรียนหนึ่ง เป็นต้น

ปัจจุบันนักเรียนโรงเรียนสวนกุหลาบวิทยาลัยมีความนิยมและให้ความสำคัญกับเทคโนโลยี และมีการใช้งานโทรศัพท์เคลื่อนที่ในชีวิตประจำวันกันมากขึ้น และปัจจุบันก็มีการออกแบบแอปพลิเคชันเพื่อให้รองรับการใช้งานประเภทต่างๆ เมื่อเป็นดังนี้จึงได้นำเทคโนโลยีต่างๆ กับพฤติกรรมของนักเรียน ความจำเป็นจากปัญหาข้างต้น มาประยุกต์ใช้ให้เกิดประโยชน์ร่วมกัน โดยเขียนแอปพลิเคชัน “Teacher Finder” แอปพลิเคชันแสดงตารางเรียนและสถานที่สอนของอาจารย์มาใช้งานเพื่อให้นักเรียนได้ดาวน์โหลดไปใช้สำหรับอำนวยความสะดวกให้กับตนเองควบคู่กับการใช้งานโทรศัพท์มือถือในชีวิตประจำวันเพื่อให้สัมฤทธิ์ผลทางการเรียนได้มากยิ่งขึ้นต่อไป

### 1.2 วัตถุประสงค์

1. เพื่อแสดงผลข้อมูลของตารางเรียนประกอบด้วย ชื่อวิชา รหัสวิชา รายชื่ออาจารย์ผู้สอน ห้องเรียน และเวลาที่สอนในแต่ละคาบในวันที่มีเรียนในห้อง ม. 502 ผ่านแอปพลิเคชันบนโทรศัพท์เคลื่อนที่ของระบบปฏิบัติการ Android
2. เพื่อแสดงผลข้อมูลสถานที่สอนในแต่ละคาบในวันที่มีเรียนของอาจารย์ที่มีรายชื่อสอนในห้อง ม. 502 ทั้งหมดผ่านแอปพลิเคชันบนโทรศัพท์เคลื่อนที่ของระบบปฏิบัติการ Android
3. เพื่อแสดงผลข้อมูลสถานะการสอนของอาจารย์ผู้สอนที่มีรายชื่อสอนในห้อง ม. 502 ทั้งหมด ในวันที่มีเรียนผ่านแอปพลิเคชันบนโทรศัพท์เคลื่อนที่ของระบบปฏิบัติการ Android

### 1.3 สมมติฐาน

1. นักเรียนห้อง ม. 502 สามารถดูข้อมูลชื่อวิชา รหัสวิชา รายชื่ออาจารย์ผู้สอน ห้องเรียน และเวลาที่สอนในตารางเรียนแต่ละคาบในวันที่มีเรียน จากแอปพลิเคชันบนโทรศัพท์เคลื่อนที่ของระบบปฏิบัติการ Android ได้อย่างสะดวกและรวดเร็ว
2. นักเรียนห้อง ม.502 สามารถดูสถานที่ที่อาจารย์ผู้สอนสอนในห้อง ม. 502 ไปสอนแต่ละคาบในวันที่มีเรียน ผ่านแอปพลิเคชันบนโทรศัพท์เคลื่อนที่ของระบบปฏิบัติการ Android ได้อย่างสะดวกรวดเร็ว
3. นักเรียนห้อง ม.502 สามารถดูสถานะการสอนของอาจารย์ผู้สอนที่มีรายชื่อสอนในห้อง ม. 502 ทั้งหมด และสามารถติดตามไปส่งงาน หรือประสานงานในกิจกรรมที่เกี่ยวข้องในสถานที่สอนนั้น ๆ ในวันที่มีเรียน ผ่านแอปพลิเคชันบนโทรศัพท์เคลื่อนที่ของระบบปฏิบัติการ Android ได้อย่างสมฤทธิ์ผล

### 1.4 ขอบเขตของการศึกษา

1. แอปพลิเคชันจะแสดงข้อมูลรายละเอียดที่เกี่ยวข้องในตารางเรียนห้อง ม. 502
2. แอปพลิเคชันจะแสดงข้อมูลสถานที่สอนของอาจารย์ผู้สอนรายชื่อสอนแต่ละคาบในตารางเรียนห้อง ม. 502
3. แอปพลิเคชันจะแสดงข้อมูลสถานะการสอนของอาจารย์ผู้สอนรายชื่อสอนแต่ละคาบในตารางเรียนห้อง ม. 502
4. แอปพลิเคชันทำงานบนโทรศัพท์เคลื่อนที่ของระบบปฏิบัติการ Android

### 1.5 ตัวแปรที่ศึกษา

- |              |  |
|--------------|--|
| ตัวแปรต้น    | ข้อมูลตารางเรียน ชื่อวิชา รหัสวิชา รายชื่ออาจารย์ผู้สอน เวลาที่สอน สถานที่สอน และสถานะการสอนแต่ละคาบในวันที่มีเรียนที่ได้จากโรงเรียนสวนกุหลาบวิทยาลัย                        |
| ตัวแปรตาม    | ข้อมูลตารางเรียน ชื่อวิชา รหัสวิชา รายชื่ออาจารย์ผู้สอน เวลาที่สอน สถานที่สอน และสถานะการสอนแต่ละคาบในวันที่มีเรียนที่ได้จากโรงเรียนสวนกุหลาบวิทยาลัย ที่แสดงผลบนแอปพลิเคชัน |
| ตัวแปรควบคุม | ตารางเรียนและข้อมูลสถานที่สอนและสถานะการสอนของอาจารย์ผู้สอนรายชื่อสอนของห้อง ม. 502 ที่ได้จากโรงเรียนสวนกุหลาบวิทยาลัย   |

## 1.6 นิยามเชิงปฏิบัติการ

1. ตารางเรียน คือ ตารางข้อมูลอธิบายกำหนดการเรียนการสอน โดยมีการแบ่งเป็นวันที่ ตามวันที่มีเรียน และคาบเรียนต่าง ๆ โดยโรงเรียนสวนกุหลาบวิทยาลัยมีคาบเรียนจำนวน 10 คาบ ยาวคาบละ 50 นาที คาบที่ 1 เริ่มเวลา 8:30 น. คาบที่ 2 จะเริ่มต่อจากคาบที่ 1 จนถึงคาบที่ 10 สิ้นสุดเวลา 16:50 น.
2. วันที่มีเรียน คือ วันที่โรงเรียนกำหนดให้มีการเรียนการสอน ได้แก่ วันจันทร์ วันอังคาร วันพุธ วันพฤหัสบดี และวันศุกร์ ของทุกสัปดาห์ ยกเว้นวันที่โรงเรียนมีประกาศให้หยุดเรียน หรือในขณะปิดภาคเรียน
3. คาบเรียน คือการแบ่งเวลาเรียนภายในวันที่มีเรียน ออกเป็นส่วนต่าง ๆ เพื่อให้การจัดสรรเวลาเรียนในแต่ละวิชามีความชัดเจน และเป็นระเบียบ ภายในคาบเรียนจะระบุ ชื่อวิชา รหัสวิชา ห้องที่ใช้ในการเรียน ห้องเรียนที่เรียน เป็นต้น
4. การตัดคาบเรียน คือ การยกเลิกการสอนในคาบเรียนหนึ่ง เป็นส่วนหนึ่งของคำสั่งของโรงเรียนซึ่งสามารถเรียกใช้ได้ในวันที่โรงเรียนมีกิจกรรมพิเศษ โดยมีสาเหตุให้นักเรียนได้พักจากการเรียน และได้เข้าร่วมกิจกรรมพิเศษนั้น ๆ ตามเวลากำหนดการในหลักสูตร ซึ่งการตัดคาบเรียนส่วนใหญ่จะตัดเฉพาะท้ายคาบ เช่น ตัดคาบเรียนตั้งแต่คาบเรียนที่ 8-10 และมักใช้ควบคู่กับการร่นคาบเรียนในวันนั้น ๆ ด้วย
5. การร่นคาบเรียน คือ การปรับเวลาของคาบเรียนให้น้อยลงชั่วคราว ให้เหลือคาบละ 40 นาทีภายในวันนั้น ๆ เป็นส่วนหนึ่งของคำสั่งของโรงเรียนซึ่งสามารถเรียกใช้ได้ในวันที่โรงเรียนมีกิจกรรมพิเศษ โดยมีสาเหตุให้นักเรียนได้เรียนในเวลาที่เหลือจากการทำกิจกรรมพิเศษนั้น ๆ โดยคาบที่ 1 จะเริ่มในเวลา 8:30 น. เช่นเดิมแต่คาบที่ 10 จะสิ้นสุดที่เวลา 15:10 น.

## บทที่ 2

### เอกสารและงานวิจัยที่เกี่ยวข้อง

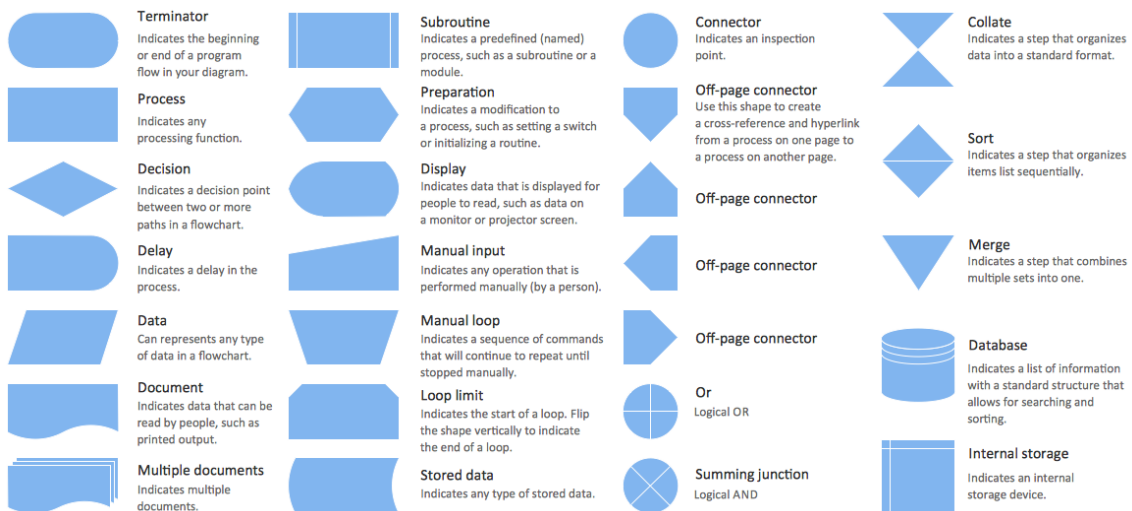
ภายในงานวิจัย มีเอกสารและงานวิจัยที่เกี่ยวข้องดังต่อไปนี้

#### 2.1 เอกสารที่เกี่ยวข้อง

เนื่องจากการเขียนแอปพลิเคชันมีความซับซ้อนยุ่งยากพอสมควร เนื่องด้วยสาเหตุหลายประการ อาทิเช่น ภาษาเขียนโปรแกรมที่ไม่คุ้นเคย หรือระบบปฏิบัติการที่ไม่เหมือนกับในเครื่องคอมพิวเตอร์ จึงทำให้เกิดความจำเป็นที่จะต้องศึกษาความรู้เบื้องต้นก่อนเสมอ ภายในเอกสารจะอธิบายหัวข้อต่าง ๆ ที่ควรศึกษาก่อนการพัฒนาแอปพลิเคชัน ดังนี้

1. ผังงาน
2. การเขียนโปรแกรมเชิงวัตถุ
3. ภาษา Java
4. ระบบปฏิบัติการ Android
5. โครงสร้างข้อมูลเบื้องต้น

##### 2.1.1 ผังงาน



ภาพที่ 2.1 ส่วนประกอบภายในผังงาน

ที่มา: ConceptDraw (2019)

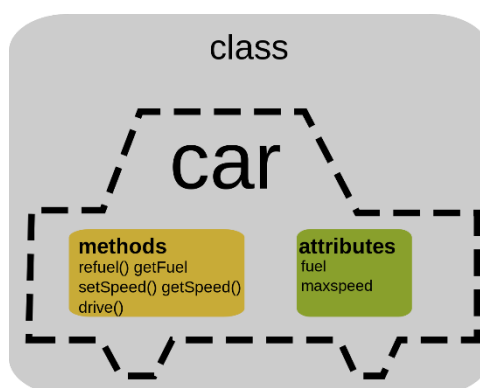
ผังงาน คือเครื่องมือแสดงกระบวนการทำงานผ่านการใช้สัญลักษณ์ที่เป็นมาตรฐานเดียวกัน ภายในสัญลักษณ์จะมีข้อความสั้น ๆ อธิบายการทำงาน ข้อมูล ผลลัพธ์ หรือคำสั่งประมวลผลสำหรับขั้นตอนนั้น ๆ และเชื่อมโยงขั้นตอนเหล่านั้นด้วยเส้นที่มีหัวลูกศรชี้ระบุทิศทางการทำงานตั้งแต่เริ่มต้นจนกระทั่งสิ้นสุด

กระบวนการ (สุพรรณษา ยวงทอง, 2557) การใช้ผังงานเข้ามาช่วยออกแบบโปรแกรมต่าง ๆ เช่น แอปพลิเคชัน จะมีประโยชน์มากในการวางแผนความคิดให้เป็นระบบ และการสร้างความเข้าใจในโปรแกรมที่ออกแบบไว้ด้วย

### 2.1.2 การเขียนโปรแกรมเชิงวัตถุ

อาานนท์ 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” ไม่ว่าจะเป็นสีแดงหรือสีเขียว ขนาดใหญ่หรือเล็ก มีความกว้าง ความยาว ความสูงเท่าใด มันก็คือวัตถุชิ้นหนึ่งเหมือนกัน



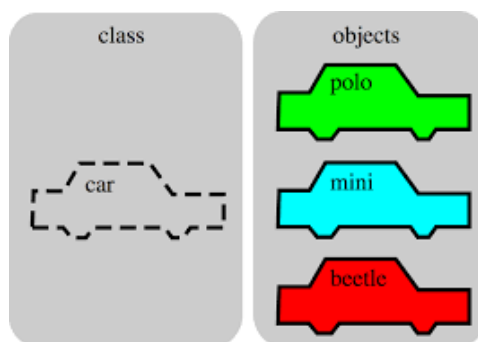
ภาพที่ 2.2 คุณสมบัติ (Attribute) และขั้นตอนวิธี (Method) ของวัตถุ (รถยนต์)

ที่มา: Friedland (2019)

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

ขวดได้” โดยในที่นี้ การสั่งให้เติมน้ำภายในขวดก็เปรียบเสมือนการทำงานของโปรแกรมโดยมีตัวแปรเป็นขวด น้ำพลาสติกนั่นเอง

นอกจากนี้ สามารถกำหนดขอบเขตให้กับคุณสมบัติ (Attribute) และขั้นตอนวิธี (Method) แต่ละอย่างของวัตถุได้ด้วย เช่น หนังสือที่มีคุณสมบัติคือมีข้อความอยู่ภายใน ไม่สามารถอ่านได้ทันที ต้องสั่งให้เปิดหนังสือก่อน จะแก้ไขข้อความก็ไม่สามารถทำได้ทันที ต้องสั่งให้เขียนข้อความภายในหนังสือก่อน เป็นต้น หรือ ตู้เอทีเอ็ม 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 ปีบแตรได้ หลังจากนั้น อาจจะมีวัตถุประเภทรถสปอร์ต ซึ่งสืบทอดมาจากประเภทรถยนต์ สามารถขับได้เช่นกันแต่อาจจะเร็วขึ้น ปีบแตรได้เหมือนกัน เป็นต้น

แต่การสืบทอดวัตถุนั้นก็มีข้อจำกัดคือ สามารถสืบทอดได้เพียงประเภทเดียวเท่านั้น เช่น วัตถุประเภทนก สืบทอดมาจากสัตว์ปีก ส่วนวัตถุประเภทเครื่องบินสืบทอดมาจากเครื่องบิน เป็นต้น ถ้าเราต้องการโปรแกรมที่สามารถสั่งให้ทั้งนกและเครื่องบินออกตัวได้ มันก็จะทำได้ยากถ้าทั้งสองไม่ได้สืบทอดมาจากวัตถุดังกล่าวชนิดเดียวกัน วิธีการแก้ไขคือการให้วัตถุทั้งสองมีส่วนต่อประสานหรือ Interface ซึ่งจะบังคับให้ทั้งสองวัตถุต้องมีวิธีการหรือ Method ของตัว Interface นั้น เพื่อให้การเรียกใช้งานสามารถทำได้ผ่าน Method ของตัว Interface นี้ได้เลย

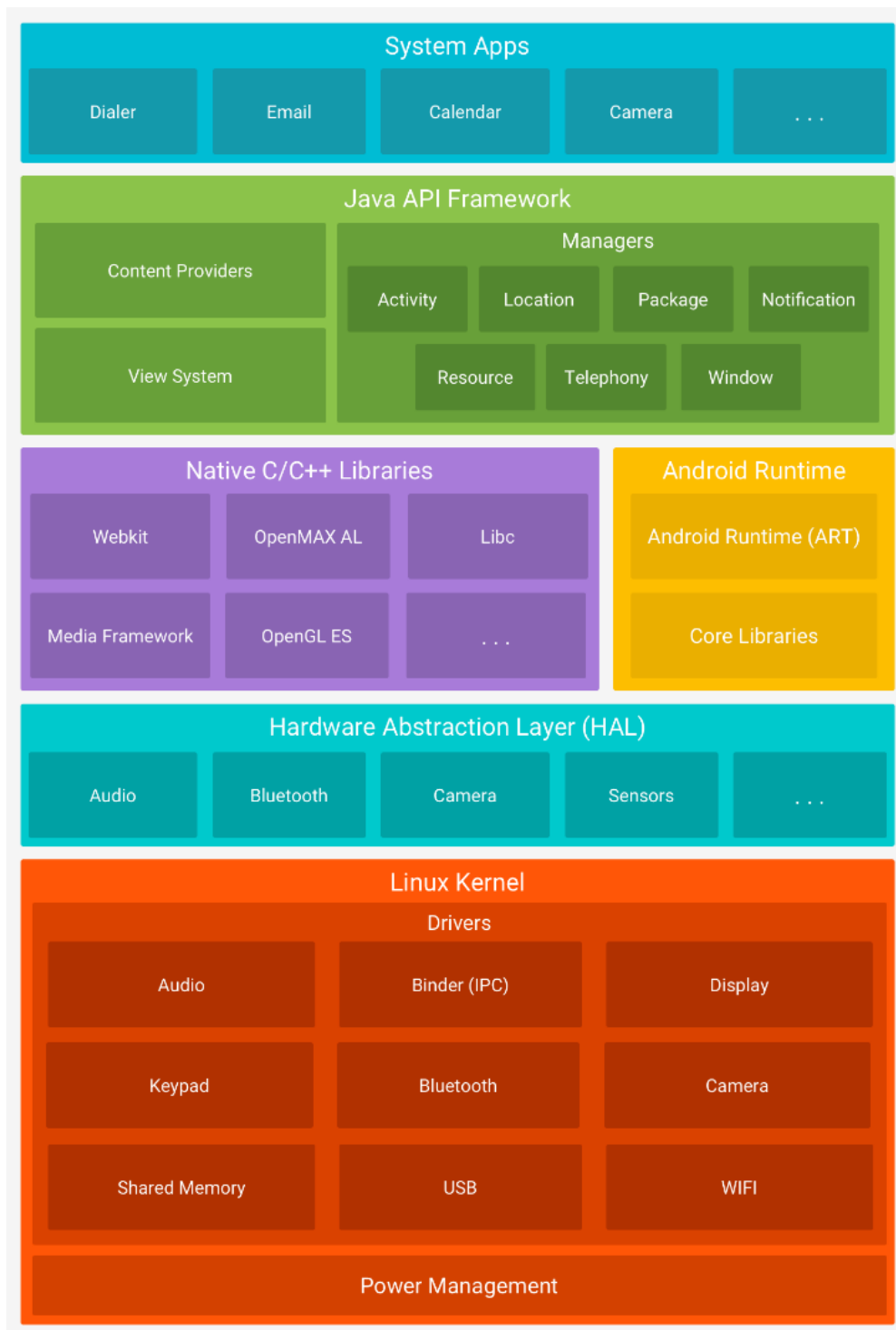
นอกจากนี้ Interface ยังสามารถเรียกใช้งานได้ง่ายและสามารถใช้ได้พร้อมกันหลาย Interface ในวัตถุเดียว เช่น รถยนต์สามารถเร่งเครื่องได้ สามารถเบรกได้ สามารถปิดน้ำฝนได้ สามารถเลี้ยวได้ และสามารถทำได้อีกหลายหน้าที่ ถ้าต้องการจะสั่งให้ทำหน้าที่ใดหน้าที่หนึ่งในหลายวัตถุพร้อมกัน ก็สามารถให้วัตถุเหล่านั้นใช้ Interface ร่วมกันได้ ในกรณีนี้คือ รถยนต์มี Interface สำหรับการเร่งเครื่อง การเบรก การปิดน้ำฝน การเลี้ยว ทำให้สามารถสั่งการร่วมกับวัตถุอื่น เช่น เรือ ที่เร่งเครื่องได้ เลี้ยวได้ แต่วิธีการอาจจะไม่เหมือนกัน เป็นต้น

ประโยชน์ในการเขียนโปรแกรมโดยอาศัยหลักการเชิงวัตถุ นั้น จะอยู่ที่การออกแบบโปรแกรม ด้วยคุณลักษณะของโปรแกรมที่ใช้วัตถุเป็นเครื่องมือในการทำงาน ทำให้การใช้งานโค้ดซ้ำ ๆ มีความง่าย รวดเร็ว และจัดระเบียบได้ง่ายในฐานะของการเป็น “วัตถุ” เมื่อโปรแกรมทำงาน โปรแกรมก็เพียงแค่สั่งระบบให้ทำการเรียกใช้ สร้าง แก้ไข ลบ ดัดแปลงวัตถุต่าง ๆ ตามลักษณะการใช้งานที่โปรแกรมออกแบบไว้ ทำให้การเขียนโปรแกรมเชิงวัตถุ เป็นหลักการที่จำเป็นต่อการพัฒนาโปรแกรมสำหรับการใช้ในโปรแกรมสำหรับผู้ใช้งานโดยเสมอ

### 2.1.3 ภาษา Java

ภาษา Java เป็นภาษาเขียนโปรแกรมที่พัฒนาโดยใช้หลักการเขียนโปรแกรมเชิงวัตถุ 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 ถูกพัฒนาโดยใช้ภาษา Java เป็นหลัก ทำให้ภาษา Java เป็นพื้นฐานความรู้ที่จำเป็นในการศึกษาก่อนเริ่มการเขียนแอปพลิเคชันเสมอ

### 2.1.4 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 ซึ่งเป็นเคอร์เนลของระบบปฏิบัติการ Linux การนำ Linux Kernel มาใช้ในระบบปฏิบัติการ Android ทำให้ระบบปฏิบัติการมีความเสถียรและความปลอดภัยเทียบเท่ากับระบบปฏิบัติการ Linux ที่ได้รับการดูแลรักษาเป็นอย่างดี จึงมั่นใจได้ในด้านเสถียรภาพและความปลอดภัยของ Android

2. Hardware Abstraction Layer คือส่วนที่ทำการมอดการใช้งาน Hardware ต่าง ๆ เช่น กล้อง ลำโพง หรือเซนเซอร์ต่าง ๆ โดยทำการควบคุมและตรวจสอบสิทธิ์ต่าง ๆ ของแอปพลิเคชันก่อนมอดสิทธิ์ทุกครั้ง

3. Android Runtime คือ ส่วนที่คอยจัดการการทำงานของแอปพลิเคชันต่าง ๆ โดยจะทำการแบ่งพื้นที่ให้แอปพลิเคชันทำงานแยกกัน และทำการควบคุมความปลอดภัยไม่ให้แอปพลิเคชันเข้าไปในกระบวนการอื่น ๆ

4. Native C++ Library คือกลุ่มของ Library ในภาษา C++ ที่ใช้ในการพัฒนาแอปพลิเคชันในส่วนเชิงลึก ซึ่งรวมไปถึง Android Runtime ด้วย การใช้งาน Libraries เหล่านี้จึงมีประโยชน์อย่างมากหากต้องการประสิทธิภาพที่สูง เช่น เกมแข่งรถที่ต้องประมวลผลกราฟฟิกในปริมาณมาก เป็นต้น ในกรณีเหล่านี้การนำภาษา C++ มาใช้โดยตรง จะได้ผลดีกว่าการใช้ Java API Framework ที่อาจทำให้ประสิทธิภาพแย่งกว่าเดิม

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

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

### 2.1.5 โครงสร้างข้อมูล

โครงสร้างข้อมูล คือคำนิยามของการจัดการข้อมูลในรูปแบบต่าง ๆ โดยทำการจัดเป็นกลุ่ม เป็นชนิด อย่างมีกฎและรูปแบบ เพื่อให้การเก็บข้อมูล การนำข้อมูลกลับมาใช้ และการประมวลผลทำได้อย่างมีประสิทธิภาพ ซึ่งโดยปกติแล้ว ข้อมูลในภาษาเครื่องจะมีแค่สัญญาณ 0 และ 1 เท่านั้น จึงเป็นเรื่องที่แทบจะเป็นไปไม่ได้เลยถ้าต้องการเขียนโปรแกรมให้ผู้ใช้อ่านได้ ถ้าไม่มีการจัดข้อมูลให้เป็นระเบียบและเป็นมาตรฐาน นอกจากนี้ การที่ไม่จัดรูปแบบข้อมูลนั้น การเขียนโปรแกรมก็จะเป็นสิ่งที่แทบจะเป็นไปไม่ได้เช่นกัน การมีโครงสร้างข้อมูลจึงเป็นสิ่งที่จำเป็นสำหรับภาษาเขียนโปรแกรมส่วนใหญ่ในปัจจุบัน (ภาชญา เขียวชาญ, 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)	$\pm 9.223372 \times 10^{18}$
float	จำนวนจริง	4 B (32 bit)	$\pm 3.402823 \times 10^{38}$
double	(มีทศนิยม)	8 B (64 bit)	$\pm 1.797693 \times 10^{308}$

ภาพที่ 2.5 โครงสร้างข้อมูลประเภทพื้นฐาน ในภาษา Java

ที่มา: Kumar (2019)

2. โครงสร้างข้อมูลประเภทผสม คือ การนำโครงสร้างข้อมูลประเภทต่าง ๆ มาจัดเรียงให้เป็นระเบียบหรือเป็นลักษณะที่กำหนดไว้ เพื่อให้การบันทึกข้อมูลทำได้ง่ายขึ้นและเป็นหมวดหมู่ ยกตัวอย่างเช่น
  1. แถวลำดับ (Array) เป็นการจัดเรียงข้อมูลให้เรียงกันเป็นเส้นตรง โดยส่วนใหญ่ข้อมูลภายในแถวลำดับมักเป็นชนิดเดียวกัน ข้อมูลภายในทุกข้อมูลจะมีจำนวนเต็มกำกับเป็นตำแหน่ง ขนาดของแถวลำดับมักจะมีขนาดคงที่ แต่สามารถปรับให้ขยายขนาดได้ด้วยวิธีต่าง ๆ ตัวอย่างการใช้งาน

เช่น สายอักขระ (String) ที่เป็นแถวลำดับของตัวอักษร ใช้สำหรับการเก็บข้อมูลในโปรแกรมเอกสาร เช่น Microsoft Word เป็นต้น

2. ระเบียบ (Record) เป็นโครงสร้างข้อมูลแบบหนึ่ง ที่ประกอบไปด้วยโครงสร้างข้อมูลอื่น ๆ เรียกว่า เขตข้อมูล ส่วนใหญ่ระเบียบจะมีจำนวนเขตข้อมูลและลำดับที่คงที่ การเข้าถึงข้อมูลภายในจะเข้าถึงได้ผ่านการกำหนดชื่อของเขตข้อมูล ยกตัวอย่างเช่น โครงสร้างข้อมูลชื่อและนามสกุล ประกอบไปด้วย ชื่อที่เป็นสายอักขระ และนามสกุลที่เป็นสายอักขระเช่นกัน เมื่อเรียกใช้งานก็สามารถรับค่าชื่อจากโครงสร้างของชื่อและนามสกุลได้ สามารถแก้ไขได้เช่นกัน เป็นต้น ระเบียบเป็นโครงสร้างข้อมูลอย่างง่ายชนิดหนึ่งที่ใช้กันอย่างกว้างขวาง เนื่องจากมีประโยชน์ในการเขียนโปรแกรมเชิงวัตถุอย่างมาก โดยส่วนใหญ่ การออกแบบวัตถุในภาษาดังกล่าว จะอาศัยหลักการมองว่าคุณสมบัติ (Attribute) ของวัตถุคือระเบียบชนิดหนึ่ง ส่วนขั้นตอนวิธี (Method) ของวัตถุคือคำสั่งในการจัดการระเบียบ เป็นต้น
3. ยูเนียน (Union) เป็นโครงสร้างข้อมูลประเภทหนึ่ง สามารถเก็บค่าได้หลายประเภทตามแต่จะกำหนดไว้ ซึ่งสามารถเปลี่ยนแปลงประเภทที่จะเก็บได้ในเวลาทำงานจริง เช่น ยูเนียนของตัวอักษรและตัวเลข สามารถเก็บตัวเลขก็ได้ เก็บตัวอักษรก็ได้ แต่ไม่สามารถเก็บได้ทั้งตัวเลขและตัวอักษรพร้อมกัน การใช้ยูเนียนจะมีประโยชน์ในด้านพื้นที่ ที่ใช้พื้นที่ในการเก็บข้อมูลน้อยกว่า แต่จะแลกมาด้วยความยุ่งยากในการตรวจสอบประเภทของค่าที่เก็บไว้ จึงมีการพัฒนายูเนียนพร้อมป้ายระบุ (Tagged Union) สำหรับการตรวจสอบว่าในขณะนั้น ยูเนียนเก็บข้อมูลประเภทใดอยู่นั่นเอง
3. โครงสร้างข้อมูลประเภทนามธรรม คือ โครงสร้างข้อมูลที่เป็นที่ที่จะต้องระบุประเภทให้แน่ชัดก่อนใช้งาน เนื่องจากโครงสร้างข้อมูลประเภทนี้ถูกออกแบบให้สามารถเก็บข้อมูลได้จากทุกประเภท และออกแบบมาเพื่อช่วยเหลือการทำงานของโปรแกรม ที่ไม่สามารถทำได้ หรือทำได้แต่ประสิทธิภาพต่ำกว่า ถ้าหากใช้โครงสร้างข้อมูลประเภทผสม ในเอกสารนี้จะยกตัวอย่างเฉพาะแถวลำดับเชื่อมโยงที่ใช้ในงานวิจัยเท่านั้น
1. แถวลำดับเชื่อมโยง (Associative Array) คือแถวลำดับประเภทหนึ่งที่เปลี่ยนจากการใช้จำนวนเต็มในการกำกับตำแหน่ง เป็นข้อมูลประเภทใด ๆ แทน ซึ่งการค้นหาจะเริ่มจากการใส่ค่ากำกับแล้วแถวลำดับเชื่อมโยงจะคืนค่าที่ได้กลับมา หรือคืนค่าว่าง (Null Value) ถ้าไม่มีค่าที่กำกับในแถวลำดับเชื่อมโยงนั้น แถวลำดับเชื่อมโยงนั้นสามารถออกแบบได้หลากหลายวิธี แต่ขอไม่กล่าวถึงในเอกสารนี้เพราะเนื้อหาในการออกแบบและประสิทธิภาพนั้นไม่ใช่สาระสำคัญของการพัฒนาแอปพลิเคชัน

## 2.2 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่ได้ทำการศึกษาเพื่อเทียบกับงานวิจัยนี้ ทั้งในการออกแบบงานวิจัย และการใช้ข้อมูลโปรแกรมเป็นต้นแบบในงานวิจัย มีทั้งสิ้น # งานวิจัยดังนี้

[รายชื่องานวิจัยที่เกี่ยวข้อง จัดเรียงตามหลักการ และความสอดคล้องกับงานวิจัย (มีจำนวนประมาณ 5-10 ผลงาน เกี่ยวข้องกับแอปพลิเคชัน การแสดงผลแอปพลิเคชัน เป็นต้น)]