ระบบควบคุมการนำเสนองาน โดยระบบสัมผัสอัจฉริยะ

UNIVERSAL DROIDMOTE

นาย ปกรณ์ กระจ่าง
นาย พิพัฒน์ บุญทอง

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา ๒๕๕๑

ระบบควบคุมการนำเสนองาน โดยระบบสัมผัสอัจฉริยะ

UNIVERSAL DROIDMOTE

นาย ปกรณ์ กระจ่าง

นาย พิพัฒน์ บุญทอง

โครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2556

UNIVERSAL DROIDMOTE

MR PAKORN KRACHANG

MR PHIPHAT BOONTHONG

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR DEGREE OF BACHELOR OF SCIENCE

IN COMPUTER SCIENCE

FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2013

หัวข้อปัญหาพิเศษ ระบบควบคุมการนำเสนองาน โดยระบบสัมผัสอัจฉริยะ UNIVERSAL DROIDMOTE

ชื่อนักศึกษา นาย ปกรณ์ กระจ่าง 53051014

นาย พิพัฒน์ บุญทอง 53051037

ปริญญา วิทยาศาสตรบัณฑิต

สาขาวิชา วิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษา ดร.สุวรรณ จันทิวาสารกิจ

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้โครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2556

คณะกรรมการสอบ	ลายมือชื่อ
ประธานกรรมการ รศ.ดร.จีรพร วีระพันธุ์	
กรรมการ ผศ.ดร.นวลสวาท หิรัญสกลวงศ์	
กรรมการและอาจารย์ที่ปรึกษา ดร.สุวรรณ จันทิวาสารกิจ	

ลิขสิทธิ์ของคณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ระบบควบคุมการนำเสนองาน โดยระบบสัมผัสอัจฉริยะ

ชื่อนักศึกษา	นาย ปกรณ์	กระจ่าง	53051014
	นาย พิพัฒน์	บุญทอง	53051037
ปริญญา	วิทยาศาสตรบัณฑิต		
สาขาวิชา	วิทยาการคอมพิวเตอร์		
ปีการศึกษา	2556		
อาจารย์ที่ปรึกษา	ดร.สุวรรณ จันทิวาสารกิจ		

บทคัดย่อ

ปัญหาพิเศษเรื่องระบบควบคุมการนำเสนองานโดยระบบอัจฉริยะได้จัดทำขึ้นเพื่อนำเทคโนโลยีมาประยุกต์ใช้ให้เกิดความสะดวกในการนำเสนองาน โดยทำการการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ และพัฒนาแอปพลิเคชันในระบบปฏิบัติการวินโดวส์ เพื่อนำมาควบคุมสำหรับการนำเสนองาน มีการออกแบบส่วนติดต่อผู้ใช้ให้สามารถใช้งานได้ง่าย สามารถปรับแต่งตามประเภทการใช้งานของผู้ใช้ เช่น โหมดการนำเสนอ โหมดสื่อมัลติมีเดีย และโหมดการใช้งานเว็บบราวเซอร์ เป็นต้น

ซึ่งระบบควบคุมการนำเสนองานโดยระบบอัจฉริยะมีความสะดวกในการใช้งาน แตกต่างจากแอปพลิเคชันประเภทเดียวกัน มีการเรียบเรียงฟังก์ชันการใช้งานที่ใช้บ่อยๆ และสามารถนำมาใช้ได้จริง

โดยระบบถูกพัฒนาขึ้นเพื่อใช้งานทั้งฝั่ง เซิร์ฟเวอร์ และไคลเอนต์ โดยใช้เครื่องมีพัฒนาคือ Eclipse IDE และ Netbean IDE ทั้งนี้ได้มีการศึกษาทฤษฎีการออกแบบหน้าจอการวางโครงร่างและปุ่ม ฟังก์ชันที่นำมาใช้งานในประเภทการใช้งานต่างๆ

UNIVERSAL DROIDMOTE

Students	Mr. Pakorn	Krachang	53051014
	Mr. Phiphat	Boonthong	53051037
Degree	Bachelor of Science		
Major	Computer Science		
Academic Year	2013		
Adviser	Suwan Juntiwasarakij, ph.D		

ABSTRACT

This special problem project aimed at developing a universal remote by applying current mobile technology to facilitate presentations and the presenters. The presenter remote was entirely developed using Android technology, and the host application was developed under an umbrella of Microsoft Windows technology. In addition, the remote was designed and crafted according to the human factor principles. Users were able to customize the remote's functionality according to their modes and preferences such as presentation, multimedia, web browsing, etc.

The universal remote developed to provide ease of use. The often-used functions were customizable according user's preferences and well-suited to the individuals. The project was developed with client-server based topology, Eclipse IDE and Netbean IDE were served as tools, and human factor and UI theories were employed to deliver its full functionality.

กิตติกรรมประกาศ

ในการดำเนินการจัดทำโครงงานพิเศษเรื่อง ระบบควบคุมการนำเสนองาน โดยระบบสัมผัสอัจฉริยะ สามารถลุล่วงไปด้วยดี ทางคณะผู้จัดทำขอขอบพระคุณ ดร.สุวรรณ จันทิวาสารกิจ อาจารย์ที่ปรึกษาปัญหาพิเศษ ที่ได้ให้คำปรึกษาและคำแนะนำ แนวทางการทำปัญหาพิเศษและแนวทางในการแก้ไขปัญหา รวมถึงช่วยปรับปรุงคู่มือการทำโครงงานพิเศษ นี้ให้มีความสมบูรณ์มากขึ้น และขอขอบคุณคณะกรรมการสอบปัญหาพิเศษซึ่งประกอบด้วย ผศ.ดร.นวลสวาท หิรัญสกลวงศ์ รศ.ดร.จีรพร วีระพันธุ์ ที่ช่วยตรวจสอบและให้คำแนะนำทำให้โครงงานมีความสมบูรณ์มากยิ่งขึ้น

ขอขอบพระคุณอาจารย์สาขาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ ทุกท่านที่ได้ประสิทธิ์ประสาทความรู้ทั้งในภาคทฤษฎี และภาคปฏิบัติแก่คณะผู้จัดทำมาตลอดเวลา 4 ปี จนกระทั่งโครงงานพิเศษเสร็จสมบูรณ์ได้ด้วยดี

สุดท้ายนี้คณะผู้จัดทำขอขอบพระคุณ บิดา มารดา และครอบครัว ที่ให้การสนับสนุนด้านกำลังใจ และเพื่อนๆของคณะผู้จัดทำ ที่ให้ความช่วยเหลือ ทำให้คณะผู้จัดทำได้สามารถดำเนินงานพัฒนาโครงการพิเศษได้จนสำเร็จ

คณะผู้จัดทำ

มีนาคม 2557

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญรูป	VII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของงานวิจัย	1
1.2 วัตถุประสงค์ของงานวิจัย	1
1.3 ขอบเขตของงานวิจัย	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนในการดำเนินงาน	3
1.6 แผนการดำเนินงาน	5
1.7 อุปกรณ์ที่ใช้ในการทำปัญหาพิเศษ	6
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	
2.1 Remote Procedure Call (RPC)	7
2.1.1 OSI Model	8
2.1.2 กระบวนการทำงานของ Remote Procedure Call	9
2.2 ระบบปฏิบัติการแอนดรอยด์	15
2.2.1 ประวัติความเป็นมา	16
2.2.2 โครงสร้างของระบบปฏิบัติการแอนดรอยด์	17
2.2.3 รุ่นในระบบปฏิบัติการแอนดรอยด์	19
2.3 การออกแบบสำหรับระบบสัมผัส (Designing for Touch)	21
2.3.1 หลักการทั่วไป (Rule of Thumb)	21
2.3.2 หลักการทั่วไปเกี่ยวกับด้านล่างของหน้าจอ	23
2.3.3 เอกลักษณ์ของแอนดรอยด์	23
2.3.4 การออกแบบขนาด	24
2.3.5 การเรียงปุ่ม	25

สารบัญ(ต่อ)

	หน้า
2.4 งานวิจัยที่เกี่ยวข้องกับการออกแบบ User Interface	27
2.4.1 บทนำของงานวิจัย	28
2.4.2 หลักการออกแบบ และตัวต้นแบบ	29
2.4.3 อินเตอร์เฟส	30
2.4.4 ข้อมูลการวัดประสิทธิภาพ	31
2.4.5 บทสรุปเกี่ยวกับความรู้และความสัมพันธ์ที่มี	32
บทที่ 3 การวิเคราะห์และออกแบบระบบ	33
3.1 วิเคราะห์และศึกษาระบบ	33
3.2 การออกแบบระบบ	34
3.2.1 Use Case Diagram ของระบบ	34
3.2.2 Class Diagram	35
3.2.3 Sequence Diagram	37
3.3 การออกแบบพัฒนาระบบชุดคำสั่งในการควบคุม	39
3.3.1 การควบคุมคีย์ต่างๆในคอมพิวเตอร์	40
3.4 โครงร่าง ส่วนของผู้ใช้งานบนโทรศัพท์เคลื่อนที่	41
บทที่ 4 ผลการวิจัยและอภิปรายผล	46
4.1 รายละเอียดของระบบ	46
4.2 ผลการดำเนินงาน	46
บทที่ 5 สรุปผลการพัฒนาระบบและข้อเสนอแนะ	
5.1 บทสรุป	57
5.2 ข้อเสนอแนะ	58
5.3 ข้อจำกัด	58
เอกสารอ้างอิง	59
ภาคผนวก ก การติดตั้งโปรแกรมและการใช้งาน	60
ก.1 การติดตั้ง Eclipse สำหรับเขียนโปรแกรม และพัฒนาแอนดรอยด์แอปพลิเคชัน	61
ภาคผนวก ข การรองรับหน้าจอที่หลากหลาย	81

สารบัญตาราง

ตารางที่	หน้า
1.1 แสดงแผนการทำงาน	5
3.1 แสดง API Package ในภาษาจาวา	39
3.2 แสดง Constructor ในคลาส Robot	40
3.3 แสดง Method ที่มีในคลาส Robot	40

สารบัญรูป

รูปที่	หน้า
2.1 กระบวนการทำงาน Remote Procedure Call	8
2.2 รายละเอียดของ OSI Model	10
2.3 Data ที่จะส่งจะถูกเพิ่ม Header ของแต่ละชั้นเข้าไป เมื่อมีการรับข้อมูลที่ปลายทางแล้ว Header จะถูกถอดออกตามลำดับชั้น	11
2.4 หน้าที่ในแต่ละ Layer ตามมาตรฐาน OSI	11
2.5 การเปรียบเทียบระหว่าง OSI Model กับการสื่อสารของ Internet โดยจะแสดงรูปแบบข้อมูล, Data และอุปกรณ์ที่ทำงานอยู่ในแต่ละ Layer	14
2.6 โครงสร้างของระบบปฏิบัติการแอนดรอยด์	17
2.7 Active area touch	22
2.8 แสดงความแตกต่างระหว่างระบบปฏิบัติการแอนดรอยด์และไอโอเอส	24
2.9 นักพัฒนาพัฒนานาฬิกา Casio	25
2.10 ความแตกต่างในการจัดวาดเครื่องมือ	26
3.1 แสดงภาพรวมการทำงานของระบบ	33
3.2 แสดง Use Case Diagram ของระบบ	34
3.3 แสดงความสัมพันธ์ระหว่าง Class ในโปรแกรมฝั่ง Server	35
3.4 แสดงความสัมพันธ์ระหว่าง Class ในโปรแกรมฝั่ง Client	36
3.5 Sequence Diagram แสดงถึงการเชื่อมต่อผ่านเครือข่ายไร้สาย	37
3.6 แสดงการใช้ฟังก์ชันการทำงานตัวอย่าง คือกดปุ่มซ้าย	38
3.7 หน้าแรก ให้ผู้ใช้เลือกประเภทการเชื่อมต่อ	42
3.8 แสดงการเชื่อมต่อ	42
3.9 แสดงหน้าหลักในการทำงาน	43
3.10 หน้าฟังก์ชันการทำงานในการ นำเสนองาน โดยผู้ใช้ต้องเปิดโปรแกรม Microsoft PowerPoint ขึ้นมาก่อน	44
3.11 หน้าฟังก์ชันการทำงานในการ เล่นสื่อมัลติมีเดีย โดยผู้ใช้ต้องเปิดโปรแกรมเล่นสื่อขึ้นมาก่อน เช่น Window media player	44
3.12 หน้าฟังก์ชันการทำงานในการ เล่นเว็บบราวเซอร์ โดยผู้ใช้ต้องเปิดเว็บบราวเซอร์ขึ้นมาก่อน เช่น Internet Explorer, Google Chrome	44
3.13 แสดงพื้นที่ Active area touch กับการออกแบบ	45
4.1 Java Application บนฝั่งเซิร์ฟเวอร์	47

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.2 หมายเลข IP Address เพื่อใช้ติดต่อระหว่างเซิร์ฟเวอร์และไคลเอนต์	47
4.3 หน้าก่อนเข้าสู่ Main Menu ซึ่งจะแสดงไอคอนประเภทของการเชื่อมต่อ และ ไอคอนเพื่อนำเข้าสู่หน้าการ	48
4.4 หน้าการเชื่อมต่อ โดยผู้ใช้ต้องเปิดแอปพลิเคชัน Universal Remote Server บนเครื่องคอมพิวเตอร์ เพื่อนำ IP Adress มาใส่ในแอปพลิเคชันบน Smart Phone	49
4.5 หน้าให้ผู้ใช้เลือกความถนัดของการใช้มือ โดยมีความถนัดมือซ้ายและขวา เพื่อให้ผู้ที่มีความถนัดแตกต่างกัน สะดวกต่อการใช้งานในการกดปุ่ม	49
4.6 ตัวอย่างการจัดเรียงไอคอนสำหรับผู้ใช้ที่ถนัดมือซ้ายและขวา	50
4.7 หน้า Main Menu ซึ่งมีประเภทการใช้งานให้เลือกด้วยกัน 3 ประเภท ได้แก่ โหมดการนำเสนอ โหมดมัลติมีเดีย และโหมดเว็บบราวเซอร์ ผู้ใช้สามารถเลือกประเภทการใช้งานตามความเหมาะสม	50
4.8 หน้า โหมดการนำเสนอ	51
4.8.1 ไอคอนและฟังก์ชันการใช้งาน	51
4.9 หน้า โหมดมัลติมีเดีย	52
4.9.1 ไอคอนและฟังก์ชันการใช้งาน ในโหมดมัลติมีเดีย	52
4.9.2 ฟังก์ชันในการเพิ่มเสียงลดเสียง	53
4.10 หน้า โหมดเว็บบราวเซอร์	54
4.10.1 ไอคอนและฟังก์ชันการใช้งาน ในเว็บบราวเซอร์	54
4.11 หน้า Information แนะวิธีการใช้งานเบื้องต้น	55
4.12 หน้า การใช้งานเมาส์และคีย์บอร์ด เพื่อควบคุมเมาส์บนหน้าจอคอมพิวเตอร์	55
4.13 หน้า การใช้งานเมาส์และคีย์บอร์ด	56
ก.1 หน้าแสดงผลผล http://www.eclipse.org/downloads/	61
ก.2 หน้าแสดงผลผลการตรวจสอบเครื่องคอมพิวเตอร์ว่าเป็น 32bit หรือ 64bit	62
ก.3 หน้าแสดงผลผล หน้าต่างในการดาวน์โหลด Eclipse	62
ก.4 หน้าแสดงผลผล โหลด eclipse ได้แล้ว Extract ไปเก็บไว้ที่ C:/	63
ก.5 หน้าแสดงผลผล หน้าต่างในการเก็บ Eclipse	64
ก.6 หน้าแสดงผล eclipse Juno v 4.2	64
ก.7 หน้าแสดงผล workspace eclipse Juno v 4.2	65
ก.8 หน้าแสดงผล eclipse Juno v 4.2	65

สารบัญรูป(ต่อ)

รูปที่	หน้า
ก.9 หน้าแสดงผล Android SDK	66
ก.10 หน้าแสดงผล ลิงก์การโหลด Android SDK	66
ก.11 หน้าแสดงผล ไฟล์ที่จะเลือกดาวน์โหลด	67
ก.12 หน้าแสดงผล ที่จัดเก็บ Android SDK	68
ก.13 หน้าแสดงผล ADT Plugin	69
ก.14 หน้าแสดงผล .zip ไฟล์ ADT	70
ก.15 หน้าแสดงผล Eclipse Install New Software	70
ก.16 หน้าแสดงผล Install Software Add Repository	71
ก.17 หน้าแสดงผล ADT-20.0.0	72
ก.18 หน้าแสดงผล การเลือก Developer Tools	73
ก.19 หน้าแสดงผล การยอมรับเงื่อนไขการติดตั้ง	74
ก.20 หน้าแสดงผล Eclipse Juno	76
ก.21 หน้าแสดงผล หน้าหลักการใช้งาน Eclipse Overview	77
ก.22 หน้าแสดงผล การติดตั้ง Platform Android	77
ก.23 หน้าแสดงผล การเลือกการติดตั้ง Platform Android	78
ก.24 หน้าแสดงผล การใช้งาน การพัฒนาแอปพลิเคชันแอนดรอยด์	80
ข.1 Application Check Screen	83
ข.2 ไอคอนในหน้าจอที่แตกต่างกัน	84

บทที่ 1
บทนำ

1.1 แนวคิดและที่มาของปัญหา

ในปัจจุบันเทคโนโลยีของโทรศัพท์เคลื่อนที่เข้ามามีบทบาทในชีวิตประจำวัน และมีการใช้งานกันอย่างแพร่หลายและต่อมาทำให้มีการพัฒนาระบบปฏิบัติการบนโทรศัพท์เคลื่อนที่ เช่น Android, iOS, Windows Phone เป็นต้น ทำให้เป็นที่รู้จักกันในชื่อ โทรศัพท์เคลื่อนที่อัจฉริยะ หรือสมาร์ทโฟน ที่เปรียบเสมือนการมีคอมพิวเตอร์อยู่ในมือผู้ใช้ ซึ่งผู้ใช้สามารถใช้งานแอปพลิเคชันต่างๆ ที่ช่วยอำนวยความสะดวก เช่น การติดตามข้อมูลข่าวสาร การบันทึกข้อมูล การถ่ายภาพ การใช้สื่อ เป็นต้น จะเห็นได้ว่าในโทรศัพท์เคลื่อนที่อัจฉริยะสามารถใช้งานได้หลากหลาย และความหลายหลายของแอปพลิเคชัน ทำให้มีการแบ่งแยกหมวดหมู่ ซึ่งการพัฒนาแอปพลิเคชัน บางแอปพลิเคชัน ไม่ได้มุ่งเน้นไปถึง การใช้ความสามารถของสมาร์ทโฟนให้เกิดประโยชน์สูงสุด เช่น การพัฒนาแอปพลิเคชันที่สามารถใช้ร่วมกับการนำเสนอ เป็นต้น จะเห็นได้ว่าปัจจุบันการนำเสนอสื่อในรูปแบบที่หลากหลายต้องใช้คอมพิวเตอร์กับโปรแกรมนำเสนอ โดยผู้นำเสนอจะต้องควบคุมการนำเสนอ โดยการใช้อุปกรณ์ต่อร่วมหรือรีโมทพรีเซนเตอร์ เพื่อช่วยในการนำเสนองานอย่างมีประสิทธิภาพ แต่เนื่องจากอุปกรณ์ต่อร่วมมีราคาสูง

ด้วยเหตุนี้ผู้พัฒนาจึงพัฒนาแอปพลิเคชันบนสมาร์ทโฟนที่สามารถใช้ทรัพยากรที่มีอยู่ให้เกิดประโยชน์ และยังสามารถรองรับผู้ใช้งานในกรณีที่ไม่มีอุปกรณ์ เช่น เมาส์ คีย์บอร์ด พรีเซนเตอร์รีโมท เป็นต้น โดยจัดทำแอปพลิเคชันบนสมาร์ทโฟนที่ใช้ระบบปฏิบัติการแอนดรอยด์ติดต่อกับเครื่องคอมพิวเตอร์ เพื่อควบคุมเครื่องคอมพิวเตอร์ในงานประเภทต่างๆ โดยพัฒนาโปรแกรมนี้โดยใช้ภาษาจาวา (Java Programming Language) ทั้งนี้ รีโมทพรีเซนเตอร์มีข้อจำกัดการใช้งานจากฟังก์ชันเดิมคือต้องใช้กับโปรแกรมเฉพาะ เช่น Microsoft PowerPoint เท่านั้นซึ่งเป็นข้อจำกัดในการใช้งาน แต่ในส่วนของฟังก์ชันการใช้งาน จะครอบคลุมมากกว่าเช่นการเปิดเว็บบราวเซอร์

1.2 วัตถุประสงค์ของโครงงานพิเศษ

เพื่อศึกษาการออกแบบและพัฒนาแอปพลิเคชันบนสมาร์ทโฟนระบบปฏิบัติการแอนดรอยด์ เพื่อนำไปใช้ควบคุมคอมพิวเตอร์ผ่านการเชื่อมต่อแบบไร้สาย และทำการศึกษารูปแบบการทำงานอย่างเป็นระบบ โดยวิเคราะห์และเรียบเรียงลำดับความสำคัญของหน้าที่ในแต่ละฟังก์ชัน จัดทำเป็นแอปพลิเคชัน ที่สามารถนำไปใช้งานได้จริงและรองรับการใช้งานบนสมาร์ทโฟนได้อย่างมีประสิทธิภาพ

1.2.1 พัฒนาโปรแกรมบนระบบปฏิบัติการแอนดรอยด์ เพื่อควบคุมคอมพิวเตอร์ผ่านการเชื่อมต่อไร้สาย

1.2.2 พัฒนาการออกแบบ Graphic User Interface ในส่วนติดต่อกับผู้ใช้ ให้มีการใช้งานที่ง่ายและเหมาะสม

1.2.3 ฝึกฝนการพัฒนาโปรแกรมโดยการนำเทคโนโลยีที่มีอยู่มาใช้และพัฒนาการใช้ภาษาจาวา

1.3 ขอบเขตของโครงงานพิเศษ

1.3.1 ศึกษาเกี่ยวกับ การสร้างแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์

1.3.2 ศึกษาพฤติกรรมการใช้งานของผู้ใช้ที่ใช้รีโมทพรีเซนเตอร์

1.3.3 ศึกษาฟังก์ชันการใช้งานหลักของแต่ละโปรแกรม เช่น Microsoft Power Point

1.3.4 ศึกษา Algorithm ในการควบคุม หรือ ศึกษาชุดคำสั่งควบคุม

1.3.5 ศึกษา API ที่ใช้ในการเข้าถึงอุปกรณ์ที่เป็นตัวนำเข้า

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 สามารถนำแอปพลิเคชันดังกล่าวไปใช้งานบนสมาร์ทโฟน และ คอมพิวเตอร์ได้จริง

1.4.2 สามารถทำงานได้หลากหลายรูปแบบ (เช่น ควบคุมเดสก์ท็อปแทนเมาส์ เป็นต้น)

1.4.3 เพื่อให้เห็นถึงแนวโน้มของประโยชน์ใช้สอยได้ง่ายและสะดวกมากยิ่งขึ้น

1.4.4 เพื่อนำไปใช้งานพรีเซนได้อย่างมีประสิทธิภาพ

1.4.5 เข้าใจหลักการออกแบบ Interface ให้เหมาะสมและใช้งานง่าย

1.4.6 มีความรู้เรื่องการสร้างแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์

1.5 ขั้นตอนการดำเนินงาน

ในการพัฒนาแอปพลิเคชันนั้นอาศัยวงจรการพัฒนาซอฟต์แวร์ หรือที่เรียกว่า Software Development Life Cycle (SDLC) โดยวิธีการที่นิยมใช้คือ โครงสร้างแบบน้ำตก (Water Fall Model) ในส่วนของระบบบริหารองค์ความรู้วิสาหกิจชุมชนนั้นก็ได้ใช้หลักการของ SDLC เป็นพื้นฐานในการพัฒนาระบบซึ่งมีขั้นตอนดังนี้

1.5.1 เข้าใจปัญหา (Problem Recognition)

คือ การทำความเข้าใจและหาความต้องการผู้ใช้งานแอปพลิเคชันบนสมาร์ทโฟน ซึ่งปัจจุบันมีการพัฒนาแอปพลิเคชันบนสมาร์ทโฟนมากมาย แต่ปัญหาที่ทำให้แอปพลิเคชันนั้นๆ ไม่น่าสนใจคือ ความสวยงามของแอปพลิเคชัน รวมถึงการใช้สมาร์ทโฟนให้เกิดประโยชน์ในทางเครื่องมือ การใช้งานที่ง่าย จากสาเหตุนี้จึงนำไปสู่การริเริ่มในการสร้างแอปพลิเคชันรีโมท เนื่องจาก รีโมทในปัจจุบัน ที่มีปุ่มกดหลายปุ่ม และมีการจัดเรียงปุ่มกดที่ติดกัน ไม่สวยงาม ทำให้ไม่น่าใช้งาน และปัจจุบันสมาร์ทโฟนเข้ามามีบทบาทในชีวิตประจำวัน จึงสร้างแอปพลิเคชันนี้ โดยอาศัยการออกแบบให้ผู้ใช้สามารถใช้งานได้ง่าย

1.5.2 ศึกษาความเป็นไปได้ (Feasibility Study)

คือ ผู้พัฒนาระบบจะต้องศึกษาถึงความเป็นไปได้ของการพัฒนาแอปพลิเคชัน บนระบบปฏิบัติการแอนดรอยด์ เพื่อจะใช้เวลาในการพัฒนาได้อย่างเหมาะสมและสร้างแอปพลิเคชันออกมาให้ใช้ง่ายได้จริง ทำงานได้อย่างถูกต้อง โดยมีข้อผิดพลาดน้อยที่สุด และศึกษาข้อมูลในการพัฒนา ดังนี้ศึกษาเกี่ยวกับระบบปฏิบัติการแอนดรอยด์ ศึกษา API ที่ใช้ในการเข้าถึงอุปกรณ์ที่เป็นอินพุต ต่างๆ ศึกษาการทำ แอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์ ที่สามารถเชื่อมต่อกับคอมพิวเตอร์

1.5.3 การวิเคราะห์ (Analysis)

คือ การวิเคราะห์ความน่าจะเป็นและอุปสรรคในการพัฒนาระบบ การวิเคราะห์ถึงวิธีการในการเก็บรวบรวมข้อมูลทฤษฎีการออกแบบ บทความ วารสาร วิเคราะห์ระบบเกี่ยวกับฟังก์ชันการใช้งานที่ผู้ใช้ใช้บ่อย วิธีการใช้โทรศัพท์มือถือ ตามหลักการยศาสตร์ แล้วสรุปวิธีการพัฒนาแอปพลิเคชัน สรุปรูปแบบในการพัฒนา และวิเคราะห์การใช้งาน เพื่อหาวิธีการออกแบบแอปพลิเคชันที่ผู้ใช้สามารถใช้งานได้ง่าย

1.5.4 การออกแบบ (Design)

คือ กำหนดความสามารถของแอปพลิเคชัน การเขียนโปรแกรมในภาษาจาวา ออกแบบโปรแกรมตาม ฟังก์ชันการใช้งานที่ใช้บ่อย (Function key) ที่รวบรวมจากการใช้งานรวมถึงการ

กำหนดรูปแบบของหน้าจอ ความหนาแน่นของหน้าจอขนาดที่แตกต่างกัน รูปแบบของไอคอน การจัดวาง และการใช้ฟังก์ชันการใช้งาน หน้าตาของแอปพลิเคชัน สีที่ใช้ เป็นต้น

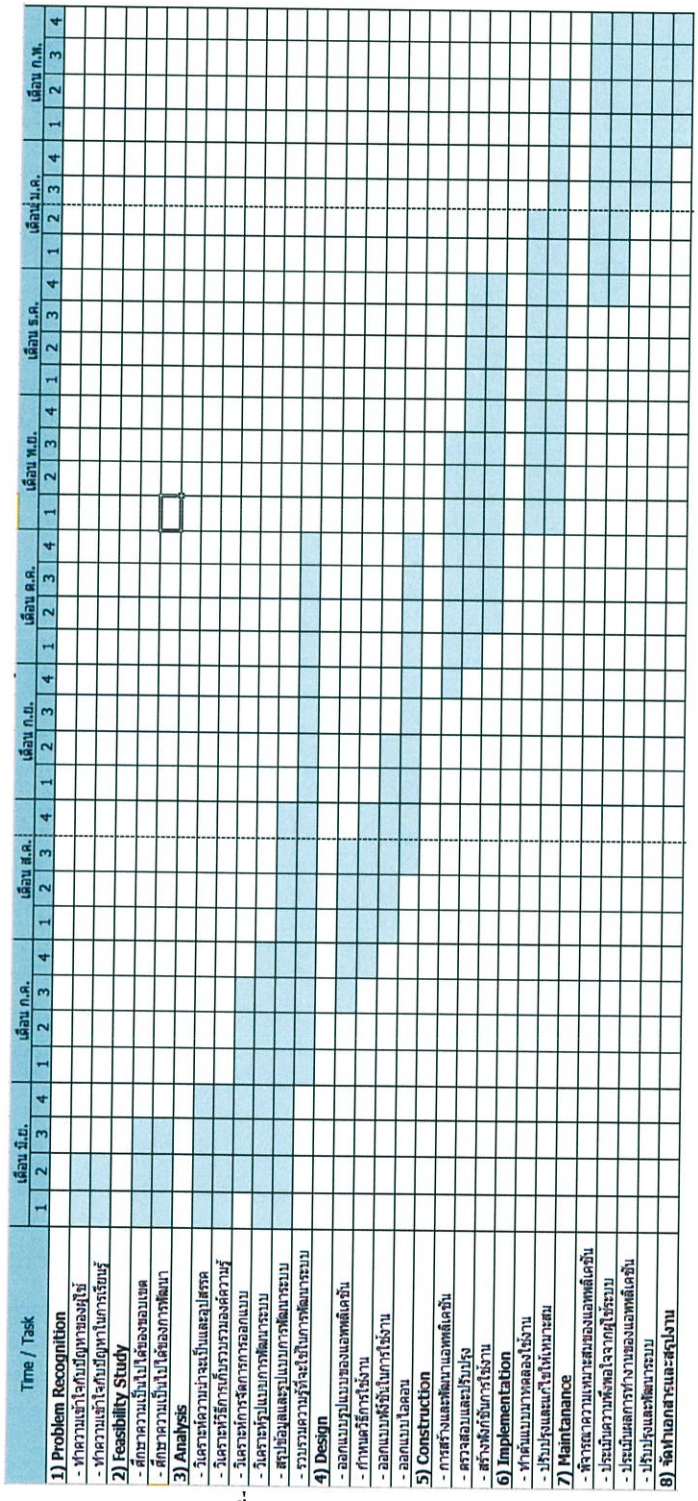
1.5.5 การพัฒนาระบบ (Construction)

คือ การสร้างแอปพลิเคชันโดย Eclipse IDE ให้แอปพลิเคชันสามารถใช้งานได้ง่ายและสะดวก รวดเร็ว โดยมีการใช้การคำนวณการเคลื่อนที่ของไอคอน นอกจากนี้ยังมีการพัฒนาการ การนำไปใช้ (Implementation) คือ การนำแอปพลิเคชันที่ทำสำเร็จไปทดลองใช้งานจริง ตัวอย่างเช่น ใช้ในการนำเสนองาน และการเล่นสื่อมัลติมีเดีย เพื่อศึกษาการทำงานปรับปรุงให้ดียิ่งขึ้น และพัฒนาไปถึงการใช้งานเมาส์ และคีย์บอร์ด เพื่อควบคุมการทำงานบนเครื่องคอมพิวเตอร์

1.5.6 การบำรุงรักษา (Maintenance)

คือ การทดสอบและแก้ไขข้อบกพร่อง ประเมินผลการทำงานของแอปพลิเคชัน ให้สามารถทำงานได้เป็นปกติ และวิเคราะห์ความพึงพอใจของผู้ใช้งานเพื่อสามารถ พัฒนาต่อยอด เพื่อปรับปรุงและพัฒนาระบบต่อไป

1.6 แผนการดำเนินงาน



Time / Task	เดือน มิ.ย.				เดือน ก.ค.				เดือน ส.ค.				เดือน ก.ย.				เดือน ต.ค.				เดือน พ.ย.				เดือน ธ.ค.				เดือน ม.ค.				เดือน ก.พ.			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1) Problem Recognition																																				
- ทำความเข้าใจกับปัญหาของผู้ใช้																																				
- ทำความเข้าใจกับปัญหาในการเรียนรู้																																				
2) Feasibility Study																																				
- ศึกษาความเป็นไปได้ของตลาดเบื้องต้น																																				
- ศึกษาความเป็นไปได้ของการพัฒนา																																				
3) Analysis																																				
- วิเคราะห์ความต้องการเป็นเอกสารสรุป																																				
- วิเคราะห์ฟีเจอร์การเก็บรวบรวมองค์ความรู้																																				
- วิเคราะห์การจัดการออกแบบ																																				
- วิเคราะห์รูปแบบการพัฒนาระบบ																																				
- สรุปข้อมูลและรูปแบบการพัฒนาระบบ																																				
- รวบรวมความรู้ที่จะใช้ในการพัฒนาระบบ																																				
4) Design																																				
- ออกแบบรูปแบบของเอกพลัดเดิน																																				
- กำหนดวิธีการใช้งาน																																				
- ออกแบบหน้าจอโปรแกรมใช้งาน																																				
- ออกแบบโอเดล																																				
5) Construction																																				
- การสร้างและพัฒนาเอกพลัดเดิน																																				
- ตรวจสอบและปรับปรุง																																				
- สร้างหน้าจัดการใช้งาน																																				
6) Implementation																																				
- ทำคัมแบบทดลองใช้งาน																																				
- ปรับปรุงและแก้ไขให้เหมาะสม																																				
7) Maintenance																																				
- ทำการตรวจความเหมาะสมของเอกพลัดเดิน																																				
- ประเมินความพึงพอใจจากผู้ใช้ระบบ																																				
- ประเมินผลความก้าวหน้าของเอกพลัดเดิน																																				
- ปรับปรุงและพัฒนาระบบ																																				
8) จัดทำเอกสารและสรุปงาน																																				

ตารางที่ 1.1 แสดงแผนการทำงาน

1.7 อุปกรณ์ที่ใช้ในการทำปัญหาพิเศษ

1.7.1 รายละเอียดทางด้านอุปกรณ์คอมพิวเตอร์

- เครื่องคอมพิวเตอร์

  - ตัวประมวลผล 1 กิกะเฮิร์ตซ์ (GHz) ขึ้นไป แบบ 32 บิต (x86) หรือ 64 บิต (x64)
  - RAM (32 บิต) 1 กิกะไบต์ (GB) หรือ RAM (64 บิต) 2 GB
  - เนื้อที่ว่างบนฮาร์ดดิสก์ 2 MB (32 บิต)

- สมาร์ทโฟน ที่รองรับระบบปฏิบัติการแอนดรอยด์

  - ระบบปฏิบัติการ : Android™ เวอร์ชัน 2.3.3
  - CPU : Qualcomm® QSD 8250, 1 GHz หน่วยความจำ ROM 512 MB
  - การ์ดหน่วยความจำ micro SD™ สูงสด 32 GB ส่งผ่านข้อมูล (Data Transfer)
  - Wi-Fi 802.11b/g, WLAN (Wireless LAN)

1.7.2 รายละเอียดทางด้านซอฟแวร์

- Eclipse IDE
- Android SDK
- Microsoft PowerPoint
- Web Browser (Internet Explorer , Mozilla Firefox , Google Chrome)
- Media Player Program

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

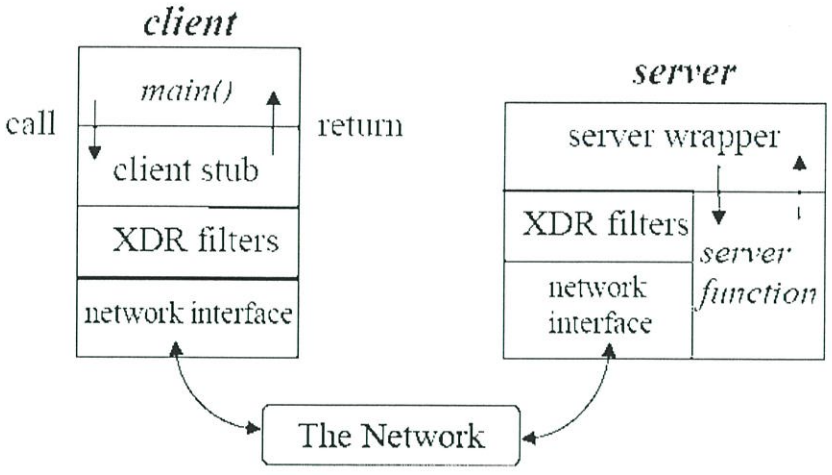
2.1 Remote Procedure Call (RPC) [1]

Remote Procedure Call (RPC) เป็นโพรโทคอลที่โปรแกรมหนึ่งสามารถใช้คำขอบริการจากโปรแกรมที่ตั้งอยู่ในคอมพิวเตอร์อีกเครื่องในเครือโดยไม่ต้องเข้าใจรายละเอียดเครือข่าย (บางครั้ง Procedure Call เรียกว่า Function Call or a Subroutine Call) RPC ใช้แบบจำลองลูกข่าย/แม่ข่าย โปรแกรมที่ขอเป็นลูกข่ายและโปรแกรมที่ให้บริการเป็นแม่ข่าย RPC เป็นปฏิบัติการแบบพร้อมกันที่ต้องการให้โปรแกรมที่ขอได้รับการแขวนจนกระทั่งผลลัพธ์จาก Remote Procedure ได้รับกลับมาเหมือนกับ Procedure Call ธรรมดา การใช้ Lightweight Process หรือ Thread ที่แบ่งปันพื้นที่เดียวกันจะยอมให้หลาย RPC ได้รับการทำอย่างพร้อมกัน

เมื่อประโยคคำสั่งโปรแกรมที่ใช้ RPC ได้รับการคอมไพล์เป็นโปรแกรมแบบ Executable แล้ว Stub (ประโยคคำสั่งส่วนที่ยังเหลืออยู่) ได้รับการรวมในคำสั่งที่คอมไพล์แล้วจะกระทำเป็นตัวแทนของคำสั่ง Remote Procedure เมื่อโปรแกรมประมวลผลและ Procedure Call ได้รับการเรียก Stub รับคำขอและส่งต่อไปยังโปรแกรม runtime ลูกข่ายในเครื่องคอมพิวเตอร์ท้องถิ่น โปรแกรม runtime ลูกข่ายมีความรู้ในการระบุเครื่องคอมพิวเตอร์ทางไกลและการประยุกต์แม่ข่ายและส่งข่าวสารข้ามเครือข่ายที่ขอ Remote Procedure เครื่องแม่ข่ายรวมโปรแกรม Runtime และ Stub ที่อินเตอร์เฟซกับ Remote Procedure เอง ผลลัพธ์ได้รับการส่งกลับทางเดียวกัน RPC มีหลายแบบจำลองและการนำมาใช้ แบบจำลองและการนำมาใช้ที่นิยมมากคือ Distributed Computing Environment (DCE) ของ Open Software Foundation มีการกำหนด RPC เป็นมาตรฐานโดย Institute of Electrical and Electronics Engineers ใน ISO Remote Procedure Call Specification, ISO/IEC CD 11578 N6561, ISO/IEC, November 1991

RPC มีขอบเขตถึง Transport layer และ Application layer ในแบบจำลอง Open Systems Interconnection (OSI) ของการสื่อสารเครือข่าย RPC ทำให้การพัฒนาโปรแกรมประยุกต์ที่รวมหลายโปรแกรมที่กระจายในเครือข่ายง่ายขึ้น วิธีการทางเลือกสำหรับการสื่อสารแบบลูกข่าย/แม่ข่ายรวมถึง message queuing และ Advanced Program-to-Program Communication (APPC) ของ IBM

2.1.1 กระบวนการทำงานของ Remote Procedure Call



รูปที่ 2.1 กระบวนการทำงาน Remote Procedure Call [8]

กระบวนการทำงาน Remote Procedure Call

ขั้นตอนแรก Client จะเรียก Client Stub เรียกขั้นตอนนี้ว่า Local Procedure Call เมื่อถึง Client Stub จะทำการ Marshaling เพื่อทำการส่งผ่าน Network ไปยังปลายทาง คือ Client จะส่ง Message ไปยัง Sever เมื่อ Sever ได้รับ จะทำการ Unmarshaling เพื่อทำการเรียก Function และประมวลผล เพื่อส่งกลับไปยัง Client ดังรูป

2.1.2 OSI MODEL [2][3]

แบบจำลองโอเอสไอ (อังกฤษ: Open Systems Interconnection Model: OSI Model) (ISO/IEC 7498-1) เป็นรูปแบบที่กล่าวถึงคุณสมบัติพิเศษและมาตรฐานการทำงานภายในของระบบการสื่อสาร โดยแบ่งเป็นชั้นนามธรรม และโพรโทคอลของระบบคอมพิวเตอร์ พัฒนาขึ้นโดยองค์กรระหว่างประเทศว่าด้วยการมาตรฐาน (ISO)

แบบจำลองนี้จะทำการจับกลุ่มรูปแบบฟังก์ชันการสื่อสารที่คล้ายกันให้อยู่ในชั้นใดชั้นหนึ่งในเจ็ดชั้นตรรกะ ชั้นใดๆจะให้บริการชั้นที่อยู่บนและตัวเองได้รับบริการจากชั้นที่อยู่ด้านล่าง ตัวอย่างเช่น ชั้นที่ให้การสื่อสารที่ Error-Free ในเครือข่ายจะจัดหาเส้นทางที่จำเป็นสำหรับแอปพลิเคชันชั้นบน ในขณะที่มันเรียกชั้นต่ำลงไปให้ส่งและรับแพ็คเก็ตเพื่อสร้างเนื้อหาของเส้นทางนั้น งานสองอย่างในเวลาเดียวกันที่ชั้นหนึ่งๆจะถูกเชื่อมต่อในแนวนอนบนชั้นนั้นๆ ตามรูปผู้ส่งข้อมูลจะดำเนินงานเริ่มจากชั้นที่ 7 จนถึงชั้นที่ 1 ส่งออกไปข้างนอกผ่านตัวกลางไปที่ผู้รับ ผู้รับจะดำเนินการจากชั้นที่ 1 ขึ้นไปจนถึงชั้นที่ 7 เพื่อให้ได้ข้อมูลนั้น

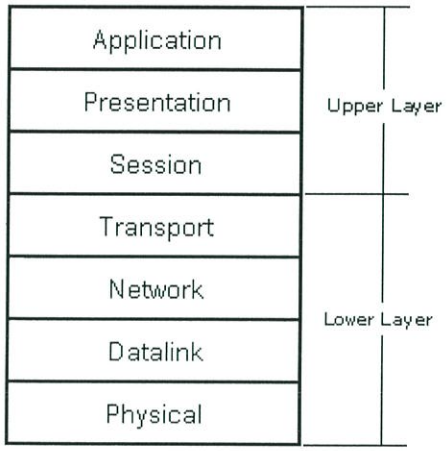
OSI Model เป็นมาตรฐานที่ใช้อ้างอิงถึงวิธีการในการส่งข้อมูลจาก Computer เครื่องหนึ่งผ่าน Network ไปยัง Computer อีกเครื่องหนึ่ง ซึ่งหากไม่มีการกำหนดมาตรฐานกลางแล้ว การพัฒนาและใช้งานที่เกี่ยวกับ Network ทั้ง Hardware และ Software ของผู้ผลิตที่เป็นคนละผู้ผลิต อาจเกิดปัญหาเนื่องจากการไม่ compatible กัน OSI เป็น Model ในระดับแนวคิด ประกอบด้วย Layer ต่างๆ 7 ชั้น แต่ละ Layer จะอธิบายถึงหน้าที่การทำงานกับข้อมูล

OSI Model พัฒนาโดย International Organization for Standardization (ISO) ในปี 1984 และเป็นสถาปัตยกรรมโมเดลหลักที่ใช้อ้างอิงในการสื่อสารระหว่าง Computer โดยข้อดีของ OSI Model คือ แต่ละ Layer จะมีการทำงานที่เป็นอิสระจากกัน ดังนั้นจึงสามารถออกแบบอุปกรณ์ของแต่ละ Layer แยกจากกันได้ และการปรับปรุงใน Layer หนึ่งจะไม่มีผลกระทบกับ Layer อื่นๆ

- 7 Layer ของ OSI Model สามารถแบ่งได้เป็น 2 กลุ่ม คือ upper layers และ lower layers
- Upper layers โดยทั่วไปจะเป็นส่วนที่พัฒนาใน Software Application โดยประกอบด้วย Application Layer, Presentation Layer และ Session Layer

- Lower Layers จะเป็นส่วนที่ทำหน้าที่ในการสื่อสารข้อมูลซึ่งอาจจะพัฒนาได้ทั้งแบบเป็น Software และ Hardware

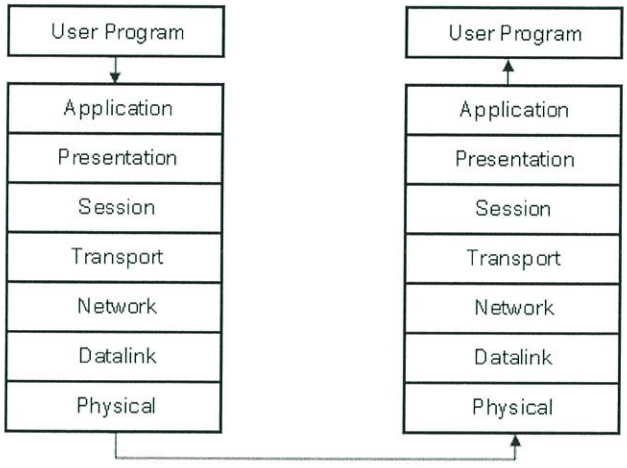
OSI Model ประกอบด้วย 7 Layer คือ



รูปที่ 2.2 รายละเอียดของ OSI Model

ข้อมูลข่าวสารที่ส่งจาก Application บน Computer เครื่องหนึ่ง ไปยัง Application บน Computer จะต้องส่งผ่านแต่ละ Layer ของ OSI Model ตามลำดับ ดังรูป

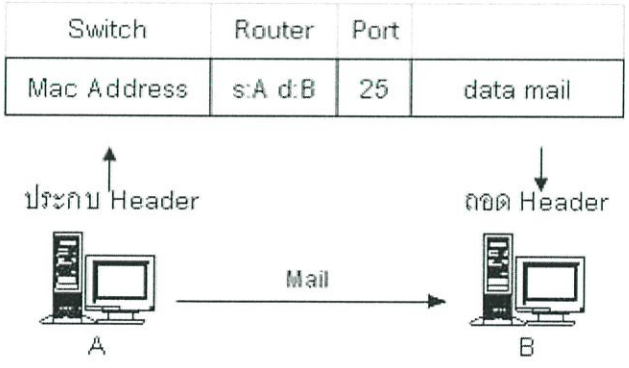
โดย Layer แต่ละ Layer จะสามารถสื่อสารได้กับ Layer ข้างเคียงในขั้นสูงกว่าและต่ำกว่า และ Layer เดียวกันในอีกระบบ Computer เท่านั้น



รูปที่ 2.3 Data ที่จะส่งจะถูกเพิ่ม Header ของแต่ละชั้นเข้าไป เมื่อมีการรับข้อมูลที่ปลายทางแล้ว header จะถูกถอดออกตามลำดับชั้น

ตัวอย่าง ในการส่ง Mail จะถูกประกบ Header เข้าไป 3 ชั้นเรียงจากบนลงมาคือ

ชั้น Transport จะใส่เบอร์ Port ของ Mail คือ Port 25
ชั้น Network จะถูกใส่ต้นทางและปลายทางโดย Router
ชั้น Datalink จะใส่เป็น Mac Address โดย Switch



รูปที่ 2.4 หน้าที่ในแต่ละ Layer ตามมาตรฐาน OSI

Physical Layer ชั้น Physical เป็นการอธิบายคุณสมบัติทางกายภาพ เช่น คุณสมบัติทางไฟฟ้าและกลไกต่างๆ ของวัสดุที่ใช้เป็นสื่อกลาง ตลอดจนสัญญาณที่ใช้ในการส่งข้อมูล คุณสมบัติที่กำหนดไว้ในชั้นนี้ประกอบด้วยคุณลักษณะทางกายภาพของสาย, อุปกรณ์เชื่อมต่อ (Connector), ระดับความต่างศักย์ของไฟฟ้า (Voltage) และอื่นๆ เช่น อธิบายถึงคุณสมบัติของสาย Unshielded Twisted Pair(UTP)

Data link Layer ชั้น Data link เป็นชั้นที่อธิบายถึงการส่งข้อมูลไปบนสื่อกลาง ชั้นนี้ยังได้ถูกแบ่งออกเป็นชั้นย่อย (Sub Layer) คือ Logical Link Control (LLC) และ Media Access Control (MAC) การแบ่งแยกเช่นนี้จะทำให้ชั้น LLC ชั้นเดียวสามารถจะใช้ชั้น MAC ที่แตกต่างกันออกไปได้หลายชั้น ชั้น MAC นั้นเป็นการดำเนินการเกี่ยวกับแอดเดรสทางกายภาพอย่างที่ใช้ในมาตรฐานอีเทอร์เนตและโทเคนริง แอดเดรสทางกายภาพนี้จะถูกฝังมาในการ์ดเครือข่ายโดยบริษัทผู้ผลิตการ์ดนั้น แอดเดรสทางกายภาพนั้นเป็นคนละอย่างกับแอดเดรสทางตรรกะ เช่น IP Address ที่จะถูกใช้งานในชั้น Network เพื่อความชัดเจนครบถ้วนสมบูรณ์ของการใช้ชั้น Data-Link นี้

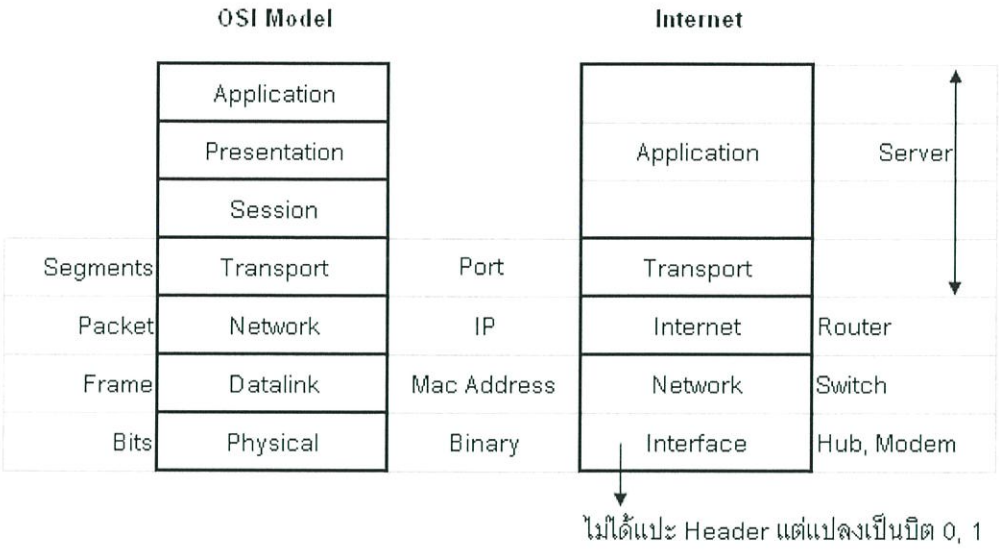
Network Layer ในขณะที่ชั้น Data-Link ให้ความสนใจกับแอดเดรสทางกายภาพ แต่การทำงานในชั้น Network จะให้ความสนใจกับแอดเดรสทางตรรกะ การทำงานในชั้นนี้จะเป็นการเชื่อมต่อและการเลือกเส้นทางนำพาข้อมูลระหว่างเครื่องสองเครื่องในเครือข่าย
ชั้น Network ยังให้บริการเชื่อมต่อในแบบ “Connection Oriented” อย่างเช่น X.25 หรือบริการแบบ “Connectionless” เช่น Internet Protocol ซึ่งใช้งานโดยชั้น Transport ตัวอย่างของบริการหลักที่ชั้น Network มีให้คือ การเลือกเส้นทางนำพาข้อมูลไปยังปลายทางที่เรียกว่า Routing
ตัวอย่างของโพรโทคอลในชั้นนี้ประกอบด้วย Internet Protocol (IP) และ Internet Control Message Protocol (ICMP)

Transport Layer ในชั้นนี้มีบางโพรโทคอลจะให้บริการที่ค่อนข้างคล้ายกับที่มีในชั้น Network โดยมีบริการด้านคุณภาพที่ทำให้เกิดความน่าเชื่อถือ แต่ในบางโพรโทคอลที่ไม่มีการดูแลเรื่องคุณภาพดังกล่าวจะอาศัยการทำงานในชั้น Transport นี้เพื่อเข้ามาช่วยดูแลเรื่องคุณภาพแทน เหตุผลที่สนับสนุนการใช้งานชั้นนี้คือ ในบางสถานการณ์ของชั้นในระดับล่างทั้งสาม (คือชั้น Physical, Data-Link และ Network) ดำเนินการโดยผู้ให้บริการโทรคมนาคม การจะเพิ่มความมั่นใจในคุณภาพให้กับผู้ใช้บริการด้วยการใช้ชั้น Transport นี้

Session Layer ชั้น Session ทำหน้าที่สร้างการเชื่อมต่อ, การจัดการระหว่างการเชื่อมต่อ และการตัดการเชื่อมต่อคำว่า "เซสชัน" (Session) นั้นหมายถึงการเชื่อมต่อกันในเชิงตรรกะ (Logic) ระหว่างปลายทางทั้งสองด้าน (เครื่อง 2 เครื่อง) ชั้นนี้อาจไม่จำเป็นต้องถูกใช้งานเสมอไป อย่างเช่นถ้าการสื่อสารนั้นเป็นไปในแบบ "Connectionless" ที่ไม่จำเป็นต้องเชื่อมต่อ เป็นต้น ระหว่างการสื่อสารในแบบ "Connection-Less" ทุกแพ็กเก็ต (Packet) ของข้อมูลจะมีข้อมูลเกี่ยวกับเครื่องปลายทางที่เป็นผู้รับติดอยู่อย่าง สมบูรณ์ในลักษณะของจดหมายที่มีการจ่าหน้าซองอย่างถูกต้องครบถ้วน ส่วนการสื่อสารในแบบ "Connection Oriented" จะต้องมีการดำเนินการบางอย่างเพื่อให้เกิดการเชื่อมต่อ หรือเกิดเป็นวงจรในเชิงตรรกะขึ้นมาก่อนที่การรับ/ส่งข้อมูลจะเริ่มต้นขึ้น เมื่อการรับ/ส่งข้อมูลดำเนินไปจนเสร็จสิ้นต้องมีการดำเนินการบางอย่าง เพื่อที่จะตัดการเชื่อมต่อลง ตัวอย่างของการเชื่อมต่อแบบนี้ได้แก่การใช้โทรศัพท์ที่ต้องมีการกดหมายเลขปลายทาง จากนั้น ต้องมีการดำเนินการบางอย่างของระบบจนกระทั่งเครื่องปลายทางมีเสียงดังขึ้น การสื่อสารจะเริ่มขึ้นจริงเมื่อมีการทักทายกันของคู่สนทนา จากนั้นเมื่อคู่สนทนาฝ่ายใดฝ่ายหนึ่งวางหู ต้องมีการดำเนินการบางอย่างที่ จะตัดการเชื่อมต่อลงชั้น Session นี้มีระบบการติดตามด้วยว่าฝั่งใดที่ส่งข้อมูลซึ่งเรียกว่า "Dialog Management" Simple Mail Transport Protocol (SMTP), File Transfer Protocol (FTP) และ Telnet เป็นตัวอย่างของโพรโทคอลที่นิยมใช้ และมีการทำงานครอบคลุมในชั้น Session, Presentation และ Application

Presentation Layer ชั้น Presentation ให้บริการทำการตกลงกันระหว่างสองโพรโทคอลถึงไวยากรณ์ (Syntax) ที่จะใช้ในการรับ/ส่งข้อมูล เนื่องจากว่าไม่มีการรับรองถึงไวยากรณ์ที่จะใช้ร่วมกัน การทำงานในชั้นนี้จึงมีบริการในการแปลข้อมูลตามที่ได้รับการร้องขอด้วย

Application Layer ชั้น Application เป็นชั้นบนสุดของแบบจำลอง ISO/OSI เป็นชั้นที่ใช้บริการของชั้น Presentation (และชั้นอื่นๆ ในทางอ้อมด้วย) เพื่อประยุกต์ใช้งาน เช่น การทำ E-mail Exchange (การรับ/ส่งอีเมล์) การโอนย้ายไฟล์ หรือการประยุกต์ใช้งานทางด้านเครือข่ายอื่นๆ



รูปที่ 2.5 การเปรียบเทียบระหว่าง OSI Model กับการสื่อสารของ Internet โดยจะแสดงรูปแบบข้อมูล, Data และอุปกรณ์ที่ทำงานอยู่ในแต่ละ Layer

2.2 ระบบปฏิบัติการ Android [4]

ในโลกของการติดต่อสื่อสารในปัจจุบันได้มีการพัฒนาที่ก้าวหน้าเป็นอย่างมาก โดยเฉพาะการสื่อสารแบบไร้สาย ที่ได้มีการพัฒนาความเร็วในการรับส่งข้อมูลที่สูงขึ้น จากเดิมที่มีการส่งได้เพียงข้อความสั้น (SMS: Short Message Service) และ MMS (Multimedia Messaging Service) ปัจจุบันสามารถทำการโทรศัพท์แบบเห็นหน้าคู่สนทนากันได้ (Video Call) แต่ต้องผ่านทางระบบของวายฟาย (Wi-Fi: Wireless Fidelity) หรือ ระบบสามจี (3G: Third Generation of Mobile Telephone)

ซึ่งสำหรับประเทศไทยแล้ว อุปกรณ์มือถือ และอุปกรณ์พกพา ส่วนมากในตลาดจะรองรับระบบการรับส่งข้อมูลความเร็วสูงเป็นที่เรียบร้อยแล้ว และได้รับความนิยมจากผู้ใช้งานเป็นจำนวนมาก โดยอุปกรณ์ส่วนใหญ่ที่มีอยู่ในท้องตลาดจะมีระบบปฏิบัติการเป็นของตัวเองที่ไม่เหมือนกับระบบปฏิบัติการที่อยู่บนคอมพิวเตอร์ส่วนบุคคล (PC: Personal Computer) ส่งผลให้แนวทางในการพัฒนาโปรแกรม เพื่อนำไปใช้งานบนอุปกรณ์เหล่านั้นยุ่งยาก และหลากหลายขึ้น

ระบบปฏิบัติการบนอุปกรณ์ดังกล่าว มีอยู่หลายตัวกันเช่น Android, iOS, Windows Phone, BlackBerry, Symbian, webOS, MeeGo และ QNX เป็นต้น โดยลักษณะของระบบปฏิบัติการข้างต้น ส่วนมากจะเป็นประเภทไม่เปิดเผยซอร์ฟแวร์ต้นฉบับ (Proprietary) ซึ่งหมายความว่า ระบบปฏิบัติการดังกล่าว ไม่สามารถนำมาศึกษา ดัดแปลงการทำงานของระบบปฏิบัติการเพื่อนำไปใช้งานตามที่ต้องการได้ ทำให้เกิดความไม่สะดวกในการพัฒนา และการพัฒนาจะถูกกำหนดทิศทางโดยบริษัทเจ้าของลิขสิทธิ์

แอนดรอยด์ (Android) คือระบบปฏิบัติการแบบเปิดเผยซอร์ฟแวร์ต้นฉบับ (Open Source) โดยบริษัท กูเกิ้ล (Google Inc.) ที่ได้รับความนิยมเป็นอย่างสูง เนื่องจากอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์ มีจำนวนมาก อุปกรณ์มีหลากหลายระดับ หลายราคา รวมทั้งสามารถทำงานบนอุปกรณ์ที่มีขนาดหน้าจอ และความละเอียดแตกต่างกันได้ ทำให้ผู้บริโภคสามารถเลือกได้ตามต้องการ และหากมองในทิศทางสำหรับนักพัฒนาโปรแกรม (Developer) แล้วนั้น การพัฒนาโปรแกรมเพื่อใช้งานบนระบบปฏิบัติการแอนดรอยด์ ไม่ใช่เรื่องที่ยาก เพราะมีข้อมูลในการพัฒนารวมทั้ง Android SDK (Software Development Kit) เตรียมไว้ให้กับนักพัฒนาได้เรียนรู้ และเมื่อนักพัฒนาต้องการจะเผยแพร่หรือจำหน่ายโปรแกรมแอนดรอยด์ ยังมีตลาดในการเผยแพร่โปรแกรม ผ่าน Android Market สำหรับ Android SDK จะยึดโครงสร้างของภาษาจาวา (Java Programming language) ในการเขียนโปรแกรม เพราะโปรแกรมที่พัฒนามาได้จะต้องทำงานอยู่ภายใต้ Dalvik Virtual Machine เช่นเดียวกับโปรแกรม

จาวา ที่ต้องทำงานอยู่ภายใต้ Java Virtual Machine (Virtual Machine เปรียบได้กับสภาพแวดล้อมที่โปรแกรมทำงานอยู่)

2.2.1 ประวัติความเป็นมา

เริ่มต้นระบบปฏิบัติการแอนดรอยด์ ถูกพัฒนามาจากบริษัท แอนดรอยด์ (Android Inc.) เมื่อปี พ.ศ. 2546 โดยมีนาย แอนดี้ รูบิน (Andy Rubin) ผู้ให้กำเนิดระบบปฏิบัติการนี้ และถูกบริษัท กูเกิ้ล ซื้อกิจการเมื่อ เดือนสิงหาคม ปี พ.ศ. 2548 โดยบริษัทแอนดรอยด์ ได้กลายเป็นมาบริษัทลูก ของบริษัทกูเกิ้ล และยังมีนาย แอนดี้ รูบิน ดำเนินงานอยู่ในทีมพัฒนาระบบปฏิบัติการต่อไป

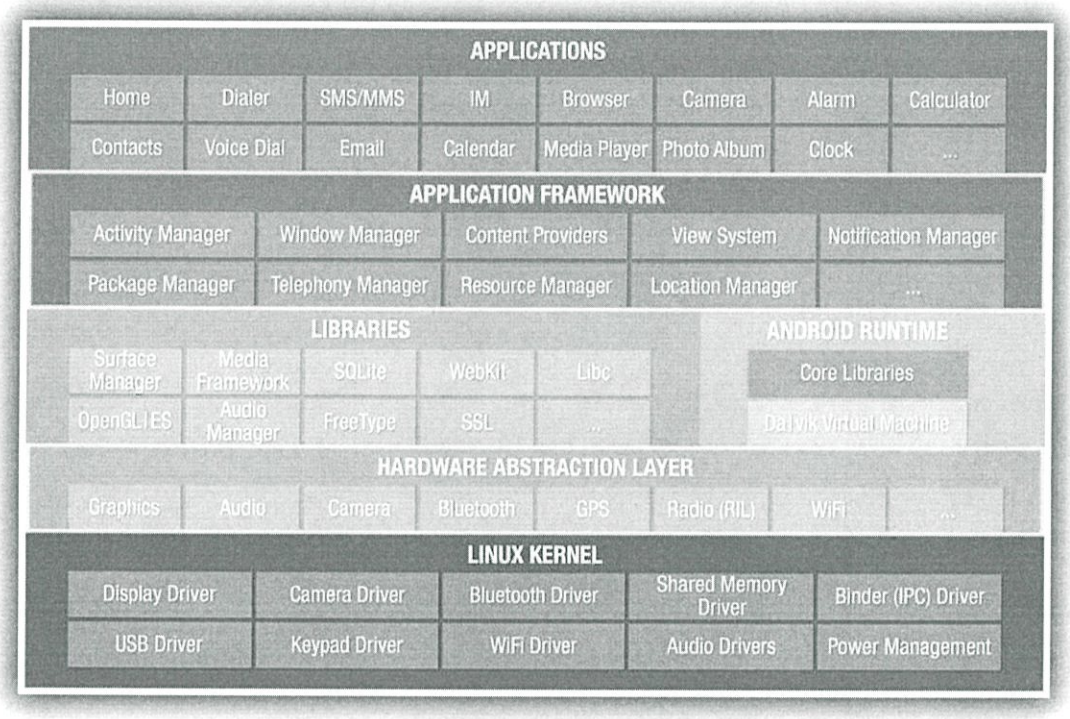
ระบบปฏิบัติการแอนดรอยด์ เป็นระบบปฏิบัติการที่พัฒนามาจากการนำเอา แกนกลางของระบบปฏิบัติการลินุกซ์ (Linux Kernel) ซึ่งเป็นระบบปฏิบัติการที่ออกแบบมาเพื่อทำงานเป็นเครื่องให้บริการ (Server) มาพัฒนาต่อ เพื่อให้กลายเป็นระบบปฏิบัติการบนอุปกรณ์พกพา (Mobile Operating System)

ต่อมาเมื่อเดือน พฤศจิกายน ปี พ.ศ. 2550 บริษัทกูเกิ้ล ได้ทำการก่อตั้งสมาคม OHA (Open Handset Alliance) เพื่อเป็นหน่วยงานกลางในการกำหนดมาตรฐานกลาง ของอุปกรณ์พกพาและระบบปฏิบัติการแอนดรอยด์ โดยมีสมาชิกในช่วงก่อนตั้งจำนวน 34 รายเข้าร่วม ซึ่งประกอบไปด้วยบริษัทชั้นนำที่ดำเนินธุรกิจด้านการสื่อสาร เช่น โรงงานผลิตอุปกรณ์พกพา บริษัทพัฒนาโปรแกรม ผู้ให้บริการสื่อสาร และผู้ผลิตอะไหล่อุปกรณ์ด้านสื่อสาร

หลังจากนั้น เมื่อเดือนตุลาคม ปี พ.ศ. 2551 บริษัท กูเกิ้ล ได้เปิดตัวมือถือตัวแรกที่ใช้ระบบปฏิบัติการแอนดรอยด์ ที่ชื่อ T-Mobile G1 หรืออีกชื่อหนึ่งคือ HTC Dream โดยใช้แอนดรอยด์รุ่น 1.1 และหลังจากนั้น ได้มีการปรับพัฒนาระบบปฏิบัติการเป็นรุ่นใหม่ มาเป็นลำดับช่วงต่อมาได้มีการออกผลิตภัณฑ์จากบริษัทต่างๆ ออกมาหลากหลายรุ่น หลากหลายยี่ห้อ ตามการพัฒนาระบบปฏิบัติการแอนดรอยด์ ที่มีอยู่อย่างต่อเนื่อง ทำให้สินค้าของแอนดรอยด์ มีให้เลือกอยู่อย่างมาก

2.2.2 โครงสร้างของระบบปฏิบัติการแอนดรอยด์

การทำความเข้าใจโครงสร้างของระบบปฏิบัติการแอนดรอยด์ ถือว่าเป็นสิ่งสำคัญเพราะถ้านักพัฒนาโปรแกรม สามารถมองภาพโดยรวมของระบบได้ทั้งหมด จะให้สามารถเข้าใจถึงกระบวนการทำงานได้ดียิ่งขึ้น และสามารถนำไปช่วยในการออกแบบโปรแกรมที่ต้องการพัฒนาเพื่อให้เกิดประสิทธิภาพในการทำงาน



รูปที่ 2.6 โครงสร้างของระบบปฏิบัติการแอนดรอยด์

จากโครงสร้างของระบบปฏิบัติการแอนดรอยด์จะสังเกตได้ว่า มีการแบ่งออกมาเป็นส่วนๆ ที่มีความเกี่ยวเนื่องกัน โดยส่วนบนสุดจะเป็นส่วนที่ผู้ใช้งานทำการติดต่อโดยตรงซึ่งก็คือส่วนของ (Applications) จากนั้นก็จะลำดับลงมาเป็นองค์ประกอบอื่นๆตามลำดับ และสุดท้ายจะเป็นส่วนที่ติดต่อกับอุปกรณ์โดยผ่านทาง Linux Kernel โครงสร้างของแอนดรอยด์ พอที่จะอธิบายเป็นส่วนๆได้ดังนี้

1. Applications ส่วน Application หรือส่วนของโปรแกรมที่มีมากับระบบปฏิบัติการ หรือเป็นกลุ่มของโปรแกรมที่ผู้ใช้งานได้ทำการติดตั้งไว้ โดยผู้ใช้งานสามารถเรียกใช้โปรแกรมต่างๆได้โดยตรง ซึ่งการทำงานของแต่ละโปรแกรมจะเป็นไปตามที่ผู้พัฒนาโปรแกรมได้ออกแบบและเขียนโปรแกรมเอาไว้

2. Application Framework เป็นส่วนที่มีการพัฒนาขึ้นเพื่อให้นักพัฒนาสามารถพัฒนาโปรแกรมได้สะดวก และมีประสิทธิภาพมากยิ่งขึ้น โดยนักพัฒนาไม่จำเป็นต้องพัฒนาในส่วนที่มีความยุ่งยาก เพียงแค่ทำการศึกษาถึงวิธีการเรียกใช้งาน Application Framework ในส่วนที่ต้องการใช้งาน แล้วนำมาใช้งาน ซึ่งมีหลายกลุ่มด้วยกัน ตัวอย่างเช่น
    1) Activities Manager เป็นกลุ่มของชุดคำสั่งที่จัดการเกี่ยวกับวงจรการทำงานของหน้าต่างโปรแกรม (Activity)
    2) Content Providers เป็นกลุ่มของชุดคำสั่ง ที่ใช้ในการเข้าถึงข้อมูลของโปรแกรมอื่น และสามารถแบ่งปันข้อมูลให้โปรแกรมอื่นเข้าถึงได้
    3) View System เป็นกลุ่มของชุดคำสั่งที่เกี่ยวกับการจัดการ โครงสร้างของหน้าจอที่แสดงผลในส่วนที่ติดต่อกับผู้ใช้งาน (User Interface)
    4) Telephony Manager เป็นกลุ่มของชุดคำสั่งที่ใช้ในการเข้าถึงข้อมูลด้านโทรศัพท์ เช่นหมายเลขโทรศัพท์ เป็นต้น
    5) Resource Manager เป็นกลุ่มของชุดคำสั่งในการเข้าถึงข้อมูลที่เป็น ข้อความ, รูปภาพ
    6) Location Manager เป็นกลุ่มของชุดคำสั่งที่เกี่ยวกับตำแหน่งทางภูมิศาสตร์ ที่ระบบปฏิบัติการได้รับค่าจากอุปกรณ์
    7) Notification Manager เป็นกลุ่มของชุดคำสั่งที่จะถูกเรียกใช้เมื่อโปรแกรม ต้องการแสดงผลให้กับผู้ใช้งาน ผ่านทางแถบสถานะ (Status Bar) ของหน้าจอ
    8) Libraries เป็นส่วนของชุดคำสั่งที่พัฒนาด้วย C/C++ โดยแบ่งชุดคำสั่งออกเป็นกลุ่มตามวัตถุประสงค์ของการใช้งาน เช่น Surface Manage จัดการเกี่ยวกับการแสดงผล, Media Framework จัดการเกี่ยวกับการการแสดงภาพและเสียง, Open GL | ES และ SGL จัดการเกี่ยวกับภาพ 3มิติ และ 2มิติ, SQLlite จัดการเกี่ยวกับระบบฐานข้อมูล เป็นต้น
    9) Android Runtime จะมี Darvik Virtual Machine ที่ถูกออกแบบมา เพื่อให้ทำงานบนอุปกรณ์ที่มี หน่วยความจำ (Memory), หน่วยประมวลผลกลาง (CPU) และพลังงาน (Battery) ที่จำกัด ซึ่งการทำงานของ Darvik Virtual Machine จะทำการแปลงไฟล์ที่ต้องการทำงาน ไปเป็นไฟล์ .DEX ก่อนการทำงาน เหตุผลก็เพื่อให้มีประสิทธิภาพเพิ่มขึ้นเมื่อใช้งานกับ หน่วยประมวลผลกลางที่มีความเร็วไม่มาก

ส่วนต่อมาคือ Core Libraries ที่เป็นส่วนรวบรวมคำสั่งและชุดคำสั่งสำคัญ โดยถูกเขียนด้วยภาษาจาวา (Java Programming Language)

10) Linux Kernel เป็นส่วนที่ทำหน้าที่หัวใจสำคัญ ในจัดการกับบริการหลักของระบบปฏิบัติการ เช่น เรื่องหน่วยความจำ พลังงาน ติดต่อกับอุปกรณ์ต่างๆ ความปลอดภัย เครือข่าย โดยแอนดรอยด์ได้นำเอาส่วนนี้มาจากระบบปฏิบัติการ Linux รุ่น 2.6 (Linux 26. Kernel) ซึ่งได้มีการออกแบบมาเป็นอย่างดี

2.2.3 เวอร์ชันในระบบปฏิบัติการแอนดรอยด์ [5]

Android OS เป็นระบบปฏิบัติการที่พัฒนามาจาก Linux Kernel โดยเริ่มพัฒนาระบบเมื่อปี ค.ศ. 2003 โดย Android Inc. จากนั้น Google ได้ทำการพัฒนาระบบดังกล่าวลงบน โทรศัพท์เคลื่อนที่ Netbook และ Android Tablet ต่อมาในปี ค.ศ. 2007 บริษัท Google ได้หาพันธมิตรโดยได้ประกาศให้เป็น Open Source และผู้ผลิตก็สามารถนำ Android OS ไปใส่ในระบบมือถือ และเมื่อปี ค.ศ. 2003 จนถึงปัจจุบัน ทางบริษัท Google ได้พัฒนา Android OS ในเวอร์ชันต่างๆ ออกมามากมาย โดยใช้ชื่อของต่างเป็นชื่อของขนมหวาน ดังนี้

1) Android 1.0 Apple Pie เปิดตัวเมื่อวันที่ 23 กันยายน 2008 บน HTC G1 ซึ่งความสามารถของ Apple Pie คือ สามารถติดกับอุปกรณ์พื้นฐาน แต่ยังขาดคุณสมในการติดต่อกล้อง และ Bluetooth
2) Android 1.1 Banana Bread เปิดตัวเมื่อวันที่ 9 กุมภาพันธ์ 2009 มีความสามารถเพิ่มขึ้นมาจาก Apple pie คือ มีความสามารถติดต่อกับอุปกรณ์ Bluetooth A2DP และมีการเพิ่ม Soft Keyboard และความสามารถในการใช้งาน Videos
3) Android 1.5 Cupcake เปิดตัวเมื่อวันที่ 30 เมษายน 2009 เป็นระบบปฏิบัติการพัฒนาบนพื้นฐานของระบบปฏิบัติการ Linux Kernel 2.6.27
4) Android 1.6 Donut เปิดตัวเมื่อวันที่ 15 สิงหาคม 2009 เป็นระบบปฏิบัติการพัฒนาบนพื้นฐานของระบบปฏิบัติการ Linux Kernel 2.6.29 ซึ่งเราสามารถพัฒนาแอปพลิเคชันสำหรับระบบปฏิบัติการนี้ด้วย Eclipse
5) Android 2.0 / 2.1 Éclair เปิดตัวเมื่อวันที่ 26 ตุลาคม 2009 และ 12 มกราคม 2010 (2.1 SDK) โดย Éclair มีการพัฒนาให้มีความสามารถเพิ่มขึ้นจาก Donut ซึ่ง 2.1 นั้นสามารถพัฒนาแอปพลิเคชันด้วย Eclipse ได้

6) Android 2.2 Froyo เปิดตัวเมื่อวันที่ 20 พฤษภาคม 2010 สามารถพัฒนาแอปพลิเคชันด้วย Eclipse

7) Android 3.0 / 3.1 Honeycomb เปิดตัวเมื่อวันที่ 6 ธันวาคม 2010 เป็น OS แรกที่ทำสำหรับ Tablets ซึ่งรองรับแค่หน้าจอ 10" / 8.9" บางรุ่นเท่านั้น สามารถพัฒนาแอปพลิเคชันด้วย Eclipse ได้ และยังมีความสามารถที่ต่างออกไปจาก Gingerbread

8) Android 3.2 Honeycomb เปิดตัวเมื่อวันที่ 19 ตุลาคม 2011 ซึ่ง Honeycomb รองรับ Tablet เท่านั้นและเพิ่มความสามารถในการรองรับ Tablets ขนาด 7 นิ้ว

9) Android 4.0 Ice Cream Sandwich เปิดตัวเมื่อวันที่วันที่ 19 ตุลาคม 2012 ซึ่งเปิดตัวพร้อมกับ Samsung Galaxy Nexus ที่ใช้ระบบปฏิบัติการแอนดรอยด์ 4.0 Ice Cream Sandwich โดยระบบปฏิบัติการแอนดรอยด์ 4.0 Ice Cream Sandwich ใช้ได้กับทั้งสมาร์ทโฟน และแท็บเล็ตภายในระบบปฏิบัติการเดียว ไม่เหมือนกับระบบปฏิบัติการแอนดรอยด์ 3.0 Honeycomb ที่ใช้งานได้เฉพาะแท็บเล็ตเท่านั้นนอกจากนี้ ระบบปฏิบัติการแอนดรอยด์ 4.0 Ice Cream Sandwich ยังเป็นแบบ Open source ที่เปิดทางให้กับนักพัฒนาเข้ามาสร้างแอปพลิเคชัน เพื่อรองรับและพัฒนาระบบปฏิบัติการแอนดรอยด์ 4.0 Ice Cream Sandwich ต่อไปในอนาคต

10) Android 4.1 Jelly Bean เป็นการปรับปรุงและเพิ่มประสิทธิภาพให้ตัวระบบปฏิบัติการมีความเสถียร ไหลลื่น และตอบสนองได้ดียิ่งขึ้น จะไม่มีการกระตุก เพราะได้มีการปรับปรุงโครงสร้างและใช้เครื่องมือแบบใหม่ในการ Render ที่เรียกว่า Vsync และ Triple Buffering ทำให้หน้าจอแสดงผลกราฟิกต่างๆได้สูงถึง 60 FPS (ซึ่งเครื่องมือดังกล่าวจะเรียกกันว่า Systrace จะมีอยู่ในชุดพัฒนา SDK ให้นักพัฒนาด้วย)

2.3 การออกแบบสำหรับระบบสัมผัส (Designing for touch) [6]

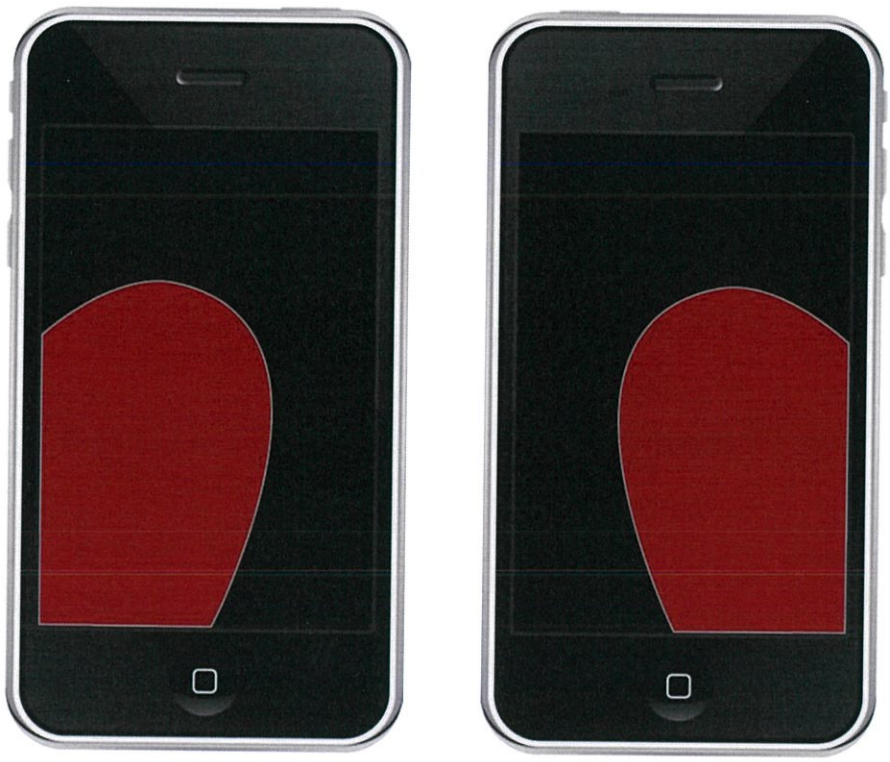
การสัมผัสหน้าจอโดยใช้นิ้วได้ถูกนำมาเป็นหัวข้อของการประชุม Josh Clark นักออกแบบระบบปฏิสัมพันธ์ ได้อธิบายถึงสิ่งที่ต้องคำนึงถึงในการออกแบบระบบสัมผัสบนโทรศัพท์มือถือ และเปรียบเทียบระบบสัมผัสที่ดีของ iPhone, iPad และ Android

การออกแบบโทรศัพท์มือถือที่ดี ต้องทำมากกว่าการใส่ฟังก์ชันการทำงานมากๆลงในหน้าจอขนาดเล็ก แต่ต้องคำนึงถึงการใช้งานของผู้ใช้โดยนิ้วมือ และรองรับการสัมผัสหน้าจอที่ผิดพลาดโดยผู้ใช้ เพื่อไม่เป็นการโทษผู้ใช้ โดยในหน้าจอของมือถือ สามารถนำนักออกแบบเข้าสู่การประชุมที่นอกเหนือจากการออกแบบสิ่งที่มองเห็นและข้อมูลของการออกแบบ และเข้าไปในเรื่องทางด้านอุตสาหกรรมการออกแบบ เพราะระบบหน้าจอสัมผัส (Touchscreen) มีหลักการ การยศาสตร์ (Ergonomics) เป็นส่วนหลักไม่ใช่เรื่องการมองจุดแต่ละจุดจะเป็นอย่างไร แต่เป็นความรู้สึกว่ารู้สึกอย่างไร ซึ่ง การยศาสตร์ เป็นเรื่องการศึกษาสภาพการทำงานที่มีความสัมพันธ์ระหว่างผู้ใช้งานและสิ่งแวดล้อม การทำงานเป็นการพิจารณาว่าสถานที่ทำงานดังกล่าว ได้มีการออกแบบหรือปรับปรุงให้มีความเหมาะสมกับผู้ใช้งานอย่างไร เพื่อป้องกันปัญหาต่าง ๆ ที่อาจมีผลกระทบต่อจิตใจของผู้ใช้ในการทำงาน และสามารถเพิ่มประสิทธิภาพการทำงานได้ด้วย หรืออาจกล่าวอีกนัยหนึ่งได้ว่า เพื่อทำให้งานที่ต้องปฏิบัติดังกล่าว มีความเหมาะสมกับผู้ใช้งาน แทนที่จะบังคับให้ผู้ปฏิบัติงานต้องทนฝืนปฏิบัติงานนั้น ๆ ตัวอย่างเช่น การเพิ่มระดับความสูงของโต๊ะทำงานให้สูงขึ้น เพื่อพนักงานจะได้ไม่ต้องก้มโน้มตัวเข้าใกล้ชิ้นงาน

2.3.1 หลักการทั่วไป (Rule of thumb)

การออกแบบหน้าจอสัมผัสมีความต้องการด้านความสะดวกสบายของการใช้นิ้วมือที่ใช้สัมผัสบนหน้าจอตัวอย่างเช่น การใช้มือข้างเดียวในการจับโทรศัพท์มือถือ ถ้าจับโทรศัพท์แบบกุมไว้แล้วจะต้องใช้นิ้วหัวแม่มือในการสัมผัสหน้าจอเป็นต้น สำหรับโทรศัพท์มือถือการออกแบบการสัมผัสก็หมายถึงการออกแบบสำหรับนิ้วหัวแม่มือ

นิ้วหัวแม่มือของเราเป็นนิ้วที่ใหญ่ แต่จะมีขอบเขตและความยืดหยุ่นที่มีข้อจำกัด ในขณะที่นิ้วหัวแม่มือสามารถสัมผัสไปทั้งหน้าจอได้ แต่จะสามารถใช้งานได้แค่หนึ่งในสามของหน้าจอของโทรศัพท์ที่มีขนาดใหญ่ บริเวณด้านล่างของหน้าจอตรงข้ามกับหัวนิ้วมือบริเวณหลักที่เรามักจะใช้นิ้วแม่มือสัมผัส บริเวณที่ใช้ง่านบ่อย ตัวอย่างเช่น เมื่อมือถือของเราถูกถือด้วยมือขวา ธรรมชาติของนิ้วหัวแม่มือ จะอยู่ในส่วนล่างของหน้าจอ



รูปที่ 2.7 Active area touch

นั่นเป็นเหตุผลที่สำคัญว่าทำไมแถบเครื่องมือและการนำทาง จะอยู่ในบริเวณขอบด้านล่างของอินเตอร์เฟสโทรศัพท์ ตรงข้ามกับสิ่งที่เราคุ้นเคยกับหน้าจอแบบดั้งเดิม ซอฟต์แวร์ในคอมพิวเตอร์ส่วนใหญ่จะใส่เมนูไว้ที่ด้านบนของหน้าจอ และวางตำแหน่งของส่วนนำทางหลักไว้ด้านบนของหน้าเว็บ อย่างไรก็ตามจุดหมายหลักจะอยู่ด้านล่าง การให้ความสำคัญกับหน้าจอด้านล่างของหัวนิ้วมือมีความสำคัญกว่า เมื่อเทียบกับซ้ายขวา ซึ่งส่วนใหญ่ใช้มือซ้ายกับขวาสลับกันเมื่อใช้โทรศัพท์มือถือ

บางครั้งผู้ใช้ที่ถนัดมือขวาจะสลับไปใช้มือซ้าย (ตัวอย่างเช่น เมื่อเริ่มเขียน) และผู้ที่ถนัดมือซ้ายในทำนองเดียวกันบางครั้งจะเปลี่ยนไปใช้มือขวา ในขณะที่ผู้ใช้ส่วนใหญ่ค่อนข้างถนัดมือขวามากกว่ามือซ้าย แต่ก็ไม่มีมีอิทธิพลต่อการออกแบบมากพอ

2.3.2 หลักการทั่วไปเกี่ยวกับด้านล่างของหน้าจอ

อย่างไรก็ตาม การใช้มือซ้ายหรือขวานั้นไม่มีผล ที่จะช่วยให้คำแนะนำเกี่ยวกับวิธีการจัดลำดับขั้น เวลาสัมผัสปุ่มที่ใช้งานบ่อยควรวางไว้ล่างของหน้าจอเพื่อเรียกใช้งายได้ง่าย ในขณะที่การควบคุมอื่น ๆ ควรจะวางไว้ในจุดที่ไม่โดนการสัมผัส มันเป็นรูปแบบใน iOS เช่นการวางปุ่มแก้ไขที่มุมขวาบนในมุมมองง่าย แต่ห่างออกจากการเข้าถึงโดยบังเอิญ ทำให้เกิดการผิดพลาดน้อย

การดึงไอคอนควบคุมหลักมาไว้ด้านล่างของหน้าจอไม่ได้ทำเพื่อความสะดวกสบายนิ้วหัวแม่มืออย่างเดียว แต่ทำเพื่อสำหรับนิ้วมือที่บังหน้าจอ บริเวณด้านล่างของหน้าจอคือที่สามารถทำให้เนื้อหาถูกปกปิดโดยมือได้ง่าย เพื่อให้เนื้อหามองง่ายขึ้นควรวางตำแหน่งตัวควบคุมไว้ด้านล่างซึ่งเป็นที่คุ้นเคยรูปแบบที่ใช้กับอุปกรณ์ทางกายภาพมากที่สุด iPods, เครื่องคิดเลข, โทรศัพท์มือถือ, เครื่องชั่งน้ำหนัก โดยเนื้อหาอยู่ด้านบนและ ปุ่มควบคุมอยู่ด้านล่าง เป็นต้น

2.3.3 เอกลักษณ์ ของ Android

ทฤษฎีเกี่ยวกับการวางปุ่มได้ถูกนำไปเป็นแนวคิดของสมาร์ทโฟนระบบแอนดรอยด์ คือมีปุ่มด้านล่างของอุปกรณ์ ก่อนที่จะมีการเปิดตัว Android 3 Honeycomb ที่มีปุ่มหลักที่ใช้ควบคุมให้กดได้และพัฒนาเป็นปุ่มเสมือน (Haptic Button) ใน Android 4 Ice Cream Sandwich ปุ่มเหล่านี้เป็นปุ่มลัดของระบบ ซึ่งอยู่ด้านล่างของหน้าจอสำหรับควบคุม การเพิ่มตัวควบคุมที่ด้านล่างหน้าจอหมายถึงการซ่อนแถบเครื่องมือ แต่การกดพลาดก็เป็นเรื่องธรรมดา โดยเฉพาะปุ่มควบคุม บริเวณด้านล่าง ที่นิ้วหัวแม่มืออาจเลื่อนไปโดนอยู่บ่อยครั้ง เพราะฉะนั้น การวางไอคอนติดกันบริเวณด้านล่าง ควรจะหลีกเลี่ยงอยู่เสมอ แต่การวางไอคอนของแอปพลิเคชันบนแอนดรอยด์ควรวางไว้ด้านบน เพื่อไม่ให้เกิดการเบียดกันกับ ปุ่มหรือไอคอนหลักด้านล่าง ซึ่งขัดแย้งกับทฤษฎี แต่มันเป็นทางเลือกที่ดีกว่า เพื่อป้องกันไม่ให้เกิดข้อผิดพลาดจากนิ้วหัวแม่มือ สำหรับ Android ปุ่มการนำทางและปุ่มการควบคุม จะไว้ด้านบน นี่คือสิ่งที่แตกต่างกันกับ iPhone ที่ปุ่ม Home ไม่ได้เหมือนกับปุ่มAndroid เปรียบเทียบ Foursquare app สำหรับ Android ฝั่งซ้ายและสำหรับ iPhone จะเห็นความแตกต่าง ดังนี้

รูปที่ 2.8 แสดงความแตกต่างระหว่างระบบปฏิบัติการแอนดรอยด์และไอโอเอส

2.3.4 การออกแบบขนาด

หากผู้ใช้งานถือโทรศัพท์ ขนาดของนิ้วมือจะเป็นตัวตัดสินใจว่าจะใช้นิ้วใดในการสัมผัส นักออกแบบระบบสัมผัสต้องการสร้างเป้าหมายที่นิ้วมือใหญ่จะไม่สัมผัสล้มเหลว ซึ่งคนอ้วนจะสามารถแตะได้โดยไม่ต้องข้ามระดับโฟกัสไป

Apple ยืนยันว่าสิ่งที่พิจารณาเป็นแนวทางที่ดีที่สุดสำหรับโทรศัพท์มือถือทุกแพลตฟอร์ม คือสร้างการแตะขั้นต่ำไว้ที่ 44 จุด หรือ 7 มิลลิเมตร ซึ่งสำหรับเว็บขั้นต่ำ 44 พิกเซล ยังสามารถทำงานได้ดี สำหรับนักออกแบบที่คุ้นเคยกับคอมพิวเตอร์เดกส์ทอปจะรู้สึกว่าใหญ่อย่างมาก และเป้าหมายที่มีขนาดใหญ่ที่ไม่ใช่เรื่องง่ายที่จะแตะโดน

ในอุดมคติ เป้าหมายการแตะต่ำสุดคือ 44 x 44 แต่ความยืดหยุ่นบางครั้งจำเป็น แม้ iPhone จะควบคุมมาตรฐานกฎ 44 พิกเซล ในแป้นพิมพ์ เช่นปุ่ม เป็น สูง 44 แต่กลับกว้าง 30 ในทำนองเดียวกันในมุมมองแนวนอน ปุ่มสูง 44 แต่กว้าง 38 Apple ไม่มีทางเลือกในการรวมคีย์บอร์ด QWETY เนื่องจาก

ขนาด และปุ่มทั้งหมดก็จะไม่เป็น 44 x 44เมื่อพื้นที่จำกัด ทำให้ได้กฎของการเรียงปุ่มขนาดต่ำสุดที่ดีเป็น 44 x 30

2.3.5 การเรียงปุ่ม

ผู้พัฒนาคนหนึ่ง ใช้เวลาหลายปีของเขาไปกับ การสร้างนาฬิกาข้อมือ เครื่องคิดเลข Casio ของเขาก่อนที่จะเกษียณตัวเองมันในปี 1985 ปัญหาก็คือ การควบคุมเล็ก ๆ ส่งผลกระทบต่อลูกค้า ด้านการใช้งานภายใต้พื้นที่จำกัด ปุ่มที่ใกล้กันเกินไป จึงไม่ใช่การออกแบบที่ดี เพราะขนาดเล็กและระยะห่างที่ไม่สะดวกต่อการใช้งาน

รูปที่ 2.9 นักพัฒนาพัฒนานาฬิกา Casio

เมื่อมีการออกแบบสำหรับหน้าจอขนาดเล็กที่ ซึ่งที่คุณจะหลีกเลี่ยงไม่ได้คือ การออกแบบปุ่มที่เบียดเสียดกันบนอินเตอร์เฟซ ดังนั้นการ การเรียงปุ่มกด ให้ใช้งานได้ง่าย ก็จะสามารถทำให้ผู้ใช้งานยอมรับต่อ นาฬิกาเครื่องคิดเลขได้ หรือ อาจจะปฏิเสธการใช้งาน

ในการออกแบบหน้าจอการโทร ที่ต้องมีแป้นพิมพ์ตัวเลขที่เบียดชิดกัน จึงมีการออกแบบให้เป็นแบบเดียวกันทั่วโลก เพื่อให้ผู้ใช้ทั่วไป เข้าใจเหมือนกัน อีกทั้งยังชดเชยด้านขนาดของปุ่มกด เพราะฉะนั้น ถึงแม้ว่าปุ่มจะอยู่ใกล้กันมาก แต่ก็ออกแบบให้ง่ายต่อการสัมผัส

รูปที่ 2.10 ความแตกต่างในการจัดวางเครื่องมือ

การออกแบบของทั้งสอง แอปพลิเคชันจึงมีความคล้ายคลึงกัน แตกต่างแค่ การจัดวางเครื่องมือ และการวางปุ่มที่เรียงติดกัน

2.4 งานวิจัยที่เกี่ยวข้องกับการออกแบบ User Interface [7]

ความคิดในแง่ลบสำหรับขนาดเล็กระบบสัมผัสคาปาซิทีฟอินเตอร์เฟสหน้าจอสัมผัสที่ใช้งานการศึกษาสำหรับการป้อนข้อมูลงาน โดย Sarangi P. Parikh and Joel M. Esposito

บทคัดย่อ เทคโนโลยีระบบสัมผัสหน้าจอได้เป็นที่รู้จักอย่างแพร่หลายในเวทีสินค้าโลก ซึ่งในช่วงทศวรรษที่ผ่านมา มีการเสนอถึงประโยชน์หลายประการเช่น ซอฟต์แวร์อินเตอร์เฟสที่ปรับแต่งได้อย่างลงตัวโดยไม่จำเป็นต้องใช้เมาส์หรือคีย์บอร์ด แต่ถึงอย่างไรก็ตามก็ยังมีข้อเสียสำหรับอุปกรณ์เหล่านี้เกี่ยวกับข้อจำกัดในผู้ใช้บางราย อย่างที่รู้จักกันโดนทั่วไปคือทัชสกรีนมีความต้องการพื้นฐานเกี่ยวกับการใช้งานของผู้ใช้ คือ ผู้ใช้จะต้องดูที่หน้าจอของอุปกรณ์อินพุตเพื่อป้องกันไม่ให้กดปุ่มผิด ปัญหานี้เป็นสิ่งที่สำคัญโดยเฉพาะ สำหรับอุปกรณ์ตรวจจับนิ้วมือ ในโทรศัพท์เคลื่อนที่ระบบสัมผัสคาปาซิทีฟเช่น iPhone ซึ่งปุ่มที่มีขนาดเล็กจะทำให้มีความเสี่ยงสูงในการพิมพ์ผิด ในขณะที่ทำงานก่อนหน้านั้นมีการปรับปรุงอินเตอร์เฟสให้ดีขึ้นคือ มีข้อความเสียงหรือ การสั่น มีข้อคิดเห็นเกี่ยวกับ (การยืนยันการกดปุ่ม) ที่ได้รับการพิจารณา และในเอกสารนี้เรามีอินเตอร์เฟส ต้นแบบที่เข้าใจได้ง่าย

ความคิดเห็น เกี่ยวกับการสั่น ในแง่ลบคือ ข้อความจะถูกสร้างขึ้นโดยไม่ได้ป้อนข้อมูล บุคคลแรกที่เรามีการศึกษาการใช้งาน ที่เปรียบเทียบความคิดเห็นในทางบวกและทางลบ ของการรับรู้การสั่นจากการสัมผัส สำหรับการใช้งานการป้อนข้อมูลที่เป็นตัวเลข แต่ความแตกต่างในการทำงานไม่ได้มีผลมากเท่าที่ควรทางสถิติ แต่มีผลทางด้านจิตใจ เพราะฉะนั้นเราจึงให้ความใส่ใจ โดยใช้ในการสังเกตการณ์ของผู้ใช้งานจริง ซึ่งได้ให้คำแนะนำ เกี่ยวกับการใช้เสียงเตือน และ การรับรู้ในการสั่น การทดสอบผู้ใช้ตามรายการบันทึกการทดลอง แสดงให้เห็นว่า มีจุดสำคัญที่เห็นตรงกันว่าร้อยละยี่สิบสี่ของผู้ใช้กดปุ่มผิด เมื่อใช้เสียงเตือนอย่างเดียว ซึ่ง ความคิดเห็นเหล่านี้จะนำไปสู่การปรับเปลี่ยนจุดยุทธศาสตร์ที่สำคัญ

ดัชนีความคิดเห็น : การประเมิน/วิธีการใช้อุปกรณ์อินพุตและ กลยุทธ์อินเตอร์เฟสผู้ใช้

2.4.1 บทนำของงานวิจัย

ในช่วงหลายปีที่ผ่านมาเทคโนโลยีหน้าจอสัมผัสได้มีความก้าวหน้าอย่างรวดเร็วและมีเพิ่มขึ้นตามความนิยมสำหรับ นักพัฒนาและผู้บริโภค เช่นเดียวกันอุปกรณ์สัมผัสในแบบ ตู้เอทีเอ็ม ตามท้องตลาด แท็บเล็ต พีซี และ สมาร์ทโฟน ซึ่งในปัจจุบันมีแพร่หลายอยู่ในชีวิตประจำวัน ซึ่งพวกมันมีข้อดีหลายประการ

ประการแรก หน้าจอทัชสกรีน สามารถแทนที่หน้าจอ LCD คีย์บอร์ด คันบังคับทิศทางหรือเมาส์ เพื่อลดปริมาณทางกายภาพ ของผู้ใช้

ประการที่สอง สำหรับผู้ใช้มือใหม่จะใช้งานง่ายขึ้นเพราะแค่แตะที่อินเตอร์เฟสสามารถกำหนดค่าได้ในแบบหนึ่งต่อหนึ่ง การติดต่อระหว่าง ที่มีตัวเลือกการป้อน และป้อนข้อมูลจากผู้ใช้ใส่อุปกรณ์ เพื่อกำจัดปุ่มที่ไม่ค่อยได้ใช้ หรือแผงปุ่มควบคุมที่จำเป็นต้องใช้แบบปุ่มตัวเลข (เช่น การควบคุม Ctrl + Alt + Delete) สุดท้ายคืออินเตอร์เฟสเหล่านี้ก็จะสามารถจะปรับแต่งได้อย่างลงตัวโดยโปรแกรมเมอร์ จะเสนอให้ผู้ใช้กำหนดรูปแบบ เพื่อการออกแบบที่เหมาะสมกับขั้นตอนของงานแต่ละคน

ในทางกลับกัน ปัญหาที่เกี่ยวข้องกับการใช้งานหน้าจอแบบสัมผัส ได้รับการพูดถึงเพิ่มมากขึ้นอย่างชัดเจน พวกมันถูกนำไปใช้ ในการทำงานมากขึ้น อย่างแรกที่กลุ่มคนเหล่านี้ขาดคือการรับรู้สัมผัสปุ่มที่แท้จริง การพิมพ์แบบสัมผัสอย่างรวดเร็วแบบดั้งเดิม แป้นพิมพ์จะอำนวยความสะดวกโดยปุ่มโค้งพื้นผิวและมีความต้านทานต่อกลไกของปุ่ม ในทางตรงข้าม ปุ่มสัมผัสหน้าจอก็มีความต้องการด้านขนาด ความสนใจภาพอย่างต่อเนื่องเพื่อป้องกันไม่ให้กดปุ่มผิด วิธีการแก้ไขปัญหา ต่อมาคือการที่อุปกรณ์อินพุตและหน้าจอแสดงผลจะอยู่ที่เดียวกันซึ่งโดยทั่วไปแล้วอาจไม่ได้ตามหลักสรีรศาสตร์เพื่อการใช้งานได้ยาวนาน ข้อดีและข้อเสียที่เป็นการระบุถึงผู้ใช้งานทั้งคู่ ตัวอย่างเช่นคุณประโยชน์ของทัชสกรีนมีค่ากว่าจำนวนเงินที่ต้องเสียไป สำหรับผู้ใช้มือใหม่จะชอบการใช้งานที่ง่ายบน อินเตอร์เฟสแบบกำหนดเอง ซึ่งได้งานที่มี ในทำนองเดียวกันตามหลักสรีรศาสตร์ ข้อเสียของระบบสัมผัสหน้าจอจึงไม่เป็นปัญหามากนักเพราะเป็นระยะเวลาสั้นๆหรือต้องการให้การโต้ตอบเพียงไม่กี่อย่าง มีคุณสมบัติมากตามการสัมผัสที่คาดว่าจะมี

2.4.2 หลักการออกแบบ และตัวต้นแบบ

การศึกษาแก้ไขจาก iPhone ตามที่กล่าวไว้ในหัวข้อ 2.1 ข้อเสีย ของหน้าจอที่มีขนาดเล็กและมีปุ่มกดมาก ประสิทธิภาพ ในการทำงานก็จะลดลงไปด้วย โดยบางครั้งเราไม่สามารถเลือกเครื่องมือเหล่านั้นออกมาทำงานได้หรือเป็นอุปสรรค์ต้อง สัมผัสภายใต้พื้นที่จัดกัด (สำหรับการพิมพ์เล็บมือการใช้สไตลัส ฯลฯ)

เมื่อพิจารณาถึงความนิยม โทรศัพท์เคลื่อนที่ระบบจอสัมผัสรุ่นระบบสัมผัส คาปาซิทีฟ ที่ได้รับการออกแบบเพื่อใช้งานกับนิ้วมือ (ตัวอย่างเช่น iPhone Android, Motorola Milestone, Google Nexus One) และการเข้าใช้งานได้อย่างง่ายดายเพื่อไปยังกลุ่มประชากรขนาดใหญ่ของผู้ใช้เรา
จึงตัดสินใจที่จะทำการตรวจสอบกลยุทธ์การเพิ่ม ประสิทธิภาพ โดยเฉพาะ กลุ่มเป้าหมายที่ใช้อุปกรณ์เหล่านี้

เราเลือกที่จะแก้ไข iPhone 4 ตามเพื่อตั้งค่าต่อไปนี้: ตามหลักการการออกแบบ
การออกแบบที่ไม่จำเป็นต้องใช้ฮาร์ดแวร์เพื่อการใช้งานพิเศษใดๆ เราต้องการที่จะทำการตรวจสอบว่าอุปกรณ์โทรศัพท์เคลื่อนที่ที่มีอยู่แล้วเช่น iPhone ที่ไม่ได้ผ่านการแก้ไขซอฟต์แวร์ที่ผู้ผลิตกำหนดมาให้หรือเพิ่มฮาร์ดแวร์เพื่อการใช้งานพิเศษเข้าไป ที่ผู้บริโภคอื่นๆไม่มี

อุปกรณ์อินพุตไม่ควรต้องถูกสนใจมาก ผู้ใช้ที่ได้ประโยชน์มากที่สุด จากการปรับปรุง haptic ที่ว่างจากการดูที่ อุปกรณ์อินพุต การสังเกตการณ์อย่างง่าย ตัวอย่างเช่น นักบินเครื่องบินการควบคุมระยะไกลหรือ typists มืออาชีพแทบจะไม่ดูที่อุปกรณ์ของพวกเขา

ความคิดเห็นเกี่ยวกับการใช้งานโดยไม่ต้องมองหน้าจอโทรศัพท์แต่มีเสียงมาทดแทนจนน่ารำคาญ หลักฐานที่แสดงเรื่องราวได้คือผู้ใช้เลือกที่จะปิดคุณสมบัติการส่งเสียงการป้อนข้อความที่อยู่บน iPhone ของพวกเขาเพราะมันน่ารำคาญ

2.4.3 อินเตอร์เฟส

iPhone 4 (capacitive, 58 115.2 mm touch screen) ที่นำมาใช้ในการศึกษาครั้งนี้ ดังที่เห็นในรูป อินเตอร์เฟส แผงปุ่มกดตัวเลข 1 – 0 มีสไตล์ที่ได้รับการพัฒนาขึ้นสำหรับตั้งค่าเป็นครั้งแรกของ การทดลองเกี่ยวกับการป้อนข้อมูลที่เป็นตัวเลข

เราจะใช้สภาพแวดล้อม GUI การพัฒนาของบุคคลที่สามที่จะช่วยให้เรา สามารถควบคุมได้อย่างแม่นยำในการบันทึกข้อมูลและวางแผนผังได้ แต่ถึงอย่างไรก็ตามอินเทอร์เฟซนี้ ใกล้เคียงกับแผงปุ่มควบคุมแบบตัวเลขแบบมาตรฐานที่อยู่ในเงื่อนไขของการจัดเรียงปุ่ม ขนาดคีย์ (4 x 6 มิลลิเมตร) และช่อง (2 x 1 มิลลิเมตรในแนวตั้งและแนวนอน) ตรงกับรูป iPhone QWERTY คีย์บอร์ด อินเตอร์เฟส จะถูกล็อคในแนวตั้ง เหมือนกับอินเตอร์เฟสแผงปุ่มควบคุมการโทรออกพื้นเมืองที่ ปุ่มแต่ละปุ่ม จะสร้างตัวอักขระที่อยู่บนหน้าจอแสดงผลแบบแยกพื้นที่มีที่ "ลบ" หลักที่มีอยู่ซึ่งช่วยให้ การแก้ไข การกดปุ่ม "Enter" จะบันทึกผลและ ความก้าวหน้าในรายการถัดไป โปรดทราบว่าปุ่มตัวเลขที่ เป็น "เนื้อนุ่ม" พร้อมด้วยคีย์เพิ่มเติมโดยรอบบริเวณชายแดนของอินเตอร์เฟสที่ การสร้างอินเตอร์เฟสที่ปุ่มเป้าหมายทั้งหมดมีจำนวนเท่ากับที่อยู่ในบริเวณใกล้เคียงของปุ่ม(8) ในการออกแบบ ให้ปุ่มกดและไอคอนต่างๆ เน้นเพื่อให้ใช้งานที่สะดวกง่ายต่อการใช้งาน บนบริเวณที่พร้อมต่อการใช้งาน แต่ปัจจุบันก็มีอีกหลายความคิดเห็นเกี่ยวกับรูปแบบและการใช้งาน

iPhone 4 คือสมาร์ทโฟนระบบจอสัมผัสที่พัฒนาโดย บริษัท Apple ซึ่งเป็นรุ่นที่สี่ของ iPhone ต่อมาจากรุ่น iPhone 3Gs ถูกออกแบบเฉพาะสำหรับการใช้งานด้านการโทรศัพท์โดยมีสัญญาณภาพ (ในชื่อ Facetime) การอ่านหนังสืออิเล็กทรอนิกส์ การดูวิดีโอ ฟังเพลง เล่นเกมส์ และอินเทอร์เน็ต ถูกประกาศอย่างเป็นทางการเมื่อวันที่ 7 มิถุนายน พ.ศ. 2553 ในงาน WWDDC 2010 และขายครั้งแรกวันที่ 24 มิถุนายน พ.ศ. 2553 ในอเมริกา,อังกฤษ และอีกในหลายประเทศ โดยในประเทศไทยนั้นมีการขายครั้งแรกอย่างเป็นทางการใน วันที่ 24 กันยายน พ.ศ. 2553 โดยสามบริษัทใหญ่เป็นผู้จำหน่ายคือ เอไอเอส ดีแทค และทรูมูฟ iPhone 4 ทำงานด้วยระบบปฏิบัติการไอโอเอส (iOS) ซึ่งเป็นระบบเดียวกับที่ใช้ในiPhoneรุ่นก่อนหน้า รวมไปถึง iPad และ iPod Touchการทำงานหลักควบคุมโดยใช้ปลายนิ้วสัมผัส ข้อแตกต่างระหว่างiPhone 4 และiPhoneรุ่นก่อนหน้าได้แก่ การออกแบบโฉมใหม่ ที่มีการหุ้มขอบด้วยสเตนเลนสตีลไร้ฉนวนซึ่งทำหน้าที่เป็นเสาอากาศของเครื่อง ตัวเครื่องจะอยู่กึ่งกลางระหว่างกระจกอลูมิโนซิลิเกตชนิดพิเศษที่เพิ่มความแข็งแรงวางไว้สองด้านหน้าหลัง ภายในเครื่องมีซีพียู Apple A4 พร้อม ram 512 MB ของ eDRAM ซึ่งมีความเร็วเป็นสองเท่าของรุ่นก่อนหน้า และเร็วเป็นสี่เท่าของiPhoneรุ่นแรกสุด หน้าจอของ iPhone 4 มีขนาด 89 มิลลิเมตร (3.5 นิ้ว) โดยใช้แอลอีดีแบล็กลิตขนาดแสดงผลด้วยความละเอียด 960×640 พิกเซลซึ่งชื่อการค้าว่า เรตินาดิสเพลย์ (Retina Display)

ที่จะทำให้เราได้มากกว่าคือ สัญญาณการสั่นสะเทือน ที่มาจากผู้ใช้กระทำต่ออุปกรณ์ ส่งผลให้ตัวกระตุ้น iPhone LG Innotek miniPancake (ขนาดเส้นผ่าศูนย์กลาง 10 มิลลิเมตร หนา 3 มิลลิเมตร) ที่ยึดมั่นอยู่กับ ศูนย์กลางของด้านหลังของ iPhone เพื่อรองรับการทำงานเวลาเอียงหน้าจอ หรือ การเขย่า เป็นต้น

2.4.4 ข้อมูลการวัดประสิทธิภาพ

สำหรับการใช้งานการป้อนข้อความแต่ละครั้งเราตรวจสอบเพียงไม่กี่ครั้ง จากสิ่งที่เป็นไปได้ ในเชิงปริมาณและมาตรการในเชิงคุณภาพ ประสิทธิภาพการทำงานของแต่ละคนที่มีประสบการณ์และรวมถึง

1. ความแม่นยำ
2. การกดปุ่มต่อตัวอักษร
3. การกดปุ่มต่อวินาที (KSPS: Keystrokes Per Second)
4. ความเครียดทางจิตใจ ความสัมพันธ์ของผู้ใช้กับงานที่ทำ
5. การสำรวจความคิดเห็นทางออก

1. ความแม่นยำ โดยตรวจสอบพิจารณาว่าข้อมูลที่ป้อนเข้ามาถูกต้องหรือไม่ หากข้อความที่แสดงบนหน้าจอที่ตรงกับ ตรงกับข้อความที่ต้องการในเวลาที่เมื่อผู้ใช้กดปุ่ม" Enter " ความแม่นยำที่มีจำนวนของรายการที่ถูกต้องถูกแบ่งออกโดยจำนวนของรายการ 1.บันทึกไว้ด้วยว่าสำหรับวัดประสิทธิภาพ ผู้ใช้ต้องไม่กดผิดปุ่ม ซึ่งการแก้ไขที่เกี่ยวข้องคือการมองปุ่มป้อนก่อนมีการกดปุ่ม
2. การกดปุ่มต่อตัวอักษร ในหัวข้อนี้ใช้วัดจำนวนของการกดปุ่ม และแบ่งโดยจำนวนของตัวอักษรที่จำเป็นสำหรับการสร้างรายการได้อย่างถูกต้องและแม่นยำ (10 อักขระ ไม่รวม Enter) ซึ่งค่าใน อุดมคติคือ ตัวอย่างเช่น หมายเลข 10 หลักที่ มีการป้อนหมายเลขผิดพลาดและมีการแก้ไขจะใช้เวลา 1.2 วินาทีต่อการกดปุ่ม โดยการกดปุ่มต่อตัวอักษร ส่งผลต่อความถูกต้องของข้อมูล

   ตารางที่ 1 ค่าเฉลี่ย (ตามส่วนเบี่ยงเบนมาตรฐาน) ของ ประสิทธิภาพ สำหรับการป้อนข้อมูลตัวเลข งานเชิงบวกและลบเมื่อเทียบกับการสัมผัสที่มีการสั่น

3. การกดปุ่มต่อวินาที จำนวนการกดปุ่มต่อวินาที เป็นหนึ่งในการวัดความเร็วสำหรับการพิมพ์ ไม่ใช่เพียงกดตัวอักษรต่อหนึ่งวินาที แต่เป็นการกดปุ่มต่อวินาทีโดยไม่นับเมื่อมีการแก้ไขการพิมพ์ แต่จะนับเมื่อได้ข้อมูลที่ถูกต้อง
4. ความเครียดทางจิตใจ ความสัมพันธ์ของผู้ใช้กับงานที่ทำ NASA TLX มีการสำรวจความคิดเห็นให้ผู้เข้าร่วมประชุม โดยเมื่อผู้ใช้ทำงานงานหนึ่งเสร็จ พวกเขาจะมีมุมมองของพวกเขาโดยพวกเขาจะอธิบายเกี่ยวกับทัศนะคติและมุมมองของพวกเขา และนำความคิดเห็นของแต่ละคนที่ร่วมประชุมมาปรับใช้ เพื่อช่วยให้พวกเขาในการประเมินตนเอง เกี่ยวกับงานที่ทำความต้องการทางด้านจิตใจ เช่น จำเป็นต้องรักษาข้อมูลในหน่วยความจำ, เกิดความสับสน อินเตอร์เฟสความต้องการทาง กายภาพ เช่น การเคลื่อนไหวของร่างกาย (ตา มือ ศีรษะ) การเคลื่อนไหวของนิ้ว

สิ่งที่ส่งผลกระทบต่อจิตใจ เช่น การกดปุ่มไม่ถูกต้องอย่างต่อเนื่องที่อยู่ในบริเวณใกล้เคียง การกดปุ่มพลาด การทำงานที่ล่าช้า ทำให้เขว เช่น เปลี่ยนจุดสนใจ ในขณะที่พยายามจะทำงานที่ระบุไว้ความเหมาะสมของงาน เช่น งานด้านสร้างความสนุกสนาน งานคำนวณ ทางเลือกโดยรวม เช่น ต้องการมากสุด หรือไม่มีก็ได้

2.4.5 บทสรุปเกี่ยวกับความรู้และความสัมพันธ์ที่มี

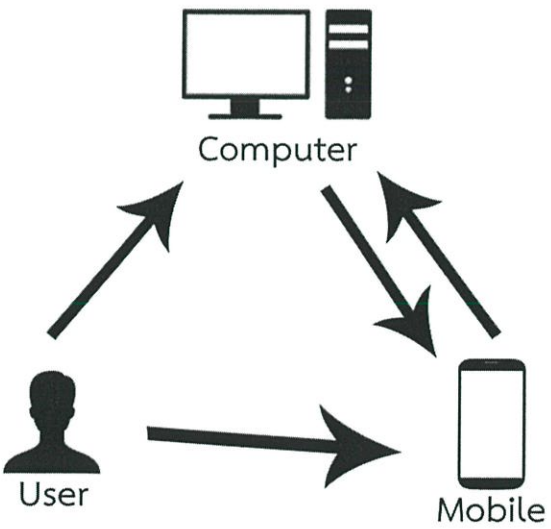
ในกรณีศึกษานี้ เราได้ทำการตรวจสอบผลกระทบเชิงลบจากความคิดเห็นของผู้ที่เคยได้ใช้ระบบสั่นในการพิมพ์งาน-อุปกรณ์โทรศัพท์เคลื่อนที่ระบบจอสัมผัสรุ่นเราเลือกที่จะเน้นไปที่การปรับปรุงซอฟต์แวร์แบบเรียบง่ายที่สามารถนำมาปรับใช้บนฮาร์ดแวร์ที่มีอยู่ เราเลือก iPhone 4 เป็นแพลตฟอร์มการทดสอบของเรา สืบเนื่องจาก 2 ปัจจัยคือเป็นที่นิยมและเพราะมันมีองค์ประกอบของการออกแบบที่ได้แสดงไว้ในก่อนหน้านี้ กล่าวคือ ตัวเก็บการตรวจจับ การพิมพ์ลายนิ้วมือ และปุ่มกดขนาดเล็ก การวัด ประสิทธิภาพ สำหรับการใช้งานการป้อนข้อมูลที่เป็นตัวเลขของเราเมื่อเปรียบเทียบความคิดเห็นในเชิงลบกับความคิดเห็นเกี่ยวกับระบบสั่น ในเชิงบวกที่เคยศึกษาเราสามารถให้คำตอบแก่คุณได้ถึงสองคำถามการวิจัยที่เกิดขึ้นออกเป็น 2 ส่วน

1. ความคิดเห็นในเชิงลบเกี่ยวกับระบบสั่น ผลที่ได้คือ ประสิทธิภาพ การพิมพ์ ที่อยู่ใกล้เคียงกับที่ได้รับโดยใช้ความคิดเห็นในเชิงบวก
2. ความคิดเห็นทางลบไม่ได้ผลในการปรับปรุงอย่างมีนัยสำคัญใดๆในการใช้งานของผู้ใช้ที่ระบุไว้ในการสำรวจที่ได้

บทที่ 3
การวิเคราะห์และออกแบบระบบ

3.1 วิเคราะห์และศึกษาระบบ

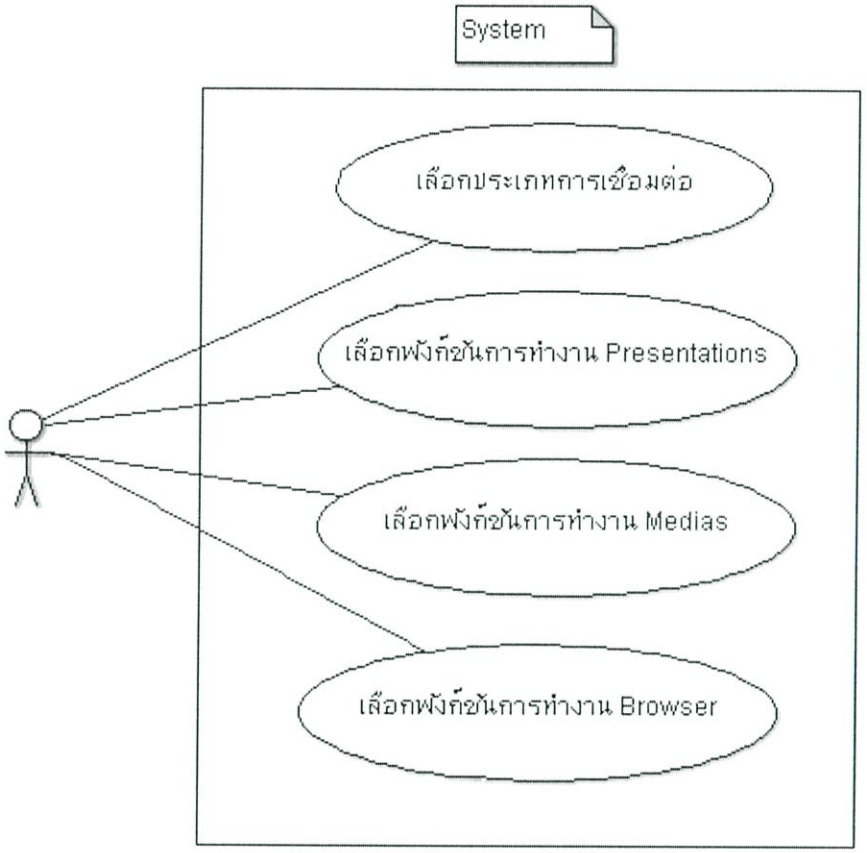
ในระบบนี้จะเป็นระบบที่ให้ผู้ใช้งานสามารถนำโทรศัพท์เคลื่อนที่ใช้ระบบปฏิบัติการ android สามารถนำมาใช้แทน Remote Presenter เพื่อการนำเสนองานได้อย่างมีประสิทธิภาพ โดยเริ่มแรก ผู้ใช้งานจะต้องเปิดแอปพลิเคชัน ที่อยู่บนคอมพิวเตอร์ และใช้โทรศัพท์เคลื่อนที่เพื่อเชื่อมต่อคอมพิวเตอร์กับโทรศัพท์เคลื่อนที่ และใช้สั่งงานต่างๆที่ช่วยในการนำเสนอ โดยการทำงานหลักจะแสดงดังรูป



รูปที่ 3.1 แสดงภาพรวมการทำงานของระบบ

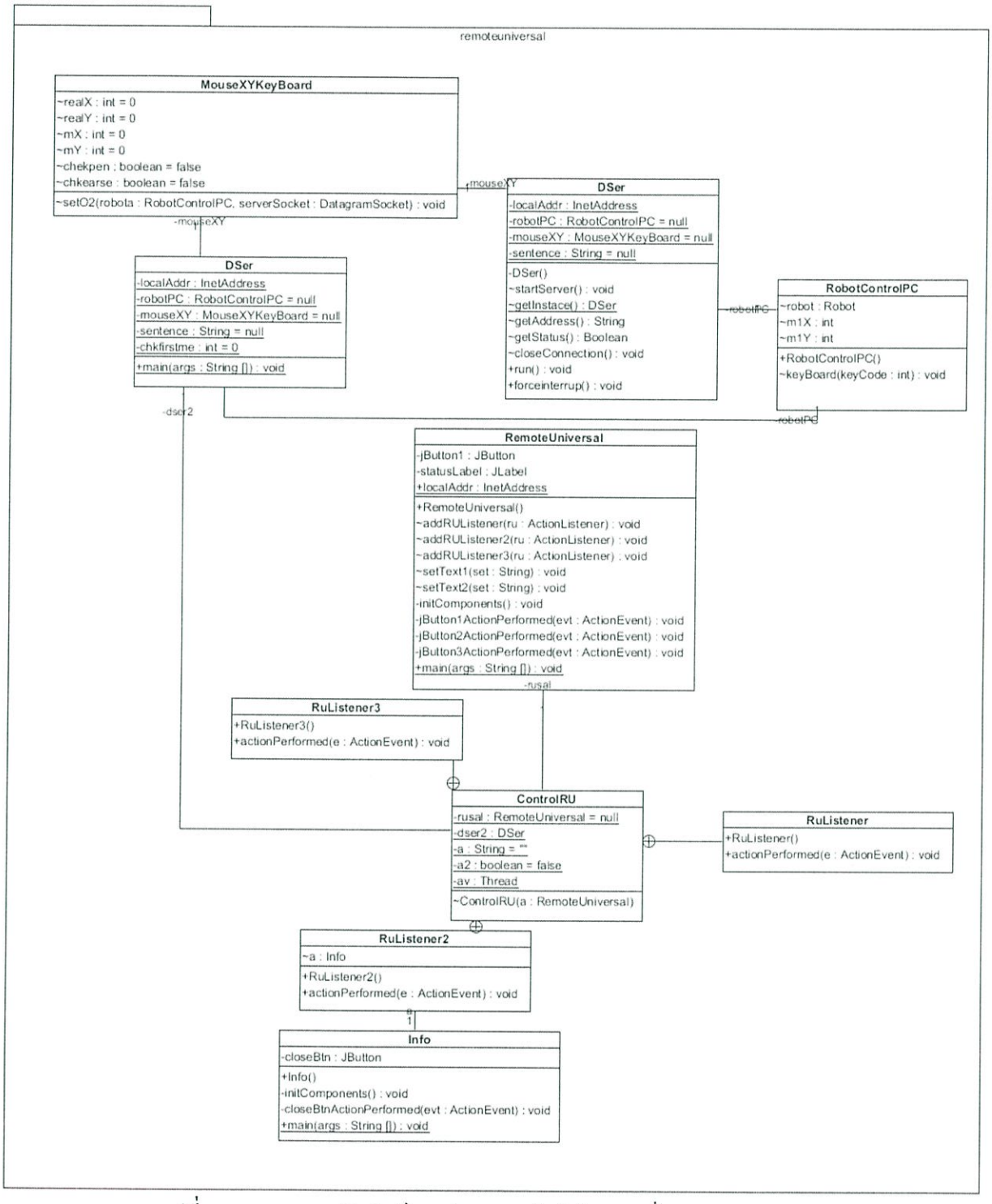
3.2 การออกแบบระบบ

3.2.1 Use Case Diagram ของระบบ

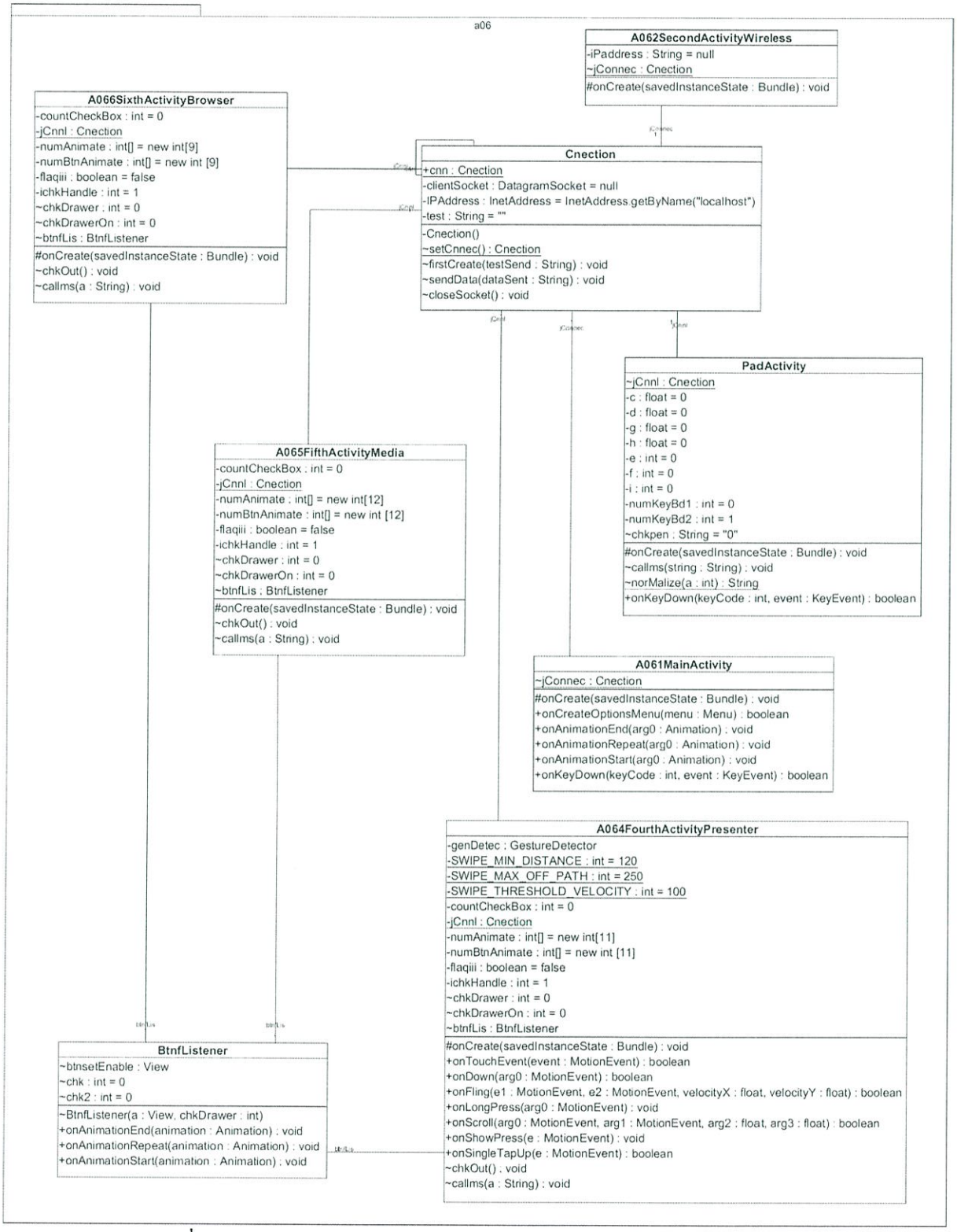


รูปที่ 3.2 แสดง Use Case Diagram ของระบบ

3.2.2 Class Diagram

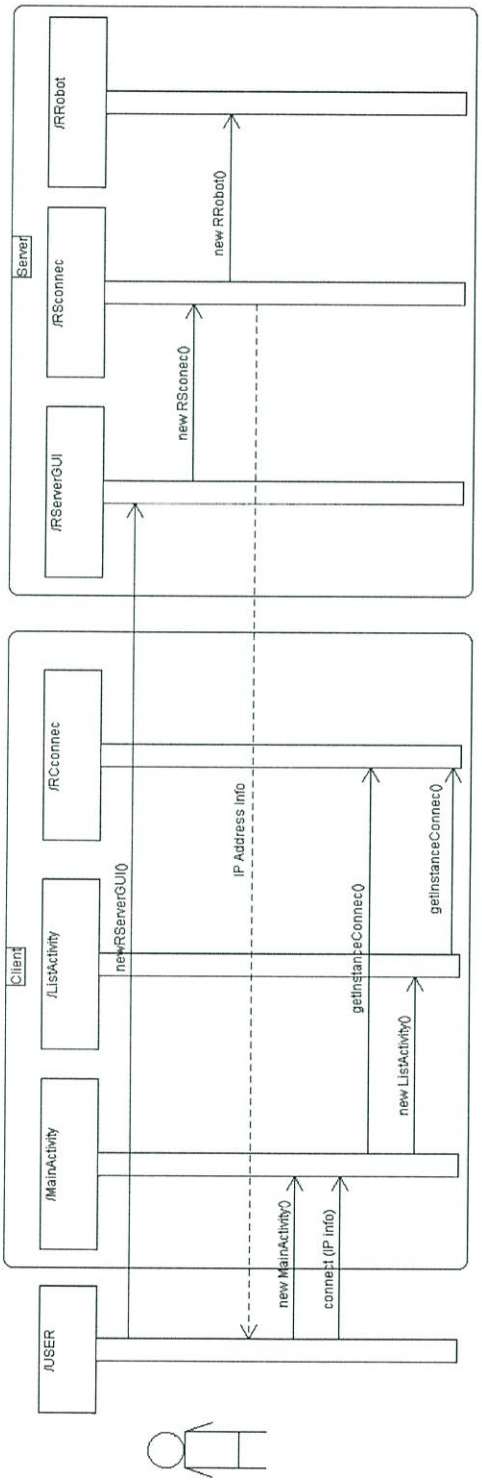


รูปที่ 3.3 แสดงความสัมพันธ์ระหว่าง Class ในโปรแกรมฝั่ง Server

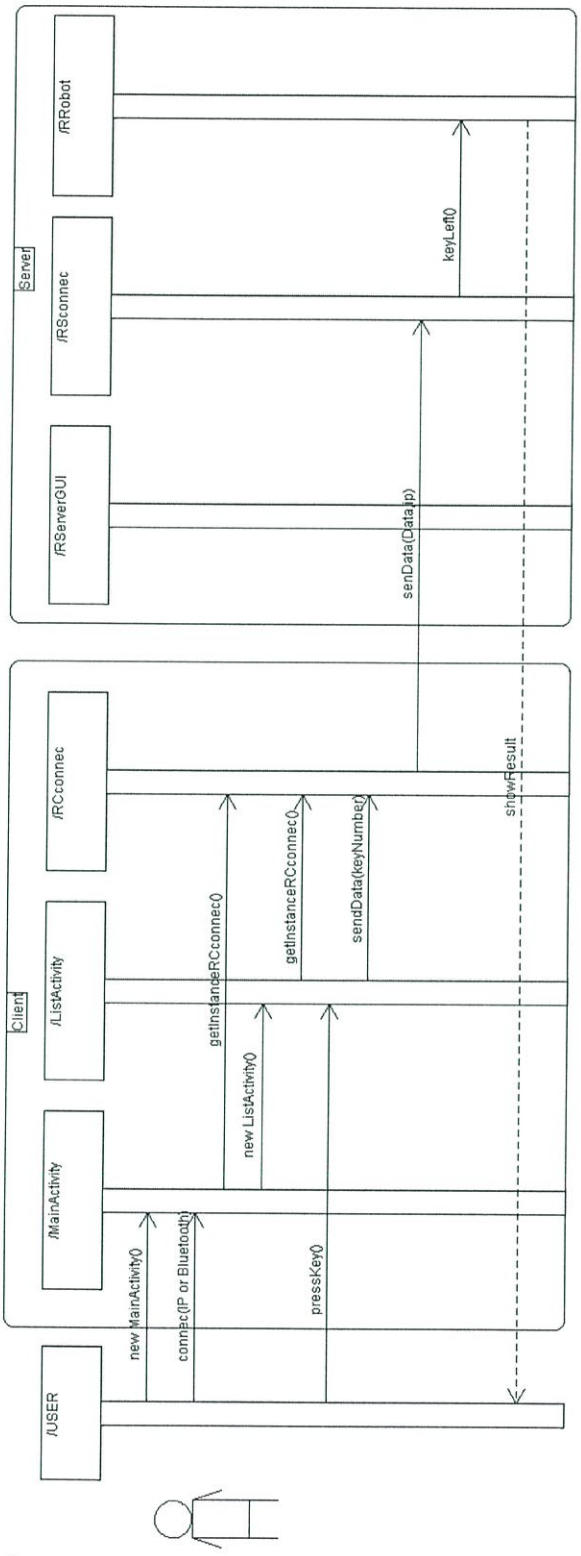


รูปที่ 3.4 แสดงความสัมพันธ์ระหว่าง Class ในโปรแกรมฝั่ง Client

3.2.3 Sequence Diagram



รูปที่ 3.5 Sequence Diagram แสดงถึงการเชื่อมต่อผ่านเครือข่ายไร้สาย



รูปที่ 3.6 แสดงการใช้ฟังก์ชันการทำงานตัวอย่าง คือกดปุ่มซ้าย

3.3 การออกแบบพัฒนาระบบชุดคำสั่งในการควบคุม

Java API Packages คือ Packages ของภาษาจาวา (Java) ที่บรรจุ Classes ต่าง ๆ เอาไว้ภายใน เพื่อให้ผู้เขียนโปรแกรมสามารถเรียกใช้งานได้ทันที โดยไม่จำเป็นต้องเขียนโปรแกรมในส่วนนั้น ๆ ทั้งหมด โดยสามารถเรียกเป็นชื่ออื่น ๆ ได้เช่น Java Application Programming Interface (Java API) หรือ Java Class Library ซึ่งชุดคำสั่งในการควบคุมอุปกรณ์ อินพุต ต่างๆคือ java.awt.Robot

Java API Packages (Reference Java SE 6.0)

Packages	Description
java.applet	Java Applet Packages บรรจุ Class สำหรับการสร้าง Applet
java.awt	Java Abstract Window Toolkit Package บรรจุ Class สำหรับการจัดการ GUIs ใน Java 1.0 และ 1.1
java.awt.event	Java Abstract Window Toolkit Event Package บรรจุ Class สำหรับจัดการ Event ของ AWT
java.io	Java Input/output Package บรรจุ Class สำหรับทำงาน Input และ Output ข้อมูล
java.lang	Java Language Package บรรจุ Class สำคัญต่าง ๆ ของการทำงานโปรแกรม และการ Compiler
java.net	Java Networking Package บรรจุ Class สำหรับการทำงานด้าน Computer Network
java.text	Java Text Package บรรจุ Class สำหรับการทำงานกับข้อความ, ตัวเลข, ตัวอักษร และอื่น ๆ
java.util	Java Utilities Package บรรจุ Class สำหรับการทำงานต่าง ๆ เช่น วันที่, เวลา, random number, tokens
java.swing	Java Swing GUI Components Package บรรจุ Class สำหรับการจัดการ GUIs ด้วย Java's Swing
java.swing.event	Java Swing Event Package บรรจุ Class สำหรับจัดการ Event ของ Swing

ตารางที่ 3.1 แสดง API Package ในภาษาจาวา

3.3.1 การควบคุมคีย์ต่างๆในคอมพิวเตอร์

java.awt.Robot Robot Classเป็นคลาสที่ทำให้สามารถควบคุมคีย์ต่างๆ ได้โดย การใช้ method keyPress(int keyCode) ซึ่งในภาษาจาวาคือการเรียก native key นั้นขึ้นมาทำงานโดยเป็นการกด 1 ครั้ง method ที่ใช้งานทั้งหมดจาก Robot Class มีดังนี้

Constructor and Description
Robot() Constructs a Robot object in the coordinate system of the primary screen.
Robot(GraphicsDevice screen) Creates a Robot for the given screen device.

ตารางที่ 3.2 แสดง Constructor ในคลาส Robot

Modifier and Type	Method and Description
Color	getPixelColor(int x, int y) ใช้เพื่อรับค่าสีจากตำแหน่งของ Pixel บนหน้าจอ
void	keyPress(int keycode) ใช้เพื่อกดคีย์ใดใดบนคีย์บอร์ด
void	keyRelease(int keycode) ใช้เพื่อปล่อยคีย์ใดใดบนคีย์บอร์ด
void	mouseMove(int x, int y) คำสั่งเพื่อให้เลื่อนเมาส์ไปยังตำแหน่งที่ต้องการ
void	mousePress(int buttons) การคลิกปุ่มบนเมาส์ใดใด
void	mouseRelease(int buttons) การปล่อยปุ่มบนเมาส์ใดใด
void	mouseWheel(int wheelAmt) การใช้เพื่อ scroll เมาส์

ตารางที่ 3.3 แสดง Method ที่มีในคลาส Robot

3.4 โครงร่าง ส่วนของผู้ใช้งานบนโทรศัพท์เคลื่อนที่

ทำการออกแบบในส่วนติดต่อกับผู้ใช้ ให้มีการใช้งานที่ง่าย และเหมาะสม โดยคำนึงถึงหลักในการออกแบบ User Interface ที่ดี ดังนี้

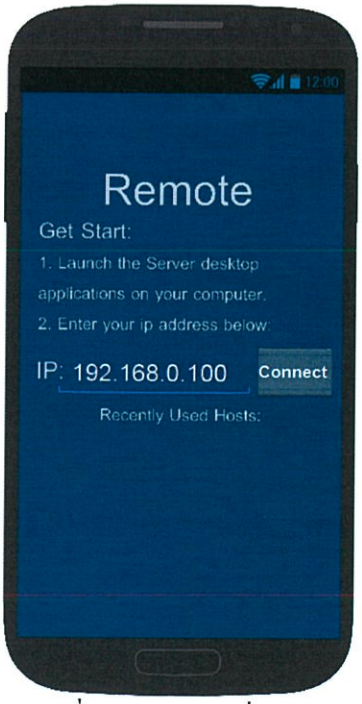
การออกแบบ User Interface มีวัตถุประสงค์หลักคือ

1) มีลักษณะเป็นหน้าต่างและคำสั่งในการทำงานคล้ายโปรแกรมทั่วๆไปที่ Run บน Windows เพราะผู้ใช้งานคอมพิวเตอร์ส่วนใหญ่คุ้นเคยกับการใช้ Windows อยู่แล้วทำให้ผู้ใช้สามารถใช้โปรแกรมได้สะดวกมากขึ้น
2) การรับคำสั่งจากผู้ใช้เน้นที่การกดปุ่มที่ได้เตรียมไว้ให้ โดยปุ่มจะมีข้อความหรือรูปภาพที่สื่อความหมายของคำสั่งให้ได้มากที่สุด ผู้ใช้เพียงแต่คลิกเมาส์ที่ปุ่มคำสั่งก็สามารถใช้งานโปรแกรมได้โดยไม่ต้องจดจำคำสั่งและพิมพ์คำสั่งเอง
3) เพื่อความสะดวกในการใช้งาน สามารถเลือกรายการใช้งานได้ตามจุดประสงค์ของผู้ใช้
4) ผู้ใช้จะป้อนหมายเลข IP Address ที่ใช้ติดต่อกับ คอมพิวเตอร์ โดยจะแสดงได้ที่ช่องข้อความ
5) ใช้รูปแบบ สีสัน สวยงามดึงดูดใจให้น่าใช้

Universal Usability ควรคำนึงถึงเรื่องต่อไปนี้

1) ความหลากหลายของผู้ใช้งานทั้งทางกายภาพและสภาพแวดล้อม
2) บุคลิกของผู้ใช้ที่แตกต่างกัน /ความต่างระหว่างบุคคล
3) ความแตกต่างของปัญญาและความสามารถในการรับรู้
4) ความหลากหลายทางเชื้อชาติและวัฒนธรรม
5) อายุของผู้ใช้งาน
6) การปรับให้เข้ากับซอฟท์แวร์และฮาร์ดแวร์

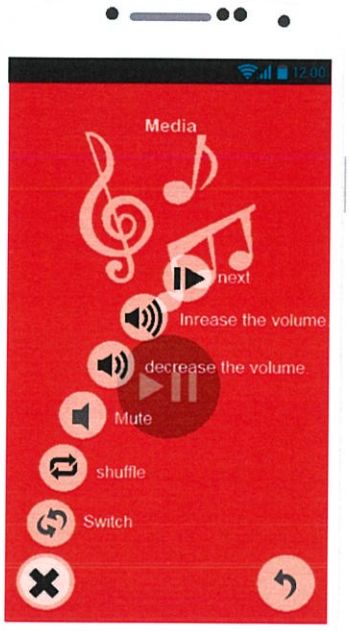
รูปที่ 3.7 หน้าแรก ให้ผู้ใช้เลือกประเภทการเชื่อมต่อ



รูปที่ 3.8 แสดงการเชื่อมต่อ

รูปที่ 3.9 แสดงหน้าหลักในการทำงาน

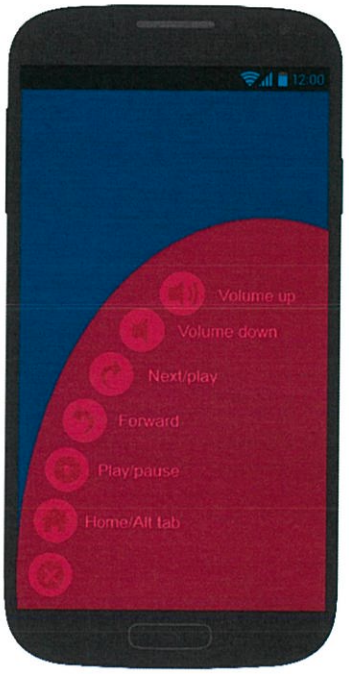
รูปที่ 3.10 หน้าฟังก์ชันการทำงานในการ นำเสนองาน โดยผู้ใช้ต้องเปิดโปรแกรม Microsoft PowerPoint ขึ้นมาก่อน



รูปที่ 3.11 หน้าฟังก์ชันการทำงานในการ เล่นสื่อมัลติมีเดีย โดยผู้ใช้ต้องเปิดโปรแกรมเล่นสื่อขึ้นมาก่อน เช่น Window Media Player

รูปที่ 3.12 หน้าฟังก์ชันการทำงานในการ เล่นเว็บบราวเซอร์ โดยผู้ใช้ต้องเปิด เว็บบราวเซอร์ขึ้นมาก่อน เช่น Internet Explorer, Google Chrome

ซึ่งการออกแบบทั้งหมดจะอ้างอิงได้กับหลักการ พื้นที่ที่ถูกสัมผัสได้ง่าย (Active Area Touch) ที่ให้ผู้ใช้สามารถกดแอปพลิเคชันขณะจับโทรศัพท์เคลื่อนที่ของผู้ใช้งานด้วยมือเพียงข้างเดียวได้ และปุ่มฟังก์ชันของการทำงานจะถูกเรียงไว้ในพื้นที่ Active Area Touch ดังรูป



รูปที่ 3.13 แสดงพื้นที่ Active area touch กับการออกแบบ

บทที่ 4
ผลการวิจัยและอภิปรายผล

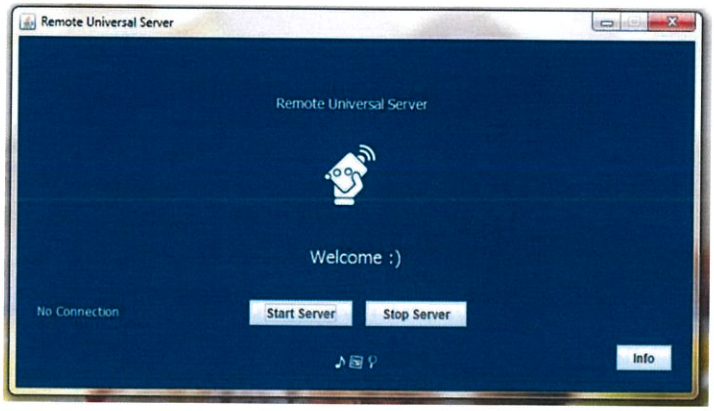
4.1 รายละเอียดของระบบ

การพัฒนาแอปพลิเคชันผ่านสมาร์ทโฟนระบบปฏิบัติการแอนดรอยด์ ให้สามารถใช้งานได้ง่ายและมีประสิทธิภาพ โดยมีจุดประสงค์เพื่อสามารถใช้ทรัพยากรที่มีอยู่ให้เกิดประโยชน์ และยังสามารถรองรับผู้ใช้งานในกรณีที่ไม่มีอุปกรณ์ โดยจัดทำแอปพลิเคชันในสมาร์ทโฟนที่ใช้ระบบปฏิบัติการแอนดรอยด์ ที่ติดต่อกับเครื่องคอมพิวเตอร์ เพื่อควบคุมเครื่องคอมพิวเตอร์ในงานต่างๆ โดยพัฒนาโปรแกรมนี้ด้วยภาษาจาวา (Java Programming Language) และศึกษาทฤษฎีการออกแบบส่วนที่ติดต่อกับผู้ใช้ ให้สามารถใช้งานได้ง่ายและยืดหยุ่นต่อความต้องการของผู้ใช้แต่ละประเภท พร้อมทั้งศึกษาฟังก์ชันหลักในการทำงานของแต่ละโปรแกรมเพื่อหาฟังก์ชันการทำงานที่จำเป็น ซึ่งทางระบบมีลักษณะเป็นโปรแกรมประมวนผลทางฝั่งเซิร์ฟเวอร์ (Server) ทำหน้าที่เชื่อมต่อคอมพิวเตอร์กับสมาร์ทโฟน โดยทางฝั่งผู้ใช้ไคลเอนต์ (Client) จะเรียกใช้งานผ่านหมายเลขประจำเครื่องคอมพิวเตอร์แต่ละเครื่องในระบบเครือข่ายที่ใช้โพรโทคอลแบบ TCP/IP (IP Address) โดยเชื่อมต่อภายในไร้สาย (Wireless LAN) เดียวกัน ส่วนทางฝั่งเซิร์ฟเวอร์ (Sever) ต้องติดตั้งแอปพลิเคชันบนเครื่องบริการ (Server) เพื่อสร้างการเชื่อมต่อ และควบคุมการทำงานผ่านรีโมท (Remote) ได้
โครงสร้างของระบบ

4.2 ผลการดำเนินงาน

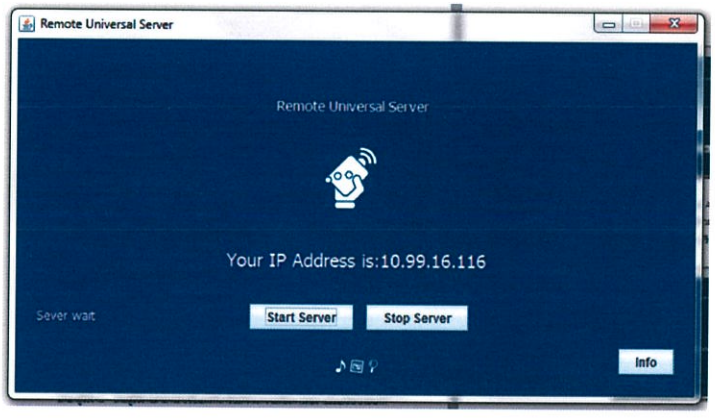
สิ่งที่ได้ทำการพัฒนาเพิ่มเติมจากการออกแบบฟังก์ชันการใช้งาน โดยสร้างความยืดหยุ่นให้แก่ผู้ใช้งาน โดยผู้ใช้งานสามารถปรับแต่งฟังก์ชันให้ตรงความต้องการต่องานในแต่ละประเภทซึ่งโครงสร้างของระบบแบ่งออกเป็น 2 ส่วนหลัก

1. ส่วนของเซิร์ฟเวอร์ คือ คอมพิวเตอร์
2. ส่วนของไคลเอนต์ คือ สมาร์ทโฟน



รูปที่ 4.1 Java Application บนฝั่งเซิร์ฟเวอร์

ผู้ใช้จะต้องเปิดแอปพลิเคชัน บนฝั่งเซิร์ฟเวอร์ก่อนเพื่อสร้างการเชื่อมต่อผ่านหมายเลขประจำเครื่องคอมพิวเตอร์แต่ละเครื่องในระบบเครือข่ายที่ใช้โพรโทคอลแบบ TCP/IP (IP Address)



รูปที่ 4.2 หมายเลข IP Address เพื่อใช้ติดต่อระหว่างเซิร์ฟเวอร์และไคแอนดรอยด์

โดยผู้ใช้จะต้องมีแอปพลิเคชันบนสมาร์ทโฟนเพื่อติดต่อกับฝั่งเซิร์ฟเวอร์

Universal Remote

จากการสร้าง Mobile Application ชื่อว่า Universal Remote แสดงผลออกมาดังนี้

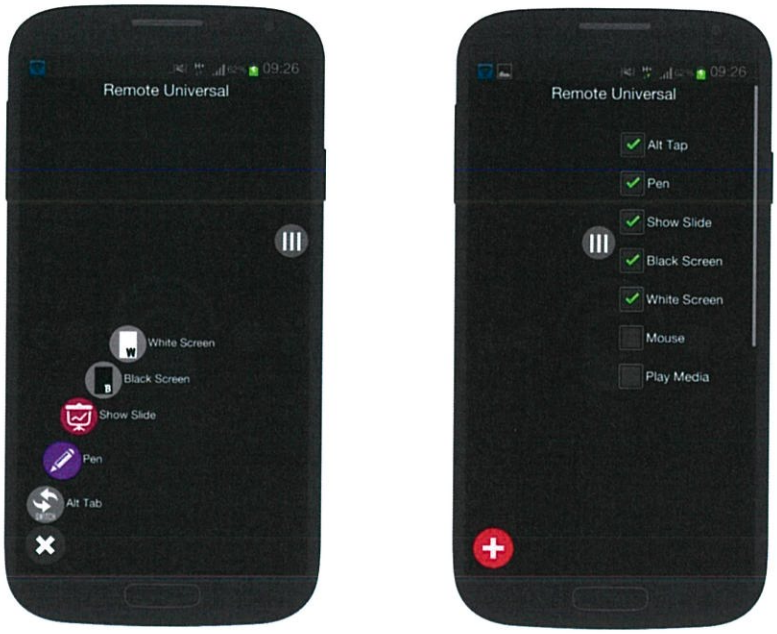
รูปที่ 4.3 หน้าก่อนเข้าสู่ Main Menu ซึ่งจะแสดงไอคอนประเภทของการเชื่อมต่อ และ ไอคอนเพื่อนำเข้าสู่หน้าการเชื่อมต่อ โดยกดปุ่มไอคอน เพื่อเข้าสู่หน้า ใส่ IP Address

รูปที่ 4.4 หน้าการเชื่อมต่อ โดยผู้ใช้ต้องเปิดแอปพลิเคชัน Universal Remote Server บนเครื่องคอมพิวเตอร์ เพื่อนำ IP Address มาใส่ในแอปพลิเคชันบน Smart Phone

รูปที่ 4.5 หน้าให้ผู้ใช้เลือกความถนัดของการใช้มือ โดยมีความถนัดมือซ้ายและขวา เพื่อให้ผู้ที่มีความถนัดแตกต่างกัน สะดวกต่อการใช้งานในการกดปุ่ม

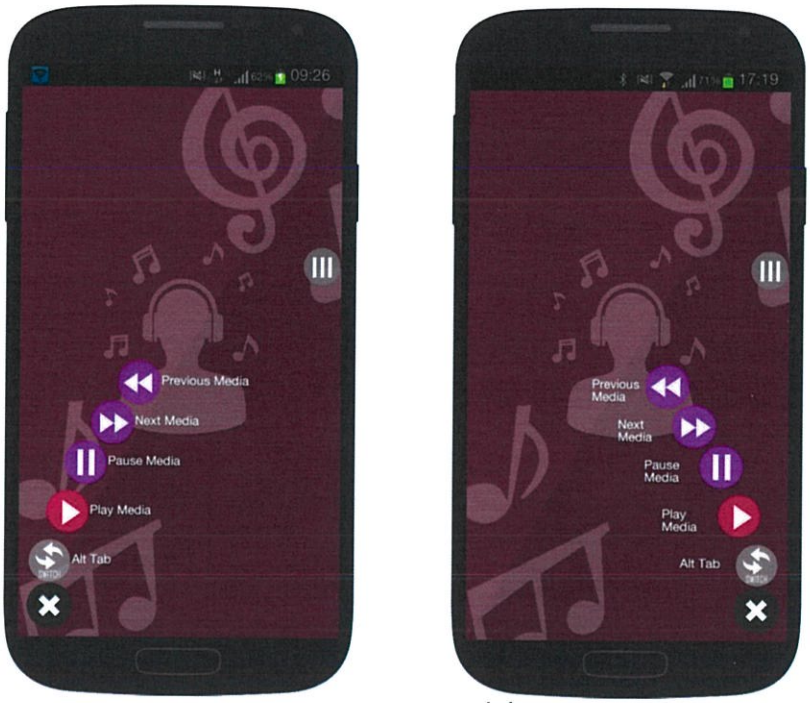
รูปที่ 4.6 ตัวอย่างการจัดเรียงไอคอนสำหรับผู้ใช้ที่ถนัดมือซ้าย (ภาพ ซ้าย) และถนัดมือขวา (ภาพขวา)

รูปที่ 4.7 หน้า Main Menu ซึ่งมีประเภทการใช้งานให้เลือกด้วยกัน 3 ประเภท ได้แก่ โหมดการนำเสนอ โหมดมัลติมีเดีย และโหมดเว็บบราวเซอร์ ผู้ใช้สามารถเลือกประเภทการใช้งานตามความเหมาะสม

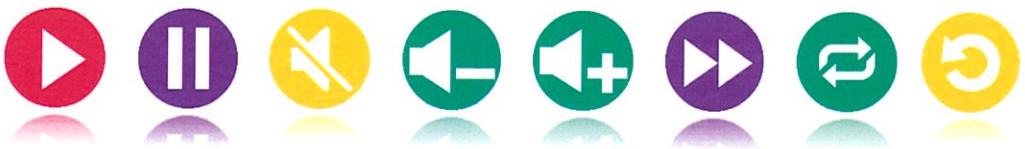


รูปที่ 4.8 หน้า โหมดการนำเสนอ จะมีฟังก์ชันการใช้งานหลักที่เกี่ยวกับการใช้งานนำเสนอ ซ่อนอยู่ โดยผู้ใช้สามารถ กดไอคอนรูปบวก เพื่อ โชว์ปุ่มฟังก์ชันหลัก และถ้าผู้ใช้ต้องการปรับแต่ง สามารถ เรียงปุ่มและนำปุ่มฟังก์ชัน อื่นๆ สามารถเลือกไอคอนบริเวณข้างหน้าจอ (ซึ่งผู้ใช้ต้องเปิดโปรแกรม Microsoft Power Point บนเครื่องคอมพิวเตอร์ก่อน)

รูปที่ 4.8.1 ไอคอนและฟังก์ชันการใช้งาน ในโหมดการนำเสนอ (เรียงจากซ้ายไปขวา สลับโปรแกรม, ทำให้หน้าการนำเสนอเป็นสีดำ หรือ ขาว เข้าสู่การนำเสนอ เปลี่ยนพ้อยเตอร์เป็นปากกา เมาส์ แสง สว่าง)

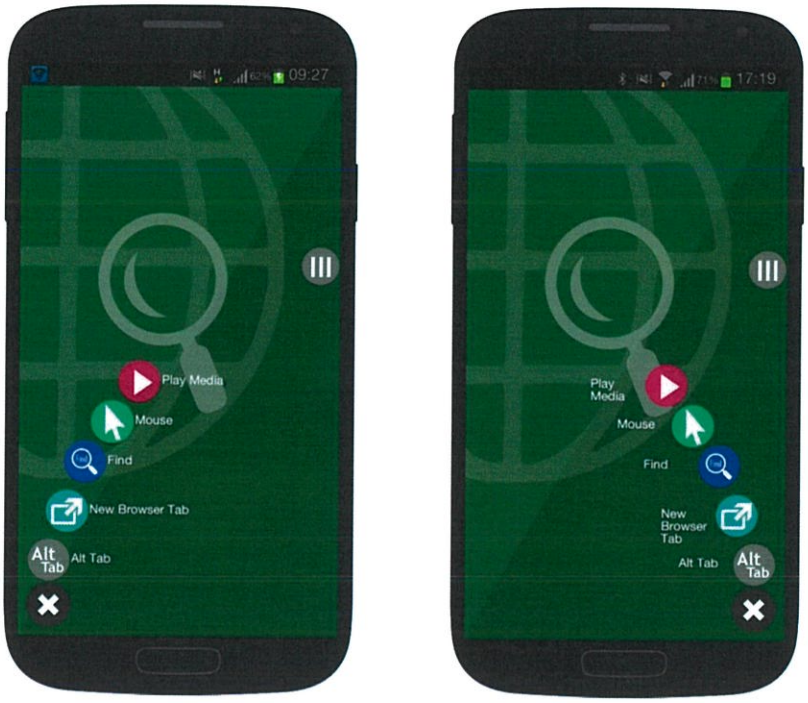


รูปที่ 4.9 หน้า โหมดมัลติมีเดีย จะมีฟังก์ชันการใช้งานหลักที่เกี่ยวกับการใช้งานมัลติมีเดีย ซ่อนอยู่ โดยผู้ใช้สามารถ กดไอคอนรูปบวก เพื่อ โชว์ปุ่มฟังก์ชันหลัก และถ้าผู้ใช้ต้องการปรับแต่ง สามารถ เรียงปุ่มและนำปุ่มฟังก์ชัน อื่นๆ สามารถเลือกไอคอนบริเวณข้างหน้าจอ(ซึ่งผู้ใช้ต้องเปิดโปรแกรม ที่เกี่ยวข้องกับมัลติมีเดีย เช่น Window Media Player เป็นต้น)



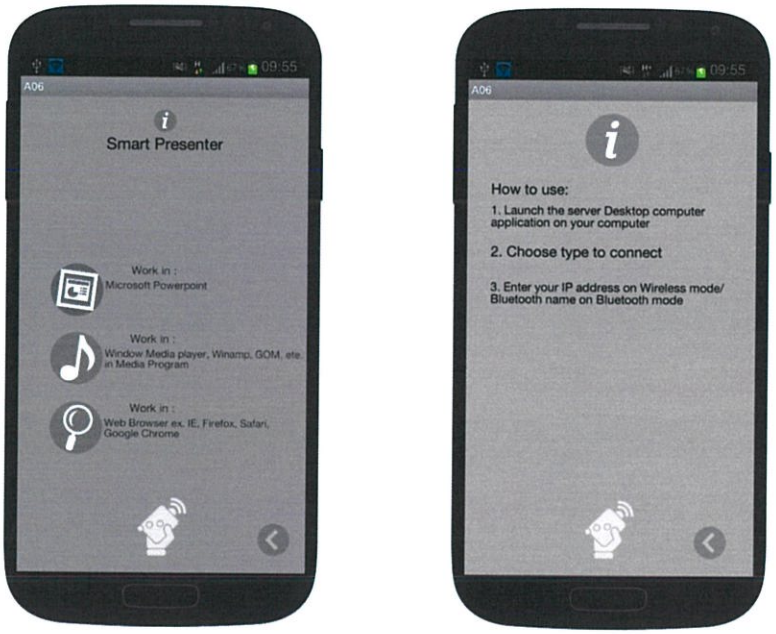
รูปที่ 4.9.1 ไอคอนและฟังก์ชันการใช้งาน ในโหมดมัลติมีเดีย (เรียงจากซ้ายไปขวา เล่น, หยุด, ปิดเสียง, ลดระดับเสียง, เพิ่มระดับเสียง, ถัดไป, สุ่ม, เล่นซ้ำ)

รูปที่ 4.9.2 ฟังก์ชันในการเพิ่มเสียงลดเสียง ผู้ใช้สามารถ เพิ่มเสียงลดระดับเสียงได้ โดยลดเสียงด้วยการกกดไอคอนฝั่งซ้าย และเพิ่มระดับเสียงด้วยไอคอนฝั่งขวา

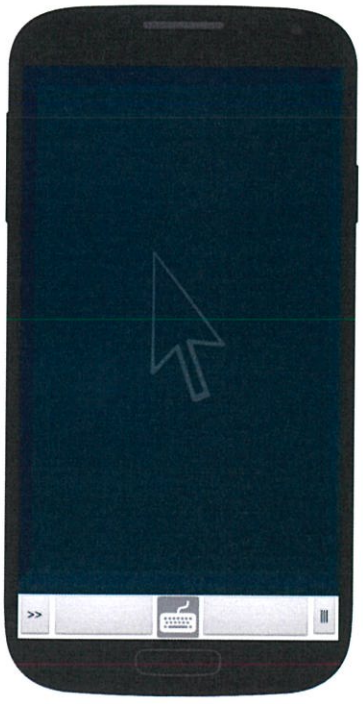


รูปที่ 4.10 หน้า โหมดเว็บบราวเซอร์ จะมีฟังก์ชันการใช้งานหลักที่เกี่ยวกับการใช้งานเมาส์และคีย์บอร์ดบนเว็บบราวเซอร์ ซ่อนอยู่ โดยผู้ใช้สามารถ กดไอคอนรูปบวก เพื่อ โชว์ปุ่มฟังก์ชันหลัก และถ้าผู้ใช้ต้องการปรับแต่ง สามารถ เรียงปุ่มและนำปุ่มฟังก์ชัน อื่นๆ สามารถเลือกไอคอนบริเวณข้างหน้าจอ (ซึ่งผู้ใช้ต้องเปิดเว็บบราวเซอร์ขึ้นมาก่อน)

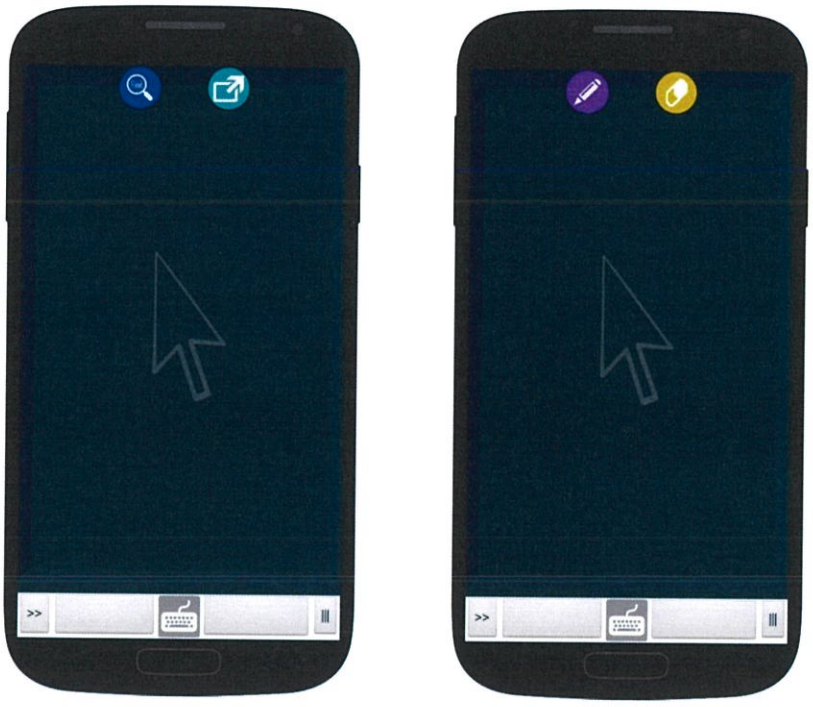
รูปที่ 4.10.1 ไอคอนและฟังก์ชันการใช้งาน ในเว็บบราวเซอร์ (เรียงจากซ้ายไปขวา ค้นหาคำหรือวลี, เปิดหน้าต่างใหม่, ควบคุมเมาส์, สลับหน้าจอ)



รูปที่ 4.11 หน้า Information แนะวิธีการใช้งานเบื้องต้น



รูปที่ 4.12 หน้า การใช้งานเมาส์และคีย์บอร์ด เพื่อควบคุมเมาส์บนหน้าจอคอมพิวเตอร์



รูปที่ 4.13 หน้า การใช้งานเมาส์และคีย์บอร์ด เพื่อควบคุมเมาส์บนหน้าจอคอมพิวเตอร์ ในโหมดการเล่นเว็บบราวเซอร์ (ฝั่งซ้าย) มีฟังก์ชันเพื่อหากลุ่มคำ หรือวลี และ เปิดหน้าต่างใหม่ในเว็บบราวเซอร์ และโหมดการนำเสนองาน (ฝั่งขวา) มีฟังก์ชันเปลี่ยนพ้อยเตอร์ให้เป็นปากกา และยางลบเพิ่มลบปากกา

บทที่ 5

สรุปผลการพัฒนาระบบและข้อเสนอแนะ

ผลจากการศึกษา

5.1 บทสรุป

ในการศึกษาและพัฒนาปัญหาพิเศษจุดมุ่งหมายหลัก คือ การพัฒนาแอปพลิเคชันบนสมาร์ทโฟนระบบปฏิบัติการแอนดรอยด์ ให้สามารถนำมาใช้งานได้จริง และสามารถพัฒนาต่อยอดในการออกแบบส่วนติดต่อกับผู้ใช้ ให้มีความน่าสนใจมากยิ่งขึ้นโดยการใช้วิธีการเฉพาะสำหรับการเรียกใช้ระบบปฏิบัติการหรือแอปพลิเคชันอื่นๆ หรือชุดคำสั่งคอมพิวเตอร์ซึ่งทำหน้าที่เชื่อมต่อการทำงานระหว่างแอปพลิเคชันกับระบบปฏิบัติการ (Application Program Interface หรือ API) และการจัดการการเชื่อมต่อ (Connection) โดยการพัฒนาและออกแบบแอปพลิเคชันนี้ จะให้ความสำคัญต่อการออกแบบและการใช้งาน เพื่อให้ผู้ใช้สามารถใช้งานได้สะดวก และปรับแต่งตามประเภทการใช้งานได้ เช่น การนำเสนองาน การเล่นสื่อ และการใช้งานเว็บบราวเซอร์ โดยใช้การออกแบบสำหรับระบบสัมผัส (Designing for Touch) เพราะการออกแบบแอปพลิเคชันบนโทรศัพท์มือถือที่ดี ต้องทำมากกว่าการใส่ฟังก์ชันการทำงานมากๆลงในหน้าจอ แต่ต้องสร้างในหนทางสำหรับการทำงานของนิ้วมือ โดยศึกษาสภาพการทำงานที่มีความสัมพันธ์ระหว่างผู้ใช้และสิ่งแวดล้อม การยศาสตร์ (Ergonomics) ได้มีการออกแบบหรือปรับปรุงให้มีความเหมาะสมกับผู้ใช้อย่างไร เพื่อป้องกันปัญหาที่อาจมีผลกระทบต่อจิตใจในการทำงาน และสามารถเพิ่มประสิทธิภาพการทำงานได้อีกด้วย หรืออาจกล่าวอีกนัยหนึ่งได้ว่าเพื่อทำให้งานที่ต้องปฏิบัติดังกล่าว มีความเหมาะสมกับผู้ปฏิบัติงาน แทนที่จะบังคับให้ผู้ปฏิบัติงานต้องทนฝืนปฏิบัติงานนั้น ๆ ตัวอย่างเช่น การเพิ่มระดับความสูงของโต๊ะทำงานให้สูงขึ้น เพื่อพนักงานจะได้ไม่ต้องก้มโน้มตัวเข้าใกล้ชิ้นงาน เป็นต้น และสามารถใช้งานทุกฟังก์ชันได้อย่างมีประสิทธิภาพ

5.2 ข้อเสนอแนะ

การออกแบบและพัฒนาแอปพลิเคชัน ออกมาให้ตรงตามความต้องการของระบบยังมีข้อจำกัดบางส่วน แต่ยังสามารถเพิ่มประสิทธิภาพของระบบ ที่สามารถจะพัฒนาต่อได้อีกตาม ข้อเสนอแนะ ดังต่อไปนี้

1) ในอนาคตระบบการเชื่อมต่อ (Connections) มีแนวโน้มในการใช้งานมากขึ้น และเพื่อความสะดวกของผู้ใช้จึงควรเพิ่มรูปแบบการเชื่อมต่อ ให้มีความหลากหลายเพื่อเป็นช่องทางในการใช้งานแอปพลิเคชัน
2) ในส่วนของการออกแบบการเคลื่อนไหวของอนิเมชัน ยังมีรูปแบบที่ยังไม่คงตัว ซึ่งอาจเพิ่มความลื่นไหลของการใช้งานอนิเมชันได้มากกว่านี้
3) ในส่วนของการใช้งานอาจเพิ่มทางเลือกให้กับผู้ใช้ได้มากกว่านี้ เช่น เลือกสีพื้นหลังได้เหมาะสม เลือกสีของเซตไอคอน เป็นต้น
4) การนำแอปพลิเคชัน นี้ไปพัฒนาต่อนั้น นอกจากจะมีความรู้ความเข้าใจในหลักการออกแบบส่วนติดต่อผู้ใช้แล้ว ยังต้องมีทักษะด้านศิลปะ เพื่อให้งานออกมาดูสวยงามน่าใช้มากยิ่งขึ้น

5.3 ข้อจำกัด

การพัฒนาแอปพลิเคชัน ยังมีข้อจำกัดบางประการ ดังต่อไปนี้

1) รูปแบบของการเชื่อมต่อ สามารถเชื่อมต่อ ผ่าน Wireless ได้อย่างเดียว คาดว่าเนื่องจาก การใช้งานบลูทูธมีความซับซ้อน และอาจทำให้การประมวลช้า
2) ความเสถียรของการเคลื่อนไหว คาดว่าเนื่องจาก ต้องมีการคำนวณในการเคลื่อนที่ของ ไอคอน อาจทำให้มีข้อจำกัดต่อการกด เช่น การกดย้ำ เป็นต้น
3) ในส่วนของโหมดสื่อมัลติมีเดีย จะทำงานได้ดีบน โปรแกรม Window Media Player โปรแกรมสื่อบางตัว จะสามารถใช้ได้แค่บางฟังก์ชันเท่านั้น
4) เนื่องจากระบบปฏิบัติการแอนดรอยด์มีหลายเวอร์ชันซึ่งแตกต่างกันทำให้เครื่องมือในการพัฒนาการใช้งานบลูทูธ มีข้อจำกัดกับเวอร์ชันเก่า ซึ่งแอปพลิเคชันนี้ต้องการให้รันได้หลายเวอร์ชันในแอนดรอยด์ หากมีการใช้งานบลูทูธจะทำให้ระบบที่มีเวอร์ชันต่ำทำงานไม่มีประสิทธิภาพจึงสามารถเชื่อมต่อได้เพียง 1 ประเภท
5) การนำเสนองานโดยใช้แอปพลิเคชัน จำเป็นต้องอยู่ในพื้นที่ที่มีบริการ Wireless Lan เพราะการใช้งานเว็บเบราเซอร์เพื่อนำเสนองานต้องการ Internet เพื่อเป็นเครือข่ายเชื่อมไปยังเว็บที่ต้องการนำเสนอ

เอกสารอ้างอิง

[1] *widebase: RemoteProcedureCall.* (2000). Retrieved July, form http://www.widebase.net

[2] *วิกิพีเดีย. แบบจำลองโอเอสไอ.* (2556). ค้นเมื่อ กรกฎาคม 2556 จากเว็บไซต์: http://th.wikipedia.org

[3] วิปัสสนา. OSI Model. (2556). ค้นเมื่อ กรกฎาคม 2556 จากเว็บไซต์: https://sites.google.com/site/wpsn51

[4] *9M. รู้จักกับแอนดรอยด์.* (2556). ค้นเมื่อ สิงหาคม 2556 จากเว็บไซต์: http://www.sourcecode.in.th

[5] *M.5-2 Borvon.Android Developers.* (2011). Retrieved August, form http://forum.esarn.com

[6] *MCLANCY. Designing for Touch.* (2013). Retrieved August, form http://mobiforge.com

[7] Sarangi P. Parikh & Joel M. Esposito. (2555). ความคิดในแง่ลบสำหรับขนาดเล็กระบบสัมผัสคาปาซิทีฟอินเตอร์เฟสหน้าจอสัมผัสที่ใช้งาน การศึกษาสำหรับการป้อนข้อมูลงาน

[8] Sarun Intakosum. เอกสารประกอบการสอนวิชา distributed operating system. (ม.ป.ป). Remote Procedure Call

[9] *androidthai. การติดตั้ง Eclipse สำหรับเขียน Java และพัฒนาแอนดรอยด์.* (2556). ค้นเมื่อ กันยายน 2556 จากเว็บไซต์ http://androidthai.in.th/conternt-android/182-eclipse-adt-sdk-v20-android.html

[10] *การรองรับหน้าจอที่หลากหลาย.* (2556). ค้นเมื่อ มกราคม จากเว็บไซต์: http://www.akexorcist.com/2012/06/android-code-supporting-multiple.html

ภาคผนวก ก.

การติดตั้งโปรแกรมและการใช้งาน

ก.1 การติดตั้ง Eclipse สำหรับเขียนโปรแกรม และพัฒนาแอนดรอยด์แอปพลิเคชัน [9]

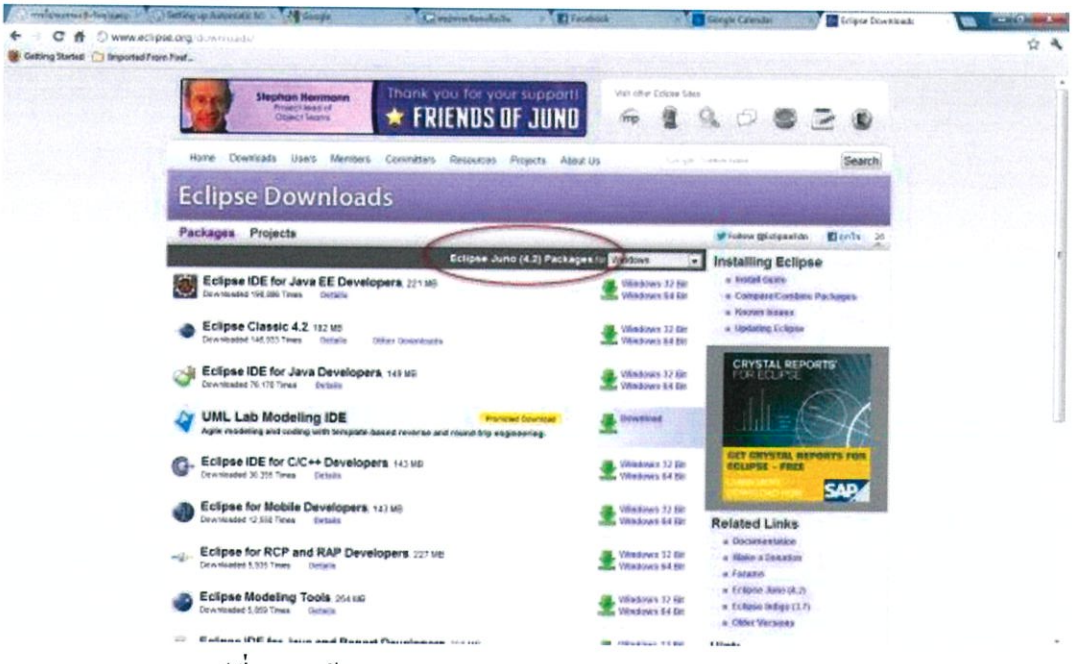
ในการพัฒนาแอปพลิเคชันแอนดรอยด์ หรือ การเขียนโปรแกรมแอนดรอยด์ ต้องมี

1. JRE หรือ Java Runtime
2. Eclipse
3. ADT Android developer tools
4. SDK

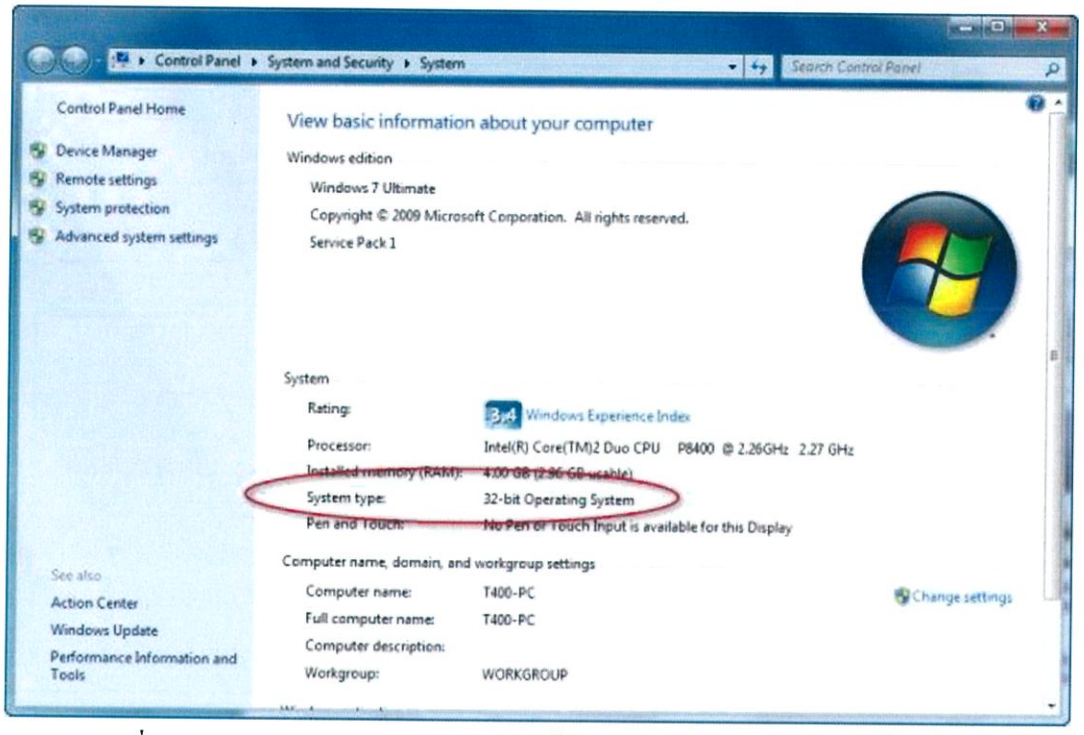
หลังจากงาน Google I/O ที่ผ่านมา มีอะไรใหม่เกิดขึ้นมากมาย เช่น

1. Android 4.1 Jelly-Bean
2. Eclipse version 4.2 Juno
3. SDK v20

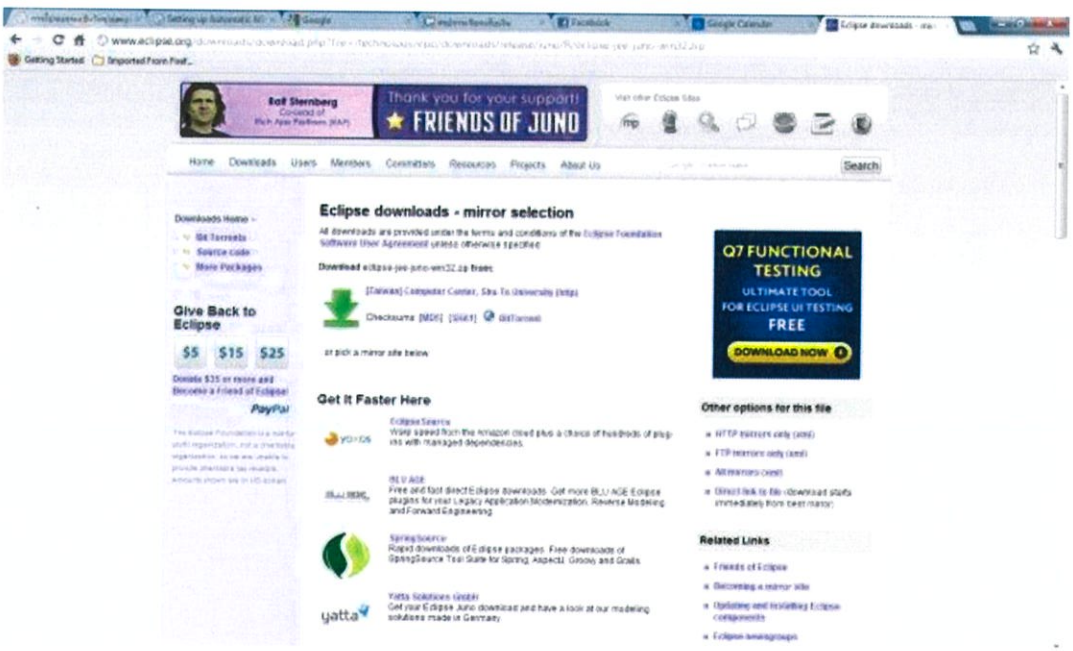
Eclipse Juno สามารถโหลดได้จาก http://www.eclipse.org/downloads/



รูปที่ ก.1 หน้าแสดงผลผล http://www.eclipse.org/downloads/

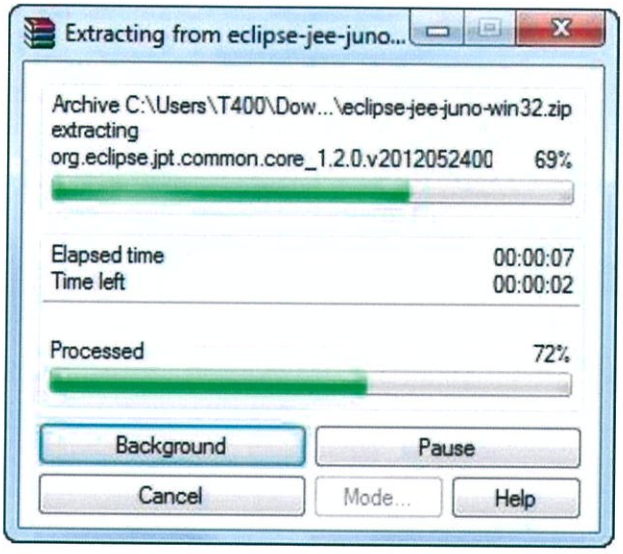


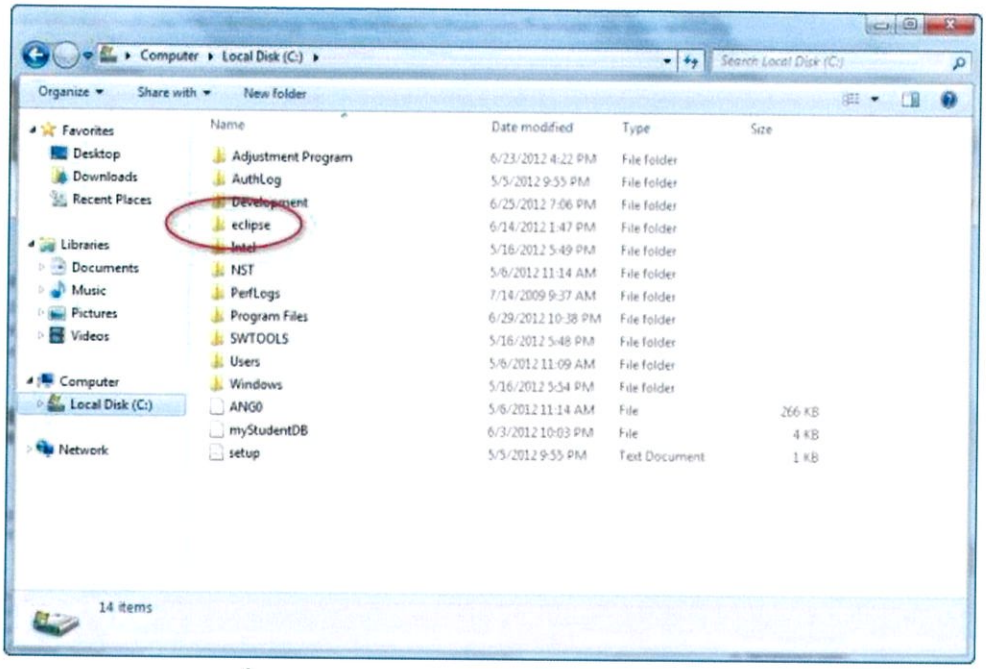
รูปที่ ก.2 หน้าแสดงผลผลการตรวจสอบเครื่องคอมพิวเตอร์ว่าเป็น 32bit หรือ 64bit



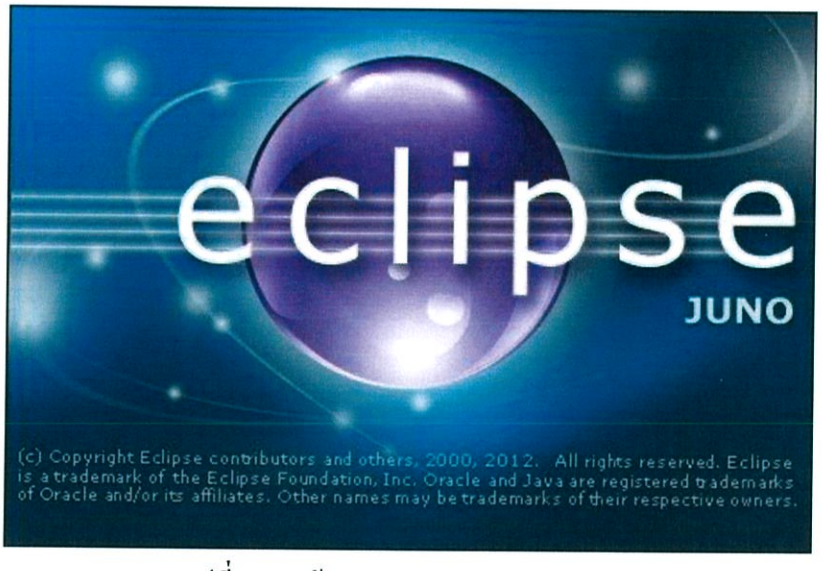
รูปที่ ก.3 หน้าแสดงผลผล หน้าต่างในการดาวน์โหลด Eclipse

รูปที่ ก.4 หน้าแสดงผลผล โหลด Eclipse ได้แล้ว Extract ไปเก็บไว้ที่ C:/

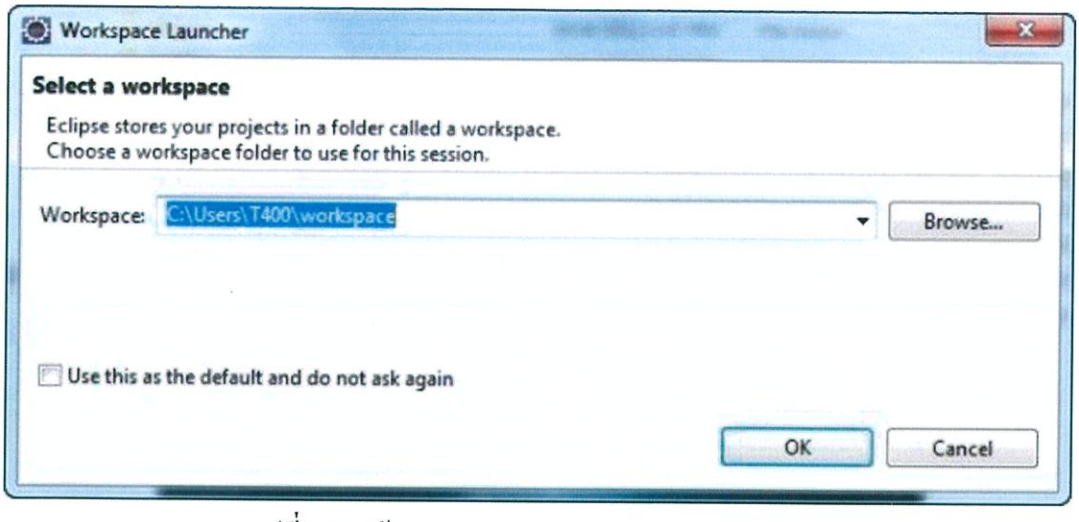




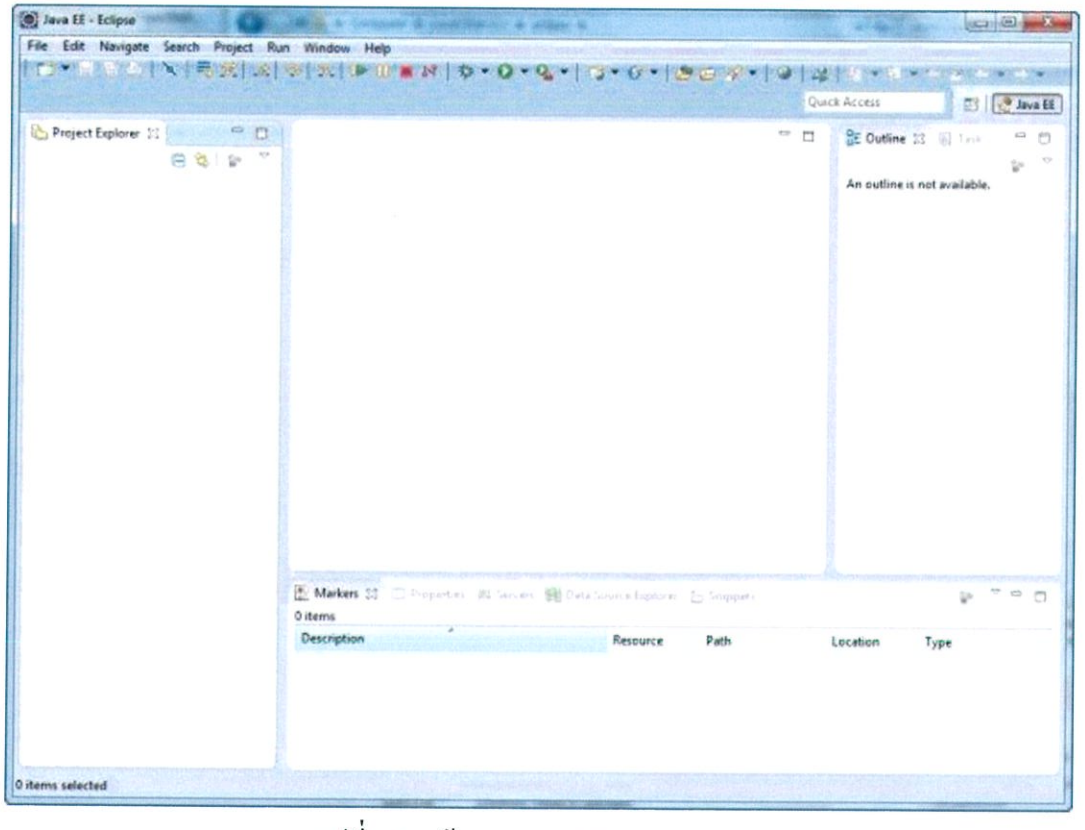
รูปที่ ก.5 หน้าแสดงผลผล หน้าต่างในการเก็บ Eclipse



รูปที่ ก.6 หน้าแสดงผล Eclipse Juno v 4.2

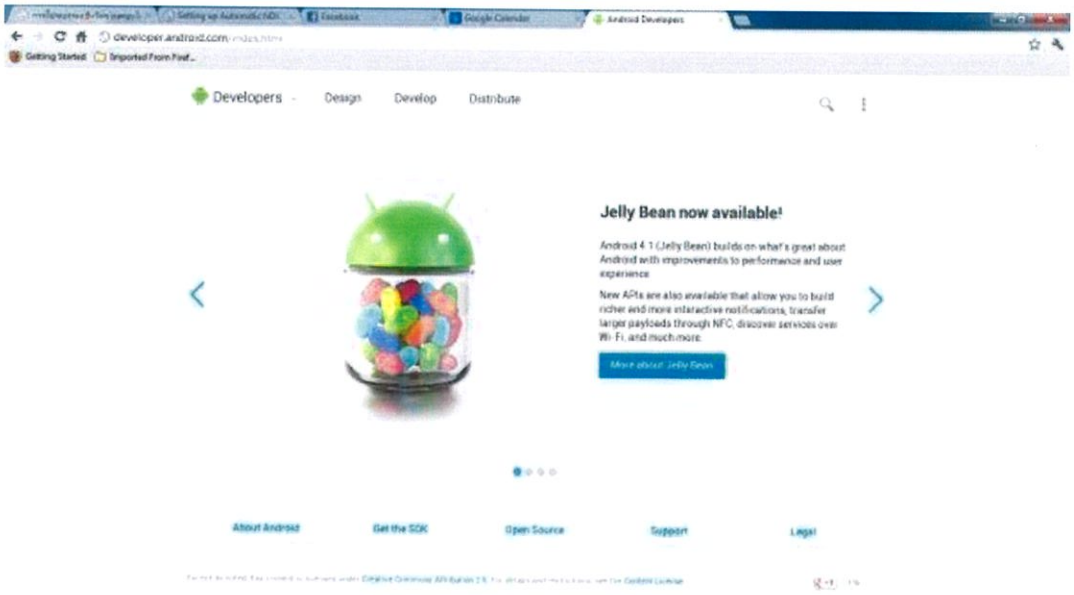


รูปที่ ก.7 หน้าแสดงผล Workspace Eclipse Juno v 4.2

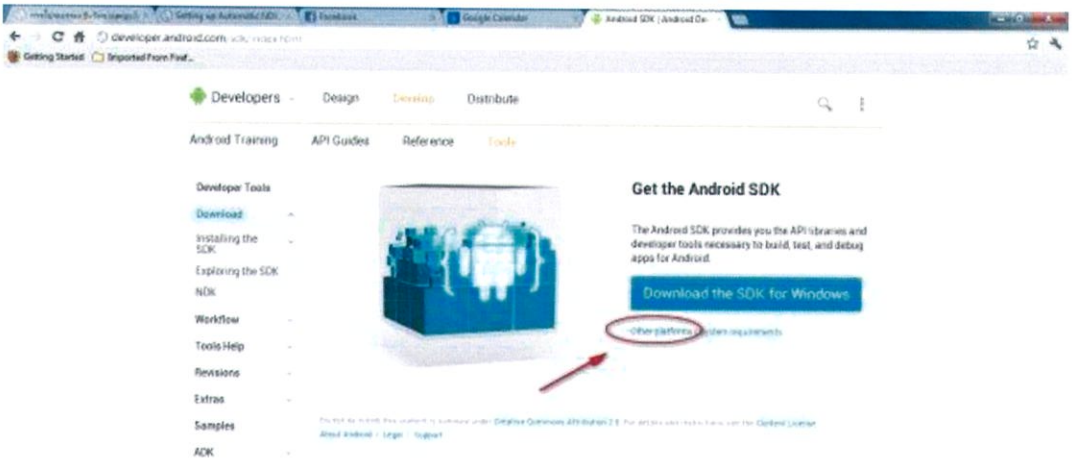


รูปที่ ก.8 หน้าแสดงผล Eclipse Juno v 4.2

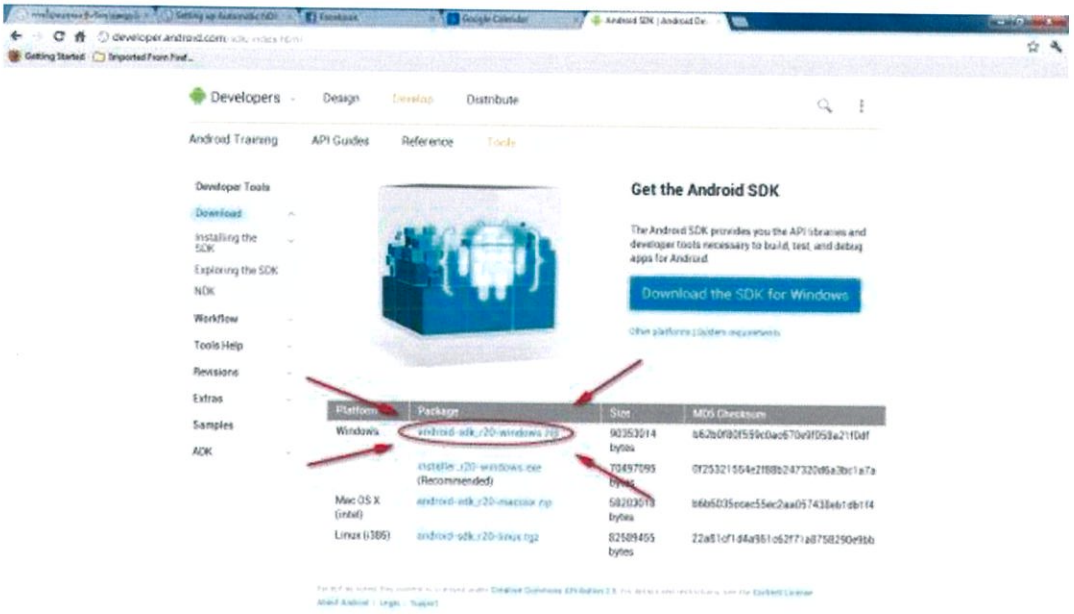
ติดตั้ง SDK v20 โดย ADT ไปที่ http://developer.Android.com/sdk/index.html



รูปที่ ก.9 หน้าแสดงผล Android SDK

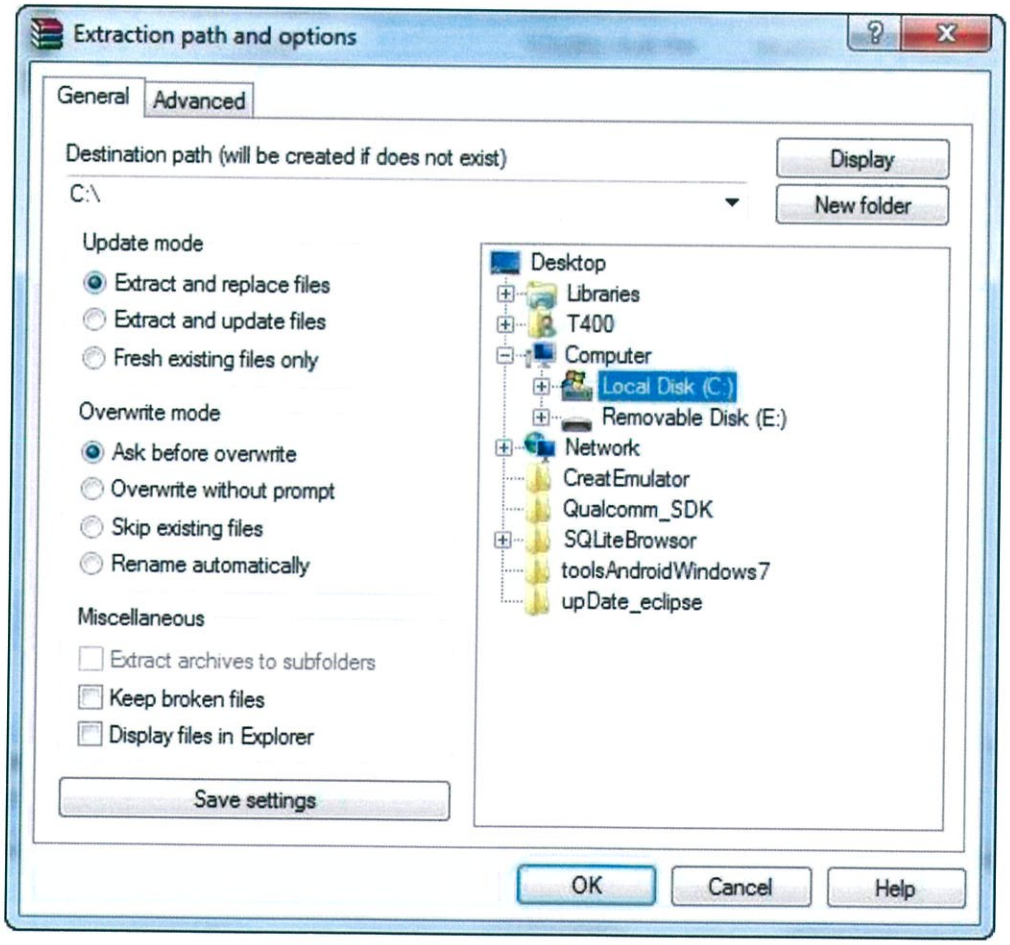


รูปที่ ก.10 หน้าแสดงผล ลิงก์การโหลด Android SDK

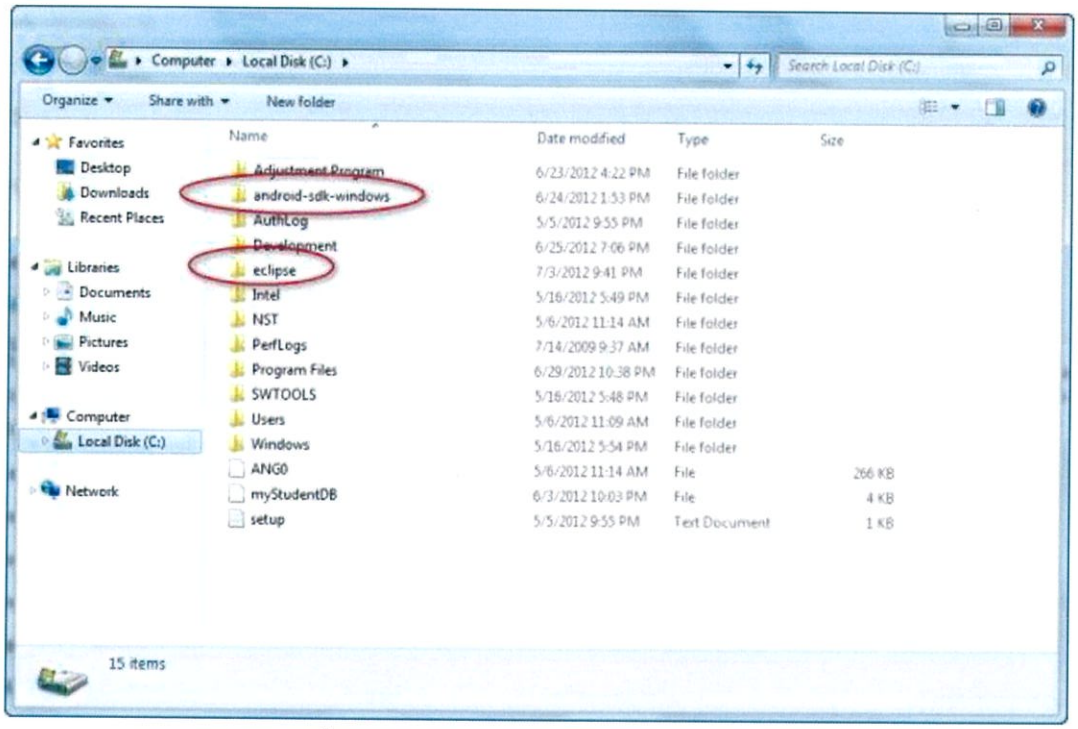


รูปที่ ก.11 หน้าแสดงผล ไฟล์ที่จะเลือกดาวน์โหลด

เมื่อได้ SDK v20 แล้วแตกออก เก็บไว้ที่ C:/

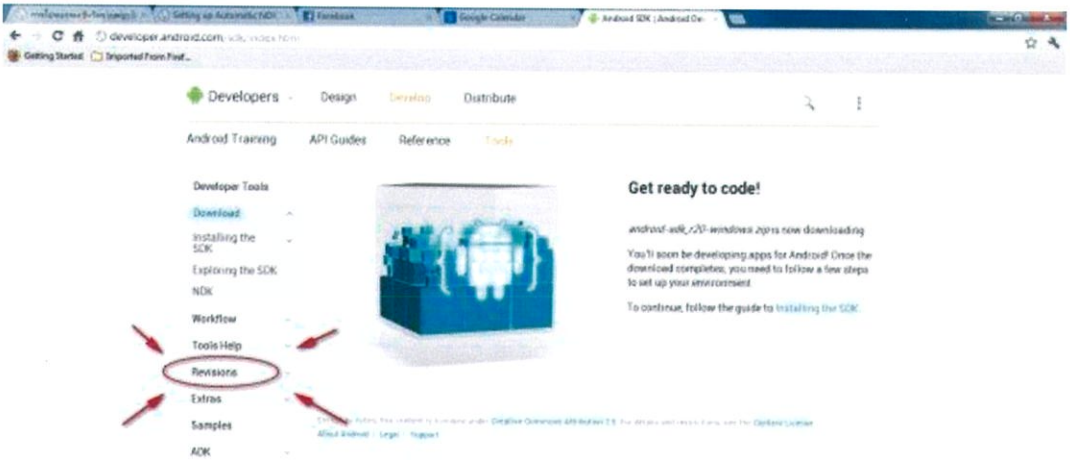


จะได้ดังนี้

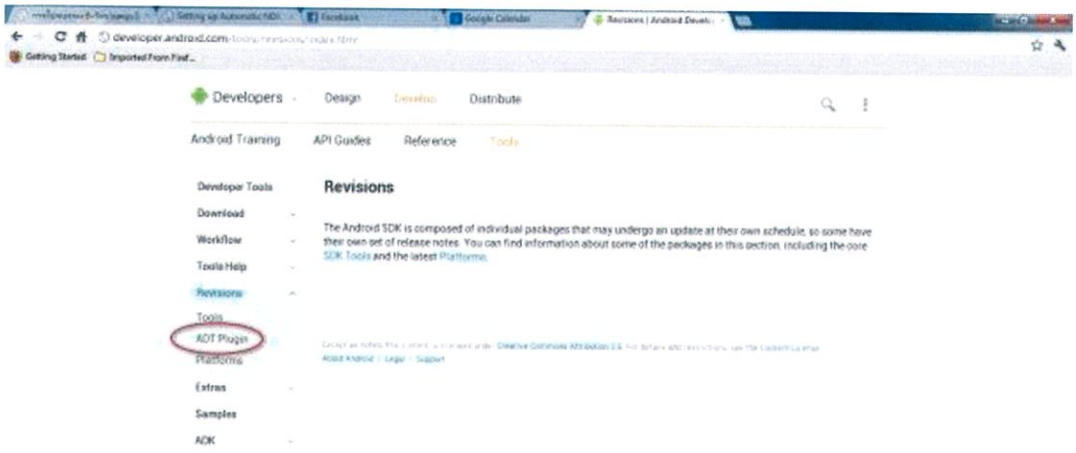


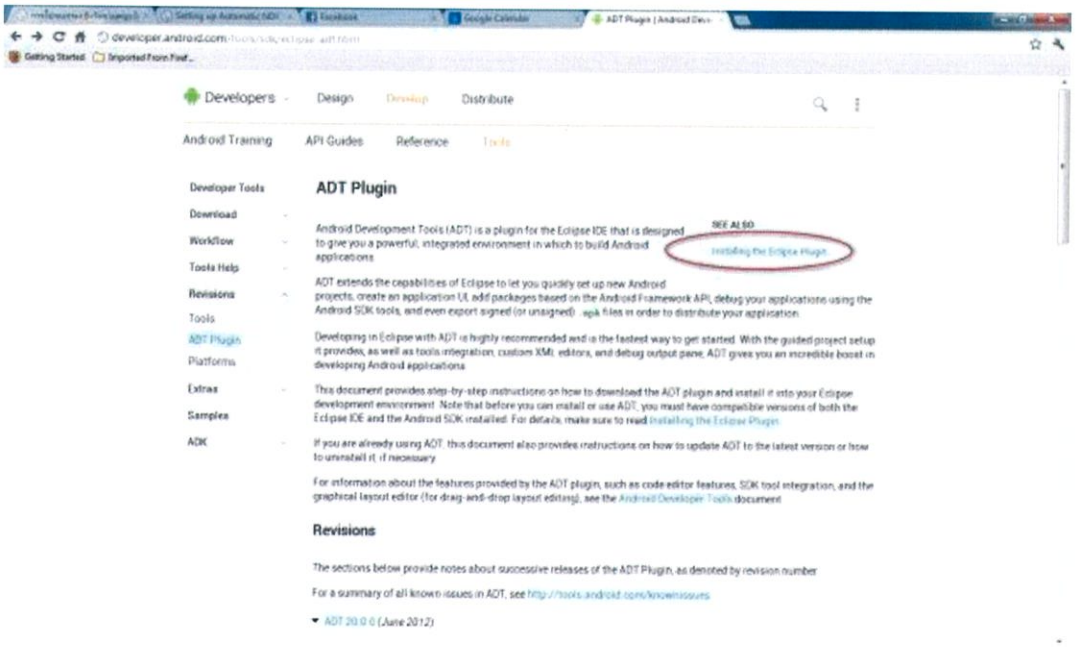
รูปที่ ก.12 หน้าแสดงผล ที่จัดเก็บ Android SDK

ก่อนที่เราจะสามารถติดตั้ง Platform Android ได้เราต้องติดตั้งผ่าน ADT



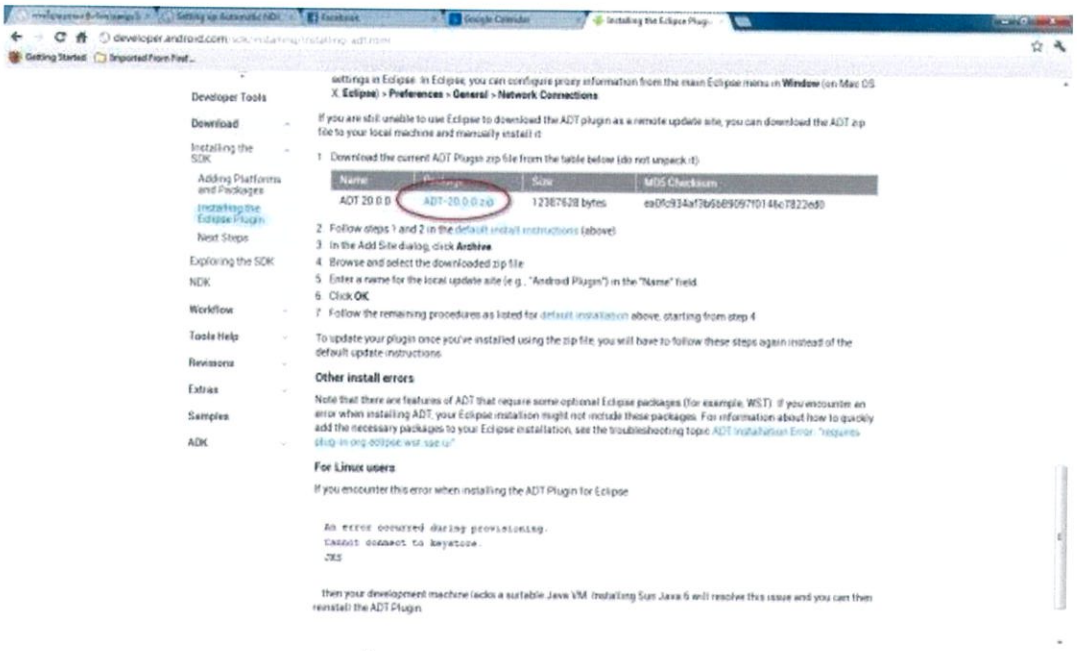
เลือก ADT Plugin





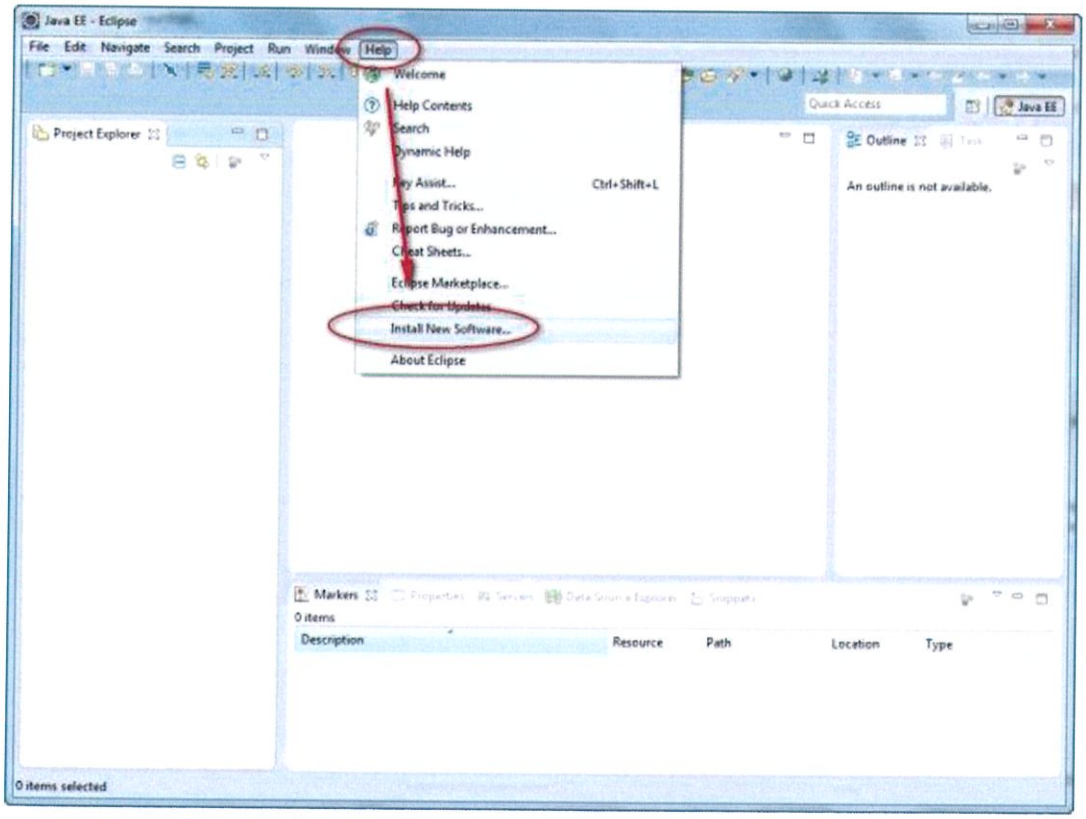
รูปที่ ก.13 หน้าแสดงผล ADT Plugin

เลือกเป็น .zip

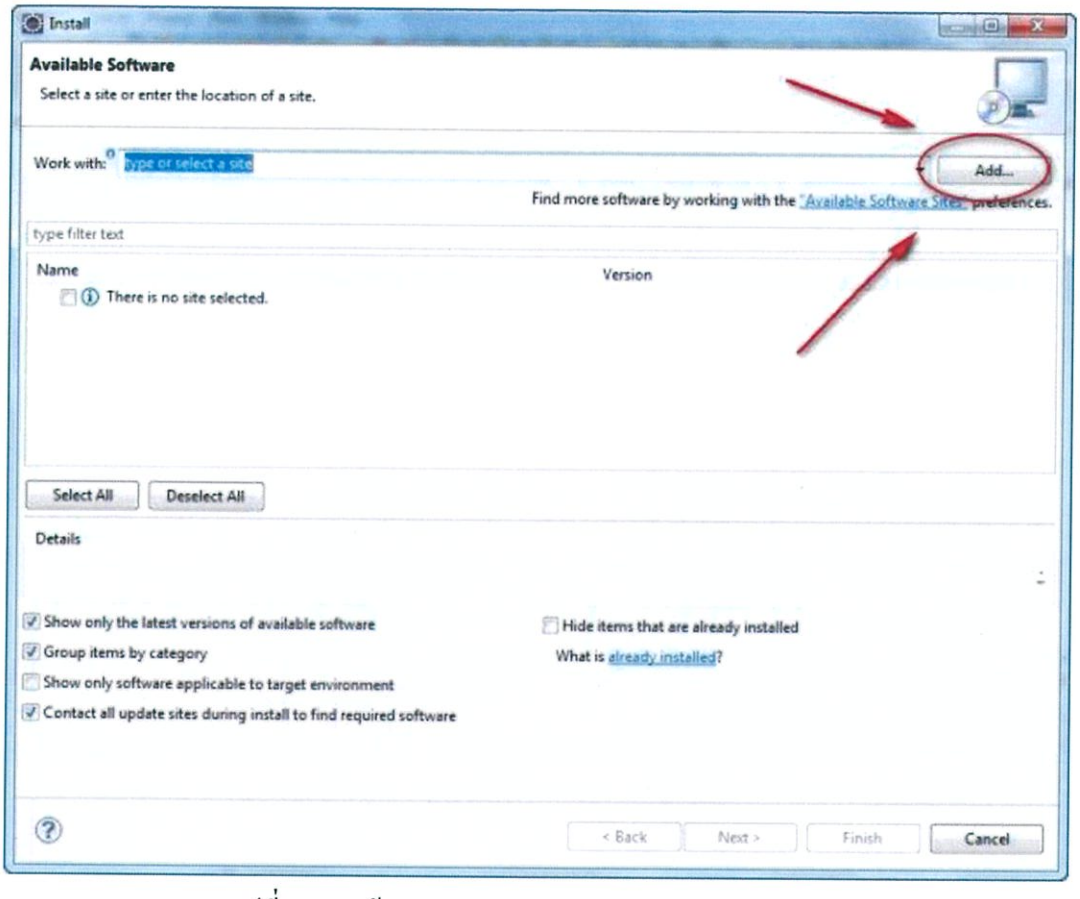


รูปที่ ก.14 หน้าแสดงผล .zip ไฟล์ ADT

โหลดเสร็จเก็บไว้ก่อน ไปที่ Eclipse > Help > Install New Software

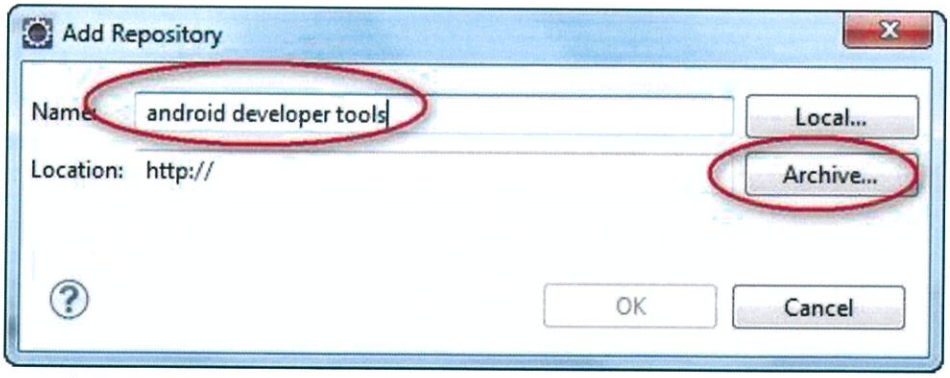


รูปที่ ก.15 หน้าแสดงผล Eclipse Install New Software

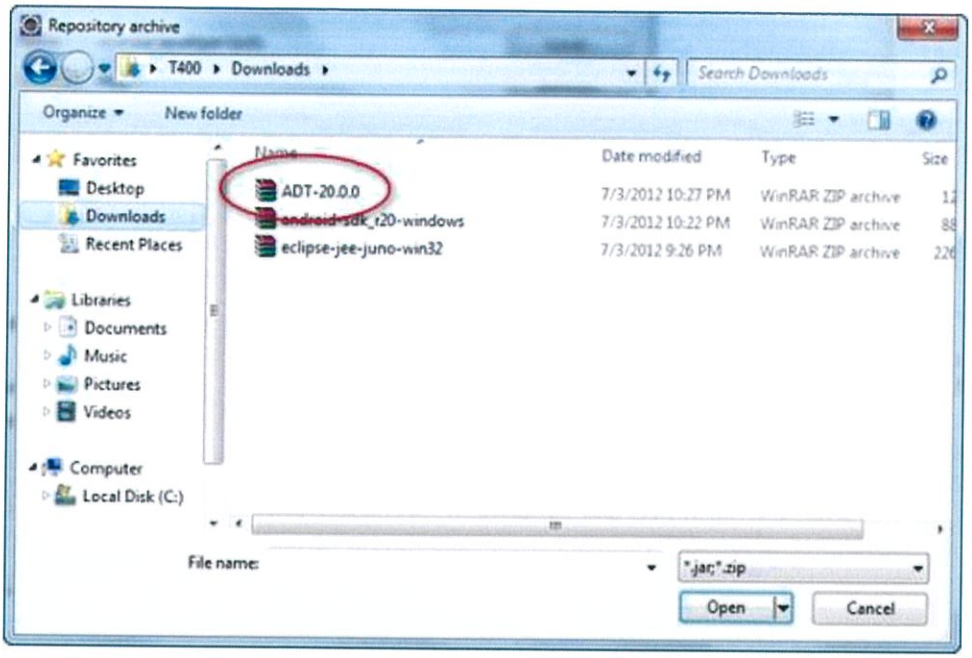


รูปที่ ก.16 หน้าแสดงผล Install Software Add Repository

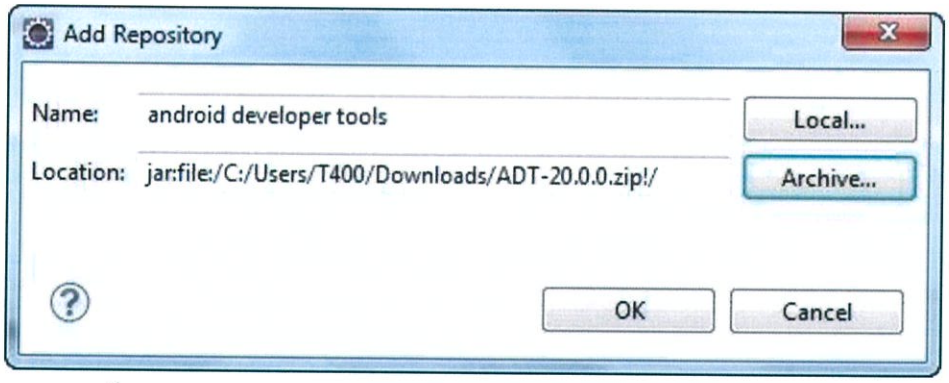
Add Repository



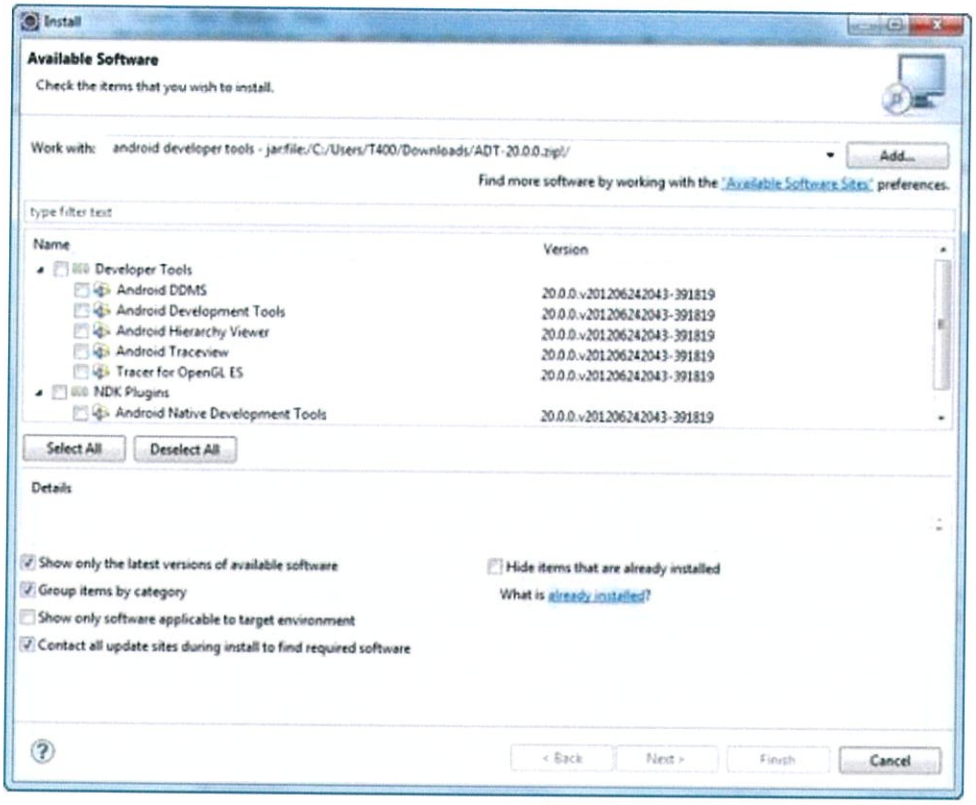
ไปที่ ADT ที่เป็น zip ที่เราโหลดมา

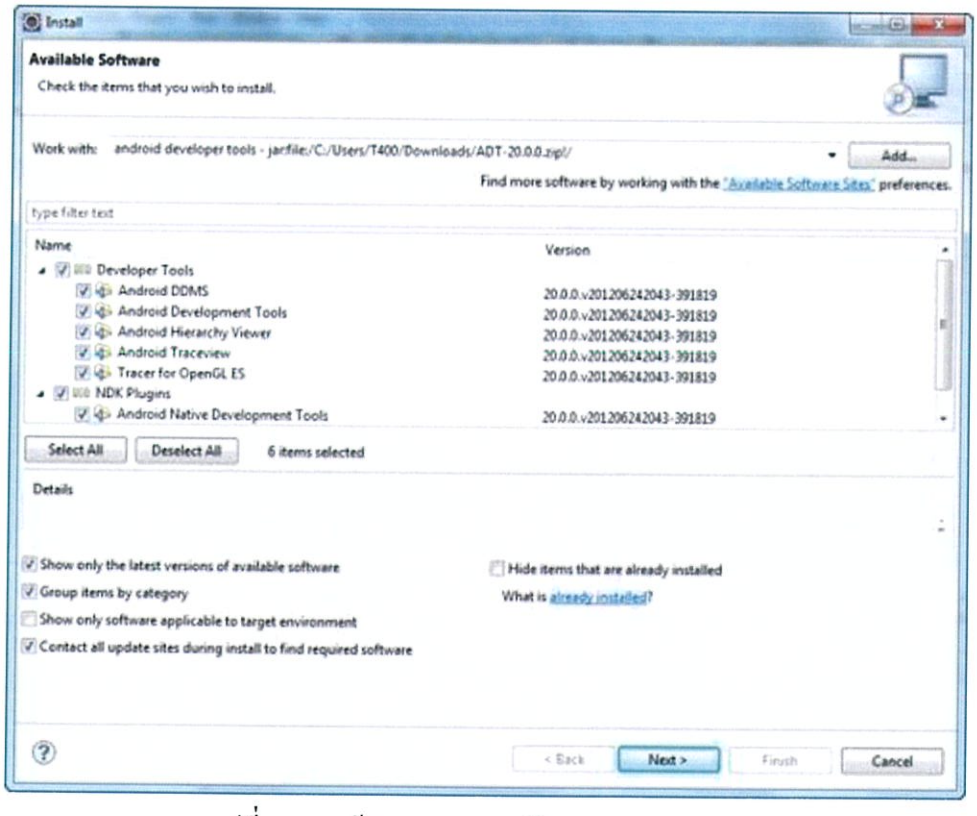


รูปที่ ก.17 หน้าแสดงผล ADT-20.0.0



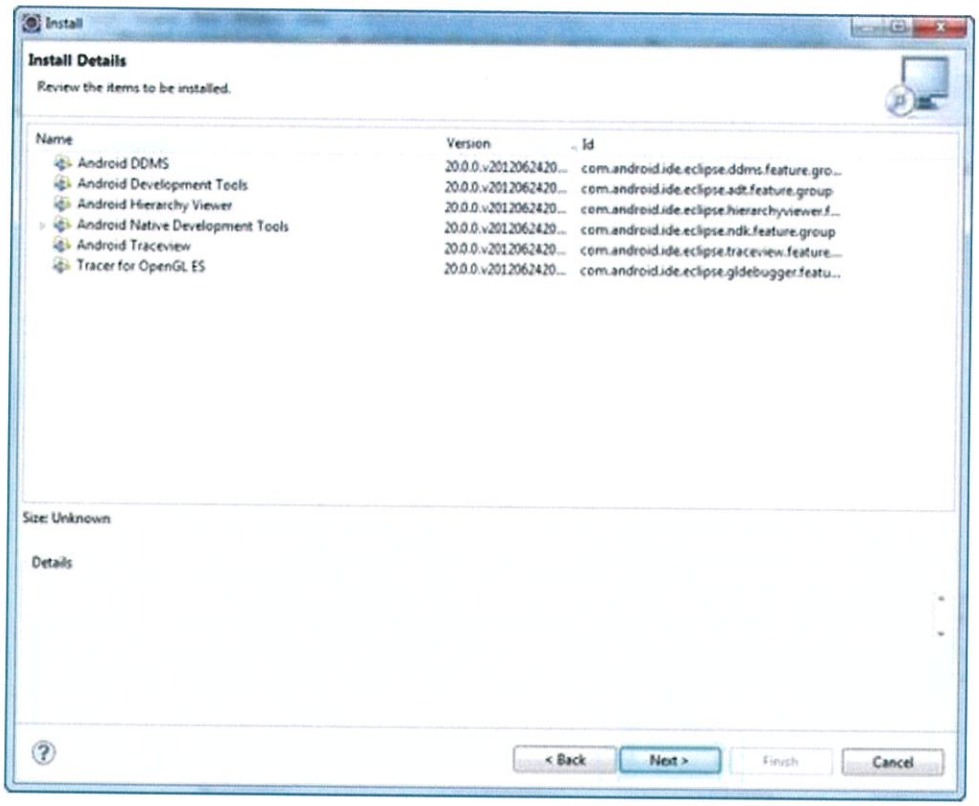
ทำการเลือกติดตั้ง

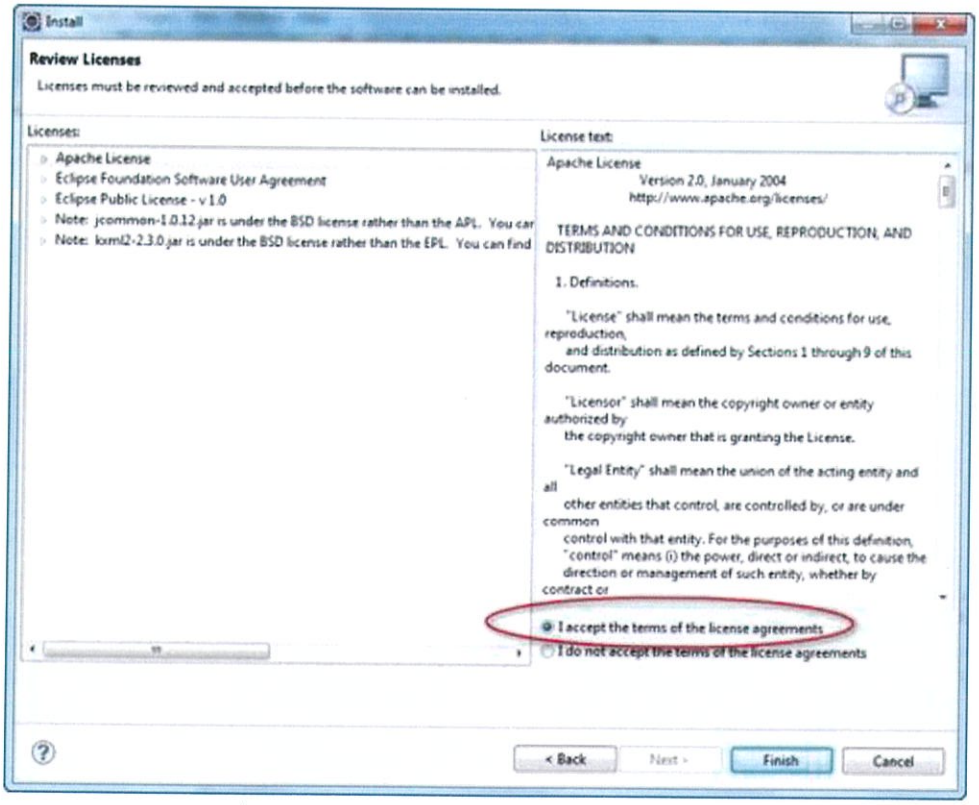




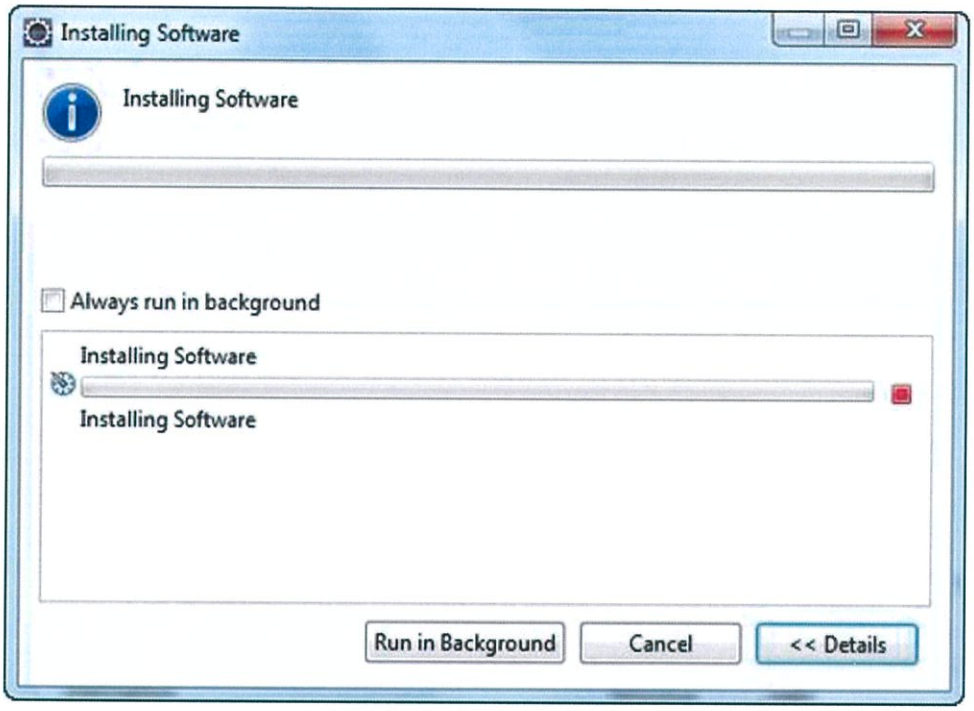
รูปที่ ก.18 หน้าแสดงผล การเลือก Developer Tools

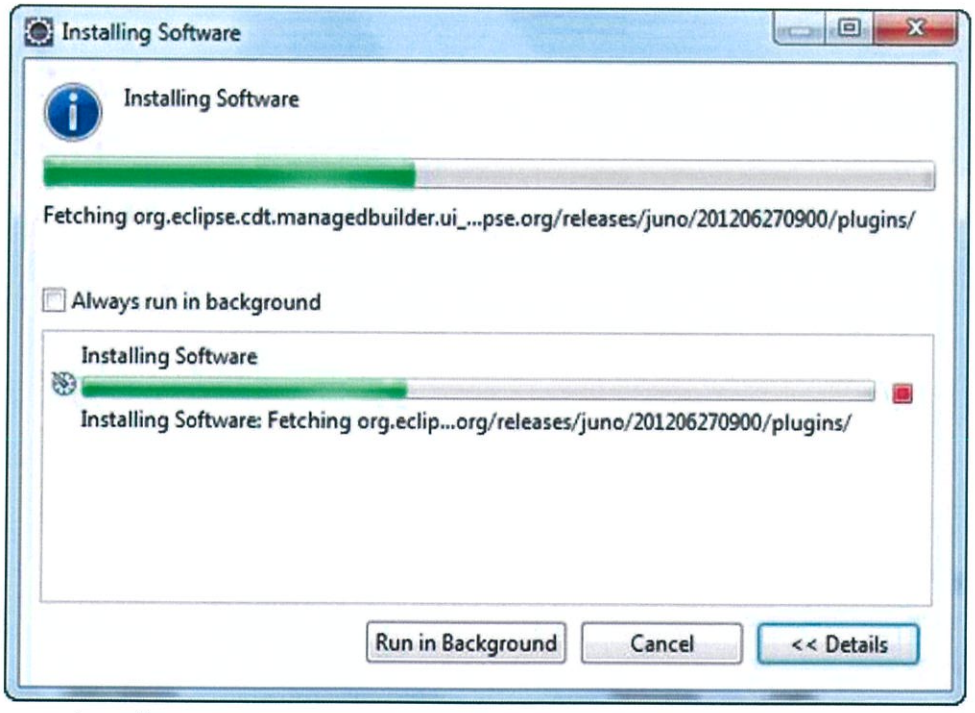
รอติดตั้ง ADT เสร็จ



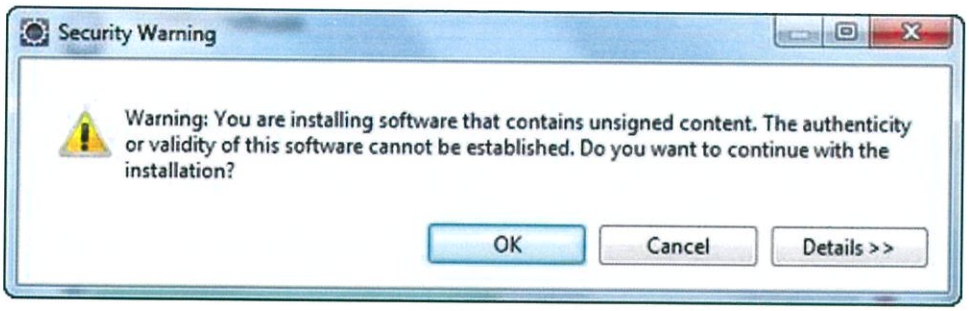


รูปที่ ก.19 หน้าแสดงผล การยอมรับเงื่อนไขการติดตั้ง

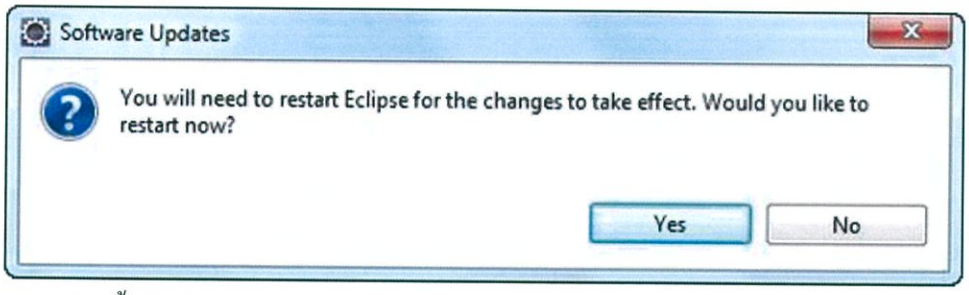




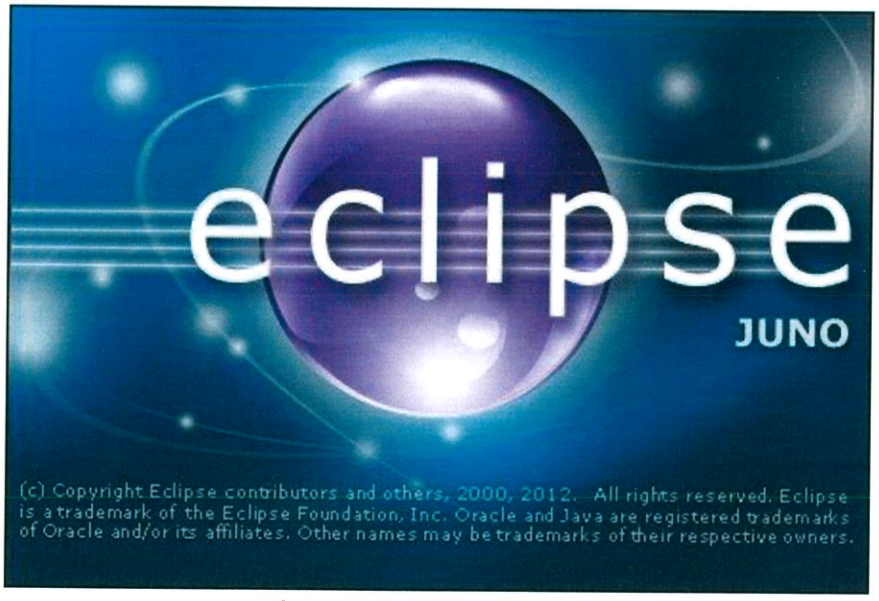
คลิก OK เพื่อติดตั้ง Plugin



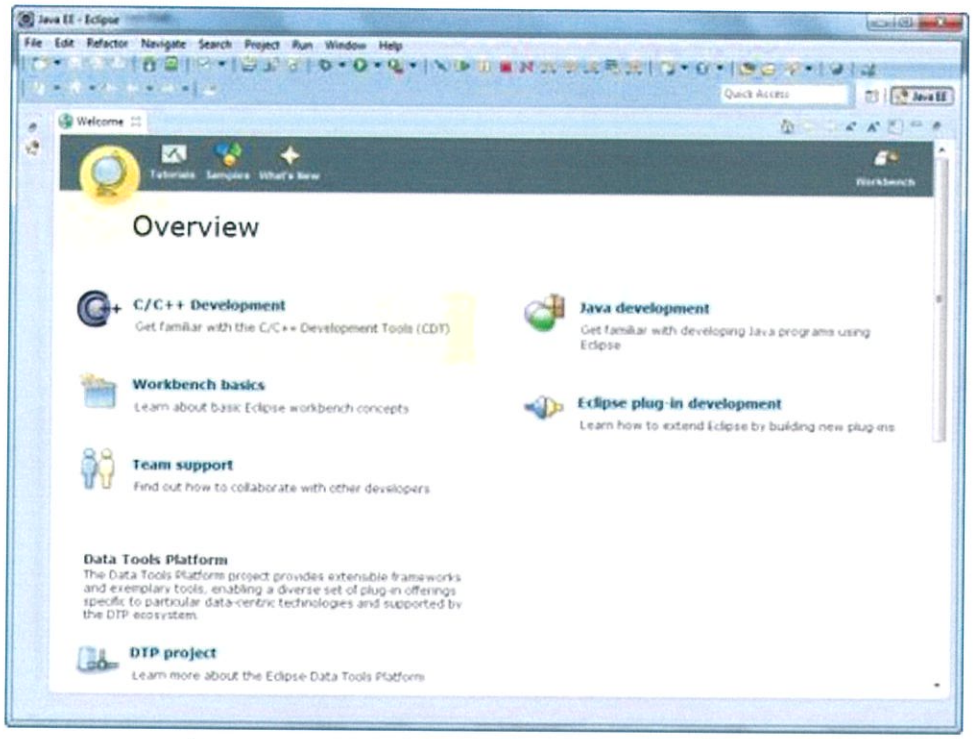
แล้วระบบจะขอ Restart Eclipse



Eclipse จะเปิดขึ้นใหม่

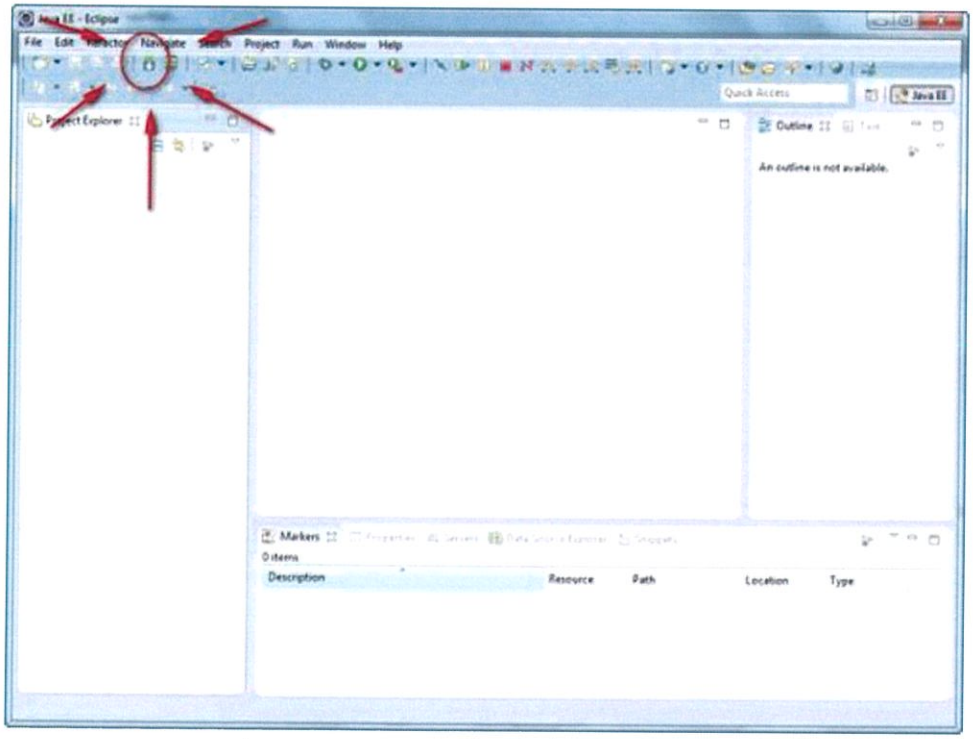


รูปที่ ก.20 หน้าแสดงผล Eclipse Juno



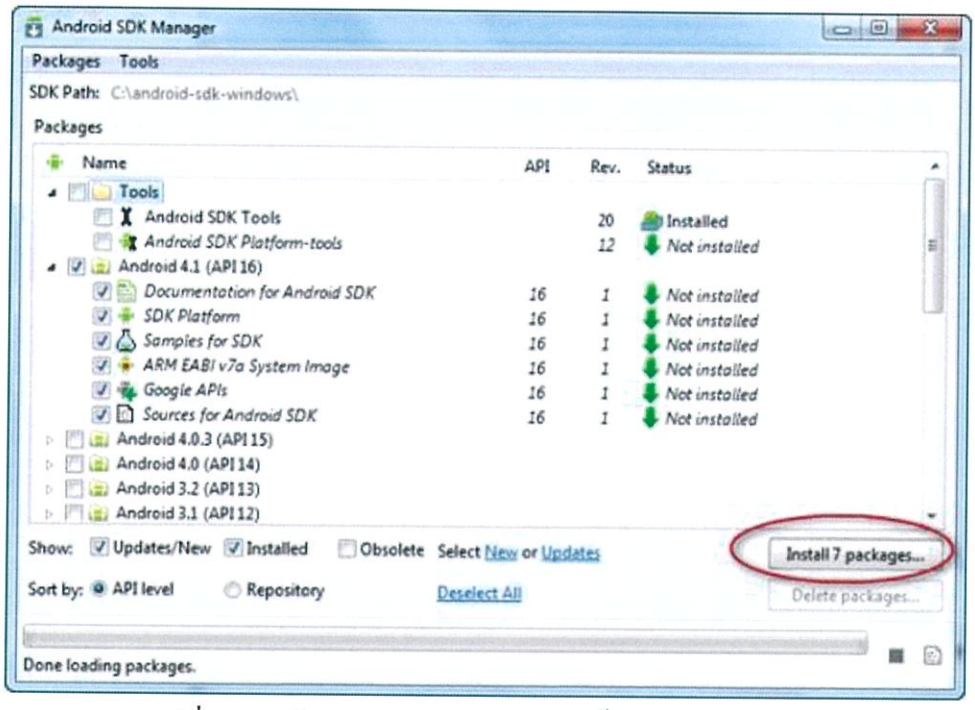
รูปที่ ก.21 หน้าแสดงผล หน้าหลักการใช้งาน Eclipse Overview

ต่อไปเป็นการติดตั้ง Platform Android

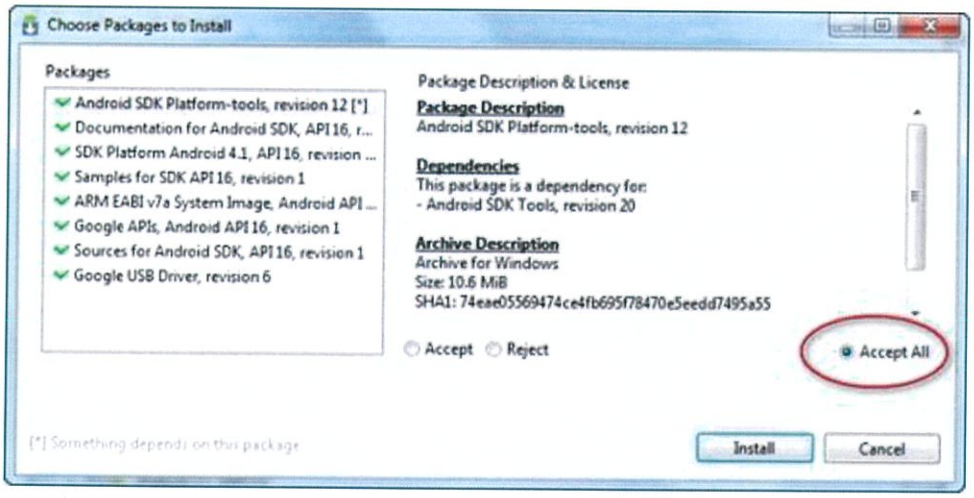


รูปที่ ก.22 หน้าแสดงผล การติดตั้ง Platform Android

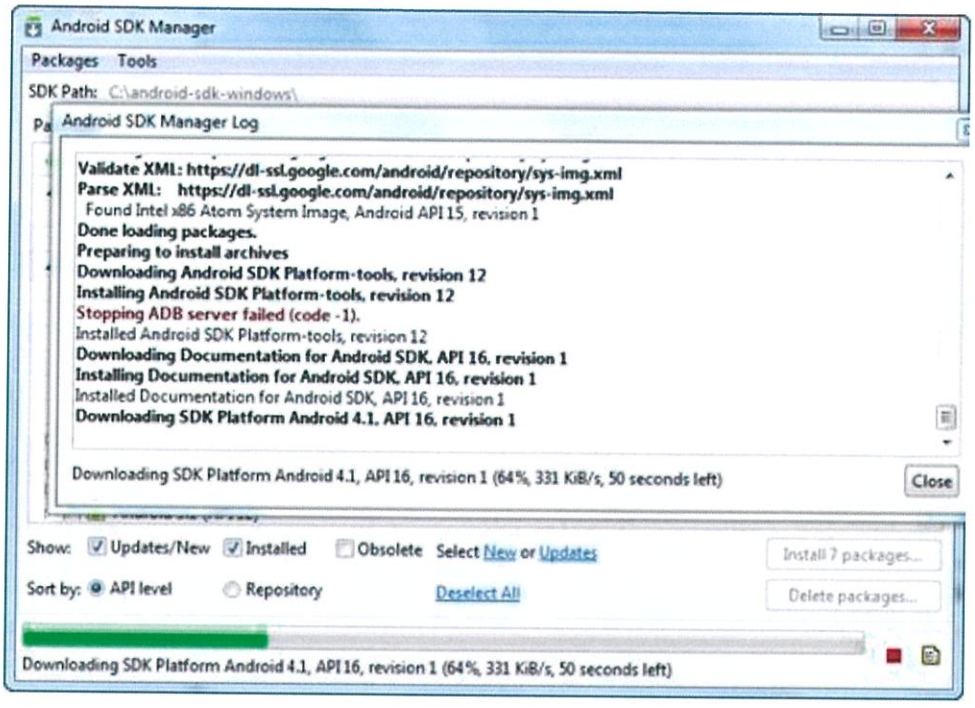
ติดตั้ง Jelly Bean ก่อน

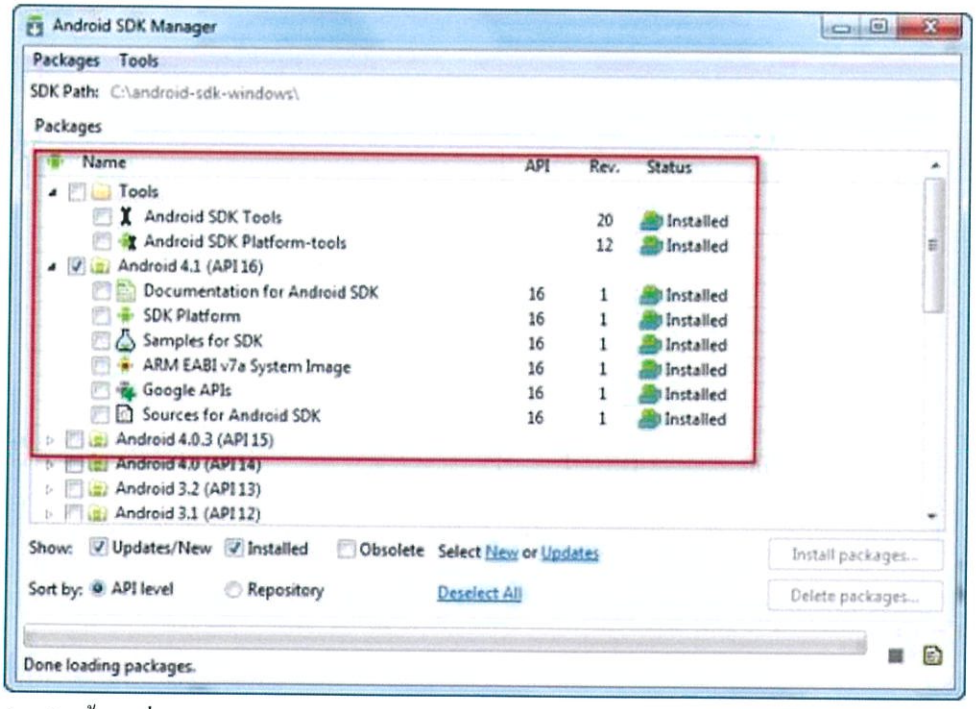


รูปที่ ก.23 หน้าแสดงผล การเลือกการติดตั้ง Platform Android

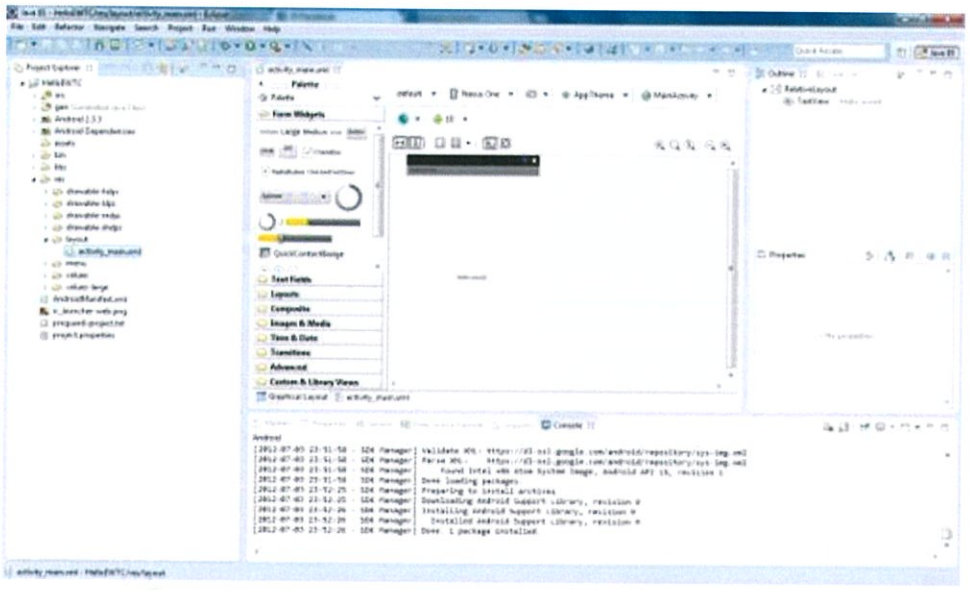


รอจนติดตั้งเสร็จ





แล้วค่อยติดตั้งตัวอื่นๆ



รูปที่ ก.24 หน้าแสดงผล การใช้งาน การพัฒนาแอปพลิชันแอนดรอยด์

จากนั้นทดสอบการเขียนโปรแกรม

ภาคผนวก ข.

การรองรับหน้าจอที่หลากหลาย

การรองรับหน้าจอที่หลากหลาย - Supporting Multiple Screens [10]

ข้อมูลในการออกแบบแอปพลิเคชันบนแอนดรอยด์ จะอ้างอิงข้อมูลจาก Android Developer ซึ่งเนื้อหา จะเปลี่ยนแปลงตามความเหมาะสมแทน การรองรับหน้าจอที่หลากหลาย (Supporting Multiple Screens) เป็นที่ทราบกันว่า อุปกรณ์แอนดรอยด์นั้นมีจอแสดงผลที่แตกต่างกันออกไปมากมายไม่ว่าจะเป็นขนาดจอหรือความละเอียดของจอ ดังนั้นการพัฒนาแอปพลิเคชันขึ้นก็ควรทำให้รองรับกับหน้าจอได้หลายขนาดจะดีที่สุด ซึ่งบทความนี้ก็จะกล่าวถึงเรื่องหน้าจอของแอนดรอยด์

Screen size

ขนาดความกว้างและยาวของหน้าจอบนอุปกรณ์แอนดรอยด์ ส่วนมากจะวัดกันเป็นนิ้ว เช่น 2.4", 3.5", 4.5", 5", 7" หรือ 10.1" เป็นต้น สำหรับในแอนดรอยด์จะแบ่งขนาดหน้าจอออกเป็น 4 ขนาด คือ Small, Normal, Large และ Extra Large โดยรายละเอียดเบื้องลึกศึกษาได้จาก Android Design ว่าด้วยเรื่อง Size และ Density ของหน้าจอ

Screen Density

ความหนาแน่นของหน้าจอซึ่งหมายถึงจำนวน Pixels ต่อหนึ่งหน่วยพื้นที่ โดยนิยมใช้หน่วยเป็น จุดต่อนิ้ว (dpi) อย่างเช่น 330 dpi ก็คือในหนึ่งนิ้วจะมี 330 จุดหรือ 300 Pixels ดังนั้นจอที่มีค่า dpi มาก ภาพที่ได้ก็จะคมชัดมากขึ้น สำหรับในแอนดรอยด์จะแบ่งความหนาแน่นของหน้าจอออกเป็น 6 ระดับ คือ Low, Medium, High, Extra High, Extra Extra High และ Extra Extra Extra High

Orientation

ทิศทางของหน้าจอที่สามารถเปลี่ยนเป็นแนวนอนและแนวตั้งได้ ซึ่งไม่ได้ถูกกำหนดแค่ตอนเริ่มใช้งานเท่านั้น แต่สามารถเปลี่ยนแปลงได้ตลอดเวลาที่ใช้งาน โดยจะเรียกทิศทางของหน้าจอเป็น Portrait และ Landscape

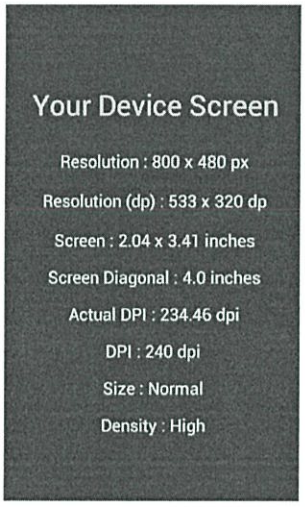
Resolution

ความละเอียดของหน้าจอเป็นจำนวน Pixel ในความกว้างและความสูง ซึ่งเป็นปัจจัยหลักในการรองรับกับหน้าจอหลายขนาดอีกทั้งยังสัมพันธ์กับขนาดหน้าจอและความหนาแน่นของหน้าจออีกด้วย โดยส่วนมากจะมีอัตราส่วนระหว่างความกว้างและความสูงดังนี้ 4:3, 16:9 และ 16:10 แต่ก็มีขนาดอื่นๆเช่นกัน สำหรับในแอนดรอยด์จะกำหนดขนาดหน้าจอเป็นตัวเลขโดยตรง อย่างเช่น 480x320, 800x480, 1024x600 และ 1280x720 เป็นต้น

Density-independent pixel (dp)

เป็น Pixel เสมือน ที่เกิดจากการคำนวณด้วยสูตร px = dp * (dpi / 160) อย่างเช่นหน้าจอขนาด 240 dpi ดังนั้น 1 dp ก็จะเท่ากับ 1.5 px โดยรายละเอียดเบื้องลึกศึกษาได้จาก [Android Design] แท้จริงแล้วหน่วย dp คืออะไร

สำหรับผู้ที่หลงเข้ามาอ่านที่อยากรู้ว่าอุปกรณ์แอนดรอยด์ของคุณ มีหน้าจอขนาดเท่าไรและความหนาแน่นเท่าไร ให้ดาวน์โหลด Check Screen [Google Play] ซึ่งเป็นแอพสำหรับเช็คขนาดหน้าจอที่เจ้าของบล็อกเขียนเอาไว้



รูปที่ ข.1 Application Check Screen

เพื่อให้แอปพลิเคชันทำงานได้อย่างเหมาะสม ควรใช้ Resource หลายๆขนาด โดยให้รองรับกับขนาดต่างๆ อย่างเช่น ถ้าแอปพลิเคชัน ต้องการใช้ Bitmap หนึ่งภาพ ก็ควรมีขนาดใหญ่หนึ่งภาพสำหรับหน้าจอ แบบ hdpi หนึ่งภาพ และ mdpi หนึ่งภาพ ldpi หนึ่งภาพ และ xhdpi ถ้าที่ต้องการให้รองรับกับอุปกรณ์แอนดรอยด์ที่ขนาดหน้าจอใหญ่มาก หรือจะกำหนดตาม Small, Normal, Large และ Extra Large ก็ได้ โดยที่เราไม่จำเป็นต้องกำหนดว่าหน้าจอขนาดใดให้ใช้ไฟล์ไหน เพราะแอนดรอยด์จะมีระบบจัดการดังกล่าวให้ เพียงแค่จัดเก็บไฟล์ ในชื่อเดียวกันในโฟลเดอร์ที่บอกขนาดต่างๆ แอปพลิเคชันก็จะใช้ภาพในโฟลเดอร์ตามขนาดของอุปกรณ์ที่ใช้ โดยศึกษาเรื่องนี้ได้จาก [Android Design] การทำงานของโฟลเดอร์ใน Resource [res]

บางทีผู้ที่หลงเข้ามาอ่านอาจจะสงสัยว่า ทำไมต้องจัดการเรื่องขนาดของภาพด้วย จึงแนะนำให้ดูภาพประกอบเลยดีกว่า อันนี้ภาพตัวอย่างจาก Android Develop ที่แสดงให้เห็นความแตกต่างระหว่างภาพแรกที่ใช้ภาพเดียวกันในหน้าจอที่ความหนาแน่นต่างๆ และภาพที่สองเป็นใช้ภาพแบ่งตามความหนาแน่นของหน้าจอตามที่แยกไว้ ซึ่งจะเห็นว่า ภาพที่สองจะมีขนาดของภาพที่ไม่เปลี่ยนแปลงสามารถรองรับกับหน้าจอได้หลายขนาด

รูปที่ ข.2 ไอคอนในหน้าจอที่แตกต่างกัน

ทำอย่างไรถึงจะรองรับหน้าจอได้หลายขนาด

- ประกาศว่าแอปพลิเคชันรองรับหน้าจอขนาดใดๆบ้าง โดยประกาศใน AndroidManifest.xml สามารถดูเนื้อหาได้จาก [Android Code] Android Compatibility ซึ่งจะทำให้แอปพลิเคชันใช้กับขนาดหน้าจอที่กำหนดเท่านั้น
- สร้าง Layout ที่แตกต่างกันสำหรับขนาดที่แตกต่างกัน โดยปกติแอนดรอยด์จะกำหนดขนาดแอปพลิเคชันให้เต็มหน้าจอ แต่ในบางครั้งหน้าจอที่ใหญ่กว่าอาจจะทำให้การแสดงผลผิดเพี้ยนได้ การสร้าง Layout สำหรับหน้าจอขนาดต่างๆขึ้นมาก็จะช่วยแก้ปัญหาดังกล่าวได้ดีเช่นกัน
- ใช้ภาพ Bitmap ในขนาดต่างกันสำหรับขนาดหน้าจอที่แตกต่างกัน ในกรณีที่เราใช้ภาพเพียงภาพเดียวโดยภาพนั้นพอดีกับหน้าจอขนาด Normal แต่เมื่อแสดงผลในหน้าจอ Large หรือ Small ภาพดังกล่าวก็จะถูกขยายหรือย่อลง ทำให้ภาพนั้นแตก ดูไม่สวยงามจึงควรใช้ภาพที่มีขนาดตามขนาดหน้าจอต่างๆ โดยแบ่งตามขนาดหน้าจอหรือความหนาแน่นของหน้าจอก็ได้

- ควรใช้ wrap_content, fill_parent หรือ ใช้หน่วย dp กำหนดขนาด โดย wrap_content จะเป็นการกำหนดขนาดให้พอดีกับวัตถุนั้นๆ ส่วน fill_parent จะเป็นการขยายขนาดให้เต็มพอดีกับหน้าจอ และ dp จะเหมาะกับการกำหนดขนาด ซึ่งจะช่วยให้ขนาดของวัตถุมีขนาดที่เหมาะสมกับหน้าจอขนาดต่างๆ
- ไม่ควรกำหนดขนาดเป็น px จากที่กล่าวมาแล้วว่าอุปกรณ์แอนดรอยด์มีความละเอียดของหน้าจอที่แตกต่างกัน ดังนั้นการกำหนดขนาดเป็นหน่วย px จะทำให้ขนาดของวัตถุคงที่ จึงมีขนาดเล็กลงเมื่อดูในอุปกรณ์แอนดรอยด์ที่มีความละเอียดหน้าจอสูง
- ไม่ควรใช้ AbsoluteLayout ถึงแม้ว่า Layout แบบนี้จะสะดวกในการกำหนดตำแหน่งที่แน่นอนก็ตาม แต่จะไม่มีความคล่องตัวกับหน้าจอหลายๆขนาด เพราะเนื่องจากกำหนดตำแหน่งตายตัวนั่นเอง ด้วยข้อจำกัดดังกล่าวจึงทำให้ AbsoluteLayout ถูกเลิกใช้กันตั้งแต่ Android 1.5 แล้ว แนะนำให้ใช้ RelativeLayout แทนสามารถกำหนดตำแหน่งโดยให้สัมพันธ์กับตำแหน่งของวัตถุรอบข้างแทน เช่น "to the right of" เป็นต้น
- สร้างโฟลเดอร์เก็บ Resource ตามขนาดของหน้าจอ อันนี้กล่าวถึงไปแล้วก่อนหน้า จึงไม่ขออธิบายเพิ่มเติม

Specific Resources

การกำหนด Resources โดยเฉพาะ

จากที่กล่าวถึงไปแล้วถึงข้อดีของการเก็บข้อมูลออกเป็นหลายๆขนาด ทีนี้จะอธิบายถึงวิธีการเก็บข้อมูลดังกล่าวให้แอนดรอยด์สามารถเรียกใช้ได้ถูกต้อง โดยมักจะใช้กับ drawable และ layout เท่านั้น ซึ่งปกติไฟล์ที่เกี่ยวกับ drawable จะถูกเก็บไว้ที่โฟลเดอร์ res/drawable และไฟล์ที่เกี่ยวกับ layout จะถูกเก็บไว้ที่ res/layout โดยการกำหนดแบบเจาะจงให้สร้างโฟลเดอร์ที่มีชื่อดังนี้

* ในตัวอย่างนี้จะเป็นตัวอย่างที่ใช้กับโฟลเดอร์ layout ในกรณีที่ใช้กับโฟลเดอร์ drawable ก็เพียงแค่เปลี่ยนจาก layout เป็น drawable เท่านั้น

res/layout/ สำหรับหน้าจอขนาด Normal และ Default

res/layout-small/ สำหรับหน้าจอขนาด Small

res/layout-normal/ สำหรับหน้าจอขนาด Normal

res/layout-large/ สำหรับหน้าจอขนาด Large

res/layout-xlarge/ สำหรับหน้าจอขนาด Extra Large

res/layout-ldpi/ สำหรับหน้าจอ ldpi

res/layout-mdpi/ สำหรับหน้าจอ mdpi

res/layout-hdpi/ สำหรับหน้าจอ hdpi

res/layout-xhdpi/ สำหรับหน้าจอ xhdpi

res/layout-port/ สำหรับหน้าจอในแนวตั้ง

res/layout-land/ สำหรับหน้าจอในแนวนอน

ในกรณีที่ต้องการกำหนดทั้งหน้าจอ ldpi และเป็นแบบแนวนอน ก็ให้ตั้งชื่อโฟลเดอร์ดังนี้

res/layout-port-ldpi/

โดยจะเรียงลำดับดังนี้ Size > Orientation > Density ดังนั้นจึงไม่สามารถตั้งชื่อโฟลเดอร์เป็น res/layout-ldpi-port/ ได้และกรณีที่ต้องการกำหนดโฟลเดอร์สำหรับหน้าจอขนาด Large แบบ hdpi และมีแนวนอน ก็กำหนดเป็น res/layout-large-land-hdpi/ และนอกจากมียังมีอื่นๆเพิ่มเติมอีกด้วย สามารถดูได้จาก Android Developer - Providing Resources

Alternative drawables

ทางเลือกต่างๆสำหรับ Drawables

สรุป ที่กล่าวมาทั้งหมดอาจจะฟังดูง่าย อย่างเช่นไฟล์รูปที่ต้องมีหลายขนาดเพื่อเก็บแยกไว้ให้เหมาะสมกับขนาดหน้าจอ แต่ปัญหาต่อมาก็คือ เราควรสร้างภาพดังกล่าว ขนาดเท่าไร ดังนั้นในการสร้างภาพสำหรับใช้งานบนแอปพลิเคชันโดยรองรับกับขนาดหน้าจอต่างๆ ให้ศึกษาจากบทความเกี่ยวกับการสร้าง Drawables Resource สำหรับรองรับหน้าจอหลายขนาด