

การพัฒนาเกมส์บนมือถือ
GAME DEVELOPMENT ON MOBILE PHONE



นายศุภชัย ทวีแสง
นายสันติ จารุสมบัติ

ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2546

เลขหมู่.....
เลขทะเบียน 55096
วัน,เดือน,ปี - 8 เม.ย. 2548

๖.....
๗.....

การพัฒนาเกมส์บนมือถือ

GAME DEVELOPMENT ON MOBILE PHONE



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2546

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนาเกมส์บนมือถือ

GAME DEVELOPMENT ON MOBILE PHONE

คณะผู้จัดทำ นายศุภชัย หวันแสง

รหัส 43010436

นายสันติ จารุสมบัติ

รหัส 43010461



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาเกมส์บนมือถือ

นายศุภชัย หวันแสง	43010436
นายสันติ จารุสมบัติ	43010461
ดร. วรวัฒน์ ลีมีโกคา	อาจารย์ที่ปรึกษา
ปีการศึกษา 2546	

บทคัดย่อ

การพัฒนาเกมส์บนมือถือ มีลักษณะเป็นเกมส์ออนไลน์ 2 มิติ เป็นการพัฒนาโปรแกรมเพื่อความบันเทิง โดยใช้ภาษา J2ME (Java 2 Micro Edition) ในส่วนของ Client และใช้ภาษา ASP ในส่วนของ Server สำหรับการติดต่อระหว่าง Client กับ Server นั้นใช้ Http connection ในการติดต่อโดยผ่านโปรโตคอล GPRS

ลักษณะของเกมส์เป็นเนื้อเรื่องของวรรณคดีไทยเรื่อง พระอภัยมณี ซึ่งจะมีตัวละครที่เป็นตัวเอกหลายตัวเช่น พระอภัยมณี ,ศรีสุวรรณ ,แก้วเกสา ,นางเงือก เป็นต้น โดยตัวผู้เล่นจะสวมบทบาทเป็นตัวละครนั้น เพื่อดำเนินตามเนื้อเรื่องในเกมส์ที่แตกต่างกันในบางช่วงของเกมส์ หลังจากนั้นตัวละครทุกตัวก็จะดำเนินเนื้อเรื่องตามปกติ

โดยเกมส์นี้นั้นนอกจากจะให้ความบันเทิงและผ่อนคลายกับผู้เล่นแล้ว ยังทำให้ผู้เล่นได้มีความรู้ความเข้าใจในวรรณคดีไทย และเป็นการอนุรักษ์วรรณคดีไทยแก่คนรุ่นหลังอีกด้วย

GAME DEVELOPMENT ON MOBILE PHONE

Mr. Suppachai Wunsang 43010436

Mr. Santi Jarusombut 43010461

Dr. Voravat Limpoka Advisor

Academic Year 2003

ABSTRACT

Game development on Mobile phone is two dimension game online. This program is entertainment program which programming with J2ME (Java 2 Micro Edition) in client and ASP in server. Connection between client and server use Http protocol. And use GPRS protocol to connect through wireless network.

Story of game based on RPG game that have detail in Thai literature. As story alias Prapaimanee. Player can select role to playing game as well as Prapaimanee, Seesuwun, Mermaid, Kaewkaesa. So user will act like the choosen character and play along the story. For the special event each character will meet another event after that all character will play and meet the same event.

Not only this game entertain and relax players but also give the Thai literature's knowledge. Next generation people still have knowledge about Thai literature to prevent losing Thai literature.

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สามารถทำได้สำเร็จสมบูรณ์อันเนื่องมาจาก คำแนะนำ คำปรึกษาจากหลายๆฝ่ายด้วยกัน โดยเฉพาะอาจารย์ที่ปรึกษาที่ให้โอกาสให้ข้าพเจ้าได้ทำปริญญานิพนธ์ฉบับนี้ คอยให้ความเอาใจใส่ตลอดจนดูแลสอบถามให้ความช่วยเหลือเสมอมา นั่นก็คือ ดร. วรวัฒน์ ลิ้มโกศา ซึ่งต้องขอบพระคุณเป็นอย่างสูง

รวมทั้งขอขอบคุณทางภาควิชาวิศวกรรมคอมพิวเตอร์ที่ได้จัดเตรียมสิ่งอำนวยความสะดวก เพื่อให้การวิจัยและการพัฒนาโปรแกรมเป็นไปด้วยความสะดวก สะบาย อีกทั้งยังมีอินเทอร์เน็ตให้ค้นคว้าหาความรู้อยู่เสมอๆ ซึ่งจากความสะดวกทั้งหลายเหล่านี้จึงเป็นที่มาของความสำเร็จของโปรเจกต์ดังที่ปรากฏอยู่

ขอขอบคุณทางห้องโถงล่าช้าที่ให้ที่พักพิงสบายๆห้องแอร์เย็นๆที่ใช้ค้นคว้าและผ่อนคลาย รวมถึงเพื่อนๆที่ๆน้องๆที่ให้ความช่วยเหลือและคำแนะนำต่างๆที่ดีๆขอบคุณครับ

จนมาถึงสุดท้ายที่จะขาดไม่ได้ก็คือ บิดา มารดา ของข้าพพระเจ้าที่ให้กำเนิดข้าพเจ้ามาและการให้ความเลี้ยงดูที่อบอุ่น ความดูแลเอาใจใส่ รวมถึงโอกาสทางการศึกษาต่างๆที่มอบให้แก่ข้าพเจ้า ข้าพเจ้าจึงอยากขอบพระคุณไว้ ณ ที่นี้ด้วย

สุภชัย หวันแสง
สันติ จารุสมบัติ

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติประกาศ	III
สารบัญ	IV
สารบัญภาพ	IX
บทที่ 1 บทนำ	1
1.1 หลักการและเหตุผล	1
1.2 วัตถุประสงค์	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	2
1.4 ขอบเขตของการพัฒนา	2
บทที่ 2 เทคโนโลยี WAP และ GPRS	3
2.1 เทคโนโลยี WAP	3
2.1.1 บทนำ	3
2.1.2 จุดกำเนิดของ WAP	3
2.1.3 มาตรฐานสำหรับ WAP	4
2.1.4 WAP กับมาตรฐานอินเทอร์เน็ต	4
2.1.5 คำศัพท์ที่เกี่ยวข้อง	5
2.1.5.1 Micro-Browser	5
2.1.5.2 HSCSD	5
2.1.5.3 GPRS	5
2.1.5.4 EDGE	5
2.1.5.5 WCDMA	6
2.2 เทคโนโลยี GPRS	6
2.2.1 บทนำ	6
2.2.2 GPRS คืออะไร	7
2.2.3 การเชื่อมต่ออินเทอร์เน็ตด้วย GPRS	7
2.2.4 ประโยชน์ของ GPRS	8
2.2.5 ความเร็วของ GPRS	9
บทที่ 3 จาวาสำหรับอุปกรณ์ขนาดเล็กและอุปกรณ์ไร้สาย	10
3.1 บทนำ	10
3.2 ภาพรวมของจาวา	10
3.3 J2ME คืออะไร	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1	เลขอร์จาวาเวอร์ชวลแมชชีน	10
3.3.2	เลขอร์คอนฟิกูเรชั่น	11
3.3.3	เลขอร์โพรไฟล์	11
3.4	คอนฟิกูเรชั่นใน J2ME	11
3.4.1	CDC	11
3.4.2	CLDC	11
3.5	โพรไฟล์ของ J2ME	13
3.6	J2ME สำหรับอุปกรณ์ไร้สาย	13
3.7	ความต้องการของระบบ	15
บทที่ 4	หลักการเขียนโปรแกรมบนอุปกรณ์ไร้สายด้วยจาวา	16
4.1	บทนำ	16
4.2	ไลบรารีของ CLDC	16
4.2.1	ซึบคลาสของ J2SE	16
4.2.2	คลาสที่มีเฉพาะใน CLDC	16
4.2.3	ประเภทของข้อมูลพื้นฐานของ CLDC	17
4.3	ไลบรารีของ MIDP	17
4.3.1	คลาสโปรแกรมจัดการแอปพลิเคชัน	17
4.3.2	คลาสของส่วนติดต่อกับกราฟิกกับผู้ใช้ (GUI Class)	17
4.3.3	คลาสของพื้นที่เก็บข้อมูลแบบคงตัว (Persistent Storage Class)	18
4.3.4	คลาสของเครือข่าย	18
4.4	MIDlet	18
4.4.1	วงจรการทำงานของ MIDlet	19
4.5	ซอฟต์แวร์จัดการแอปพลิเคชัน (Application Management Software)	20
4.6	ข้อจำกัดต่าง	20
4.6.1	ข้อจำกัดในการจัดการข้อผิดพลาด CLDC	20
4.6.2	ข้อจำกัดของ KVM	21
4.7	ระบบความปลอดภัยใน J2ME	21
4.8	การเวริฟายและฟรีเวริฟายไฟล์คลาส	21
บทที่ 5	คอมโพเนนท์หลักของส่วนติดต่อกับผู้ใช้ใน J2ME	22
5.1	บทนำ	22
5.2	Displayable และ Display	22
5.2.1	Displayable	22
5.2.1.1	Screen	23
5.2.1.2	Canvas	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2 Display	24
5.3 รูปภาพ (Image)	25
5.3.1 รูปที่แก้ไขไม่ได้	25
5.3.2 รูปที่แก้ไขได้	25
5.4 อีเวนต์และการจัดการอีเวนต์	25
บทที่ 6 การใช้ API ระดับสูงของ J2ME	26
6.1 บทนำ	26
6.2 List และ Choice	26
6.2.1 List	26
6.2.2 Choice	26
6.2.2.1 Exclusive-choice	26
6.2.2.2 Implicit-choice	26
6.2.2.3 Multiple-choice	26
6.3 Textbox	27
6.4 Alert	28
6.5 Form และ Item	28
6.5.1 Item	28
6.5.1.1 ChoiceGroup	29
6.5.1.2 ImageItem	29
6.5.1.3 StringItem	29
6.5.1.4 Gauge	29
6.5.1.5 DateField	29
6.5.1.6 TextField	29
6.5.2 Form	29
6.6 อีเวนต์ระดับสูงและการจัดการอีเวนต์	30
6.6.1 Command	30
6.6.2 ItemStateChange	30
บทที่ 7 หลักการเขียนโปรแกรมเครือข่ายของ J2ME	31
7.1 บทนำ	31
7.2 การเขียนโปรแกรมเครือข่ายด้วย J2SE และ J2ME	31
7.3 กรอบการติดต่อสื่อสารทั่วไป (The Generic Connection Framework)	31
7.4 การเขียนโปรแกรมเครือข่ายไร้สายโดยใช้ซ็อกเก็ต	32
7.5 การเขียนโปรแกรมเครือข่ายไร้สายโดยใช้ดาต้าแกรม	33
7.6 การเขียนโปรแกรมเครือข่ายไร้สายโดยใช้ HttpConnection	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.6.1 การใช้เมธอด GET กับ HttpConnection	34
7.6.2 การใช้เมธอด HEAD กับ HttpConnection	35
7.6.3 การใช้เมธอด POST กับ HttpConnection	35
บทที่ 8 หลักการเขียน ASP (Active Server Page)	36
8.1 บทนำ	36
8.2 ความรู้เบื้องต้นของภาษา HTML	36
8.3 ความรู้เบื้องต้นของ VBScript	37
8.4 ความรู้เบื้องต้นของ ASP	39
บทที่ 9 การออกแบบเซิร์ฟเวอร์	41
9.1 การทำงาน	41
9.2 ลักษณะของการติดต่อ	42
9.3 การออกแบบฐานข้อมูล	42
9.4 การออกแบบหน้าเว็บเพจ	44
9.5 การทำงานของเว็บเพจแต่ละหน้า	44
9.5.1 NEWID.ASP	44
9.5.2 NEWNAMETYPE.ASP	45
9.5.3 CHKCON.ASP	46
บทที่ 10 การออกแบบไคลเอนต์	48
10.1 การออกแบบลักษณะเกม	48
10.1.1 รูปแบบของเกม	48
10.2 การทำงานของคลาสต่างๆ	49
10.2.1 Prapaimanee	49
10.2.2 Mycanvas	49
10.2.3 TestTimer	49
10.2.4 Apaimanee, Fish, Seesuwun, Kaewkaesa	50
10.2.5 Attack	50
10.2.6 Background	50
10.2.7 Monster	50
10.2.8 MapGame	51
10.2.9 Random	51
10.3 เทคนิคต่างๆที่นำมาใช้ในเกม	51
10.3.1 เทคนิคการวาดแผนที่	51
10.3.2 เทคนิคการทำภาพแอนิเมชันการใช้ท่า	53
10.3.3 เทคนิคการแสดงการเคลื่อนไหวของตัวละคร	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.3.4 เทคนิคในการสนทนาซื้อของกับพ่อค้าในกรม	56
10.4 เครื่องมือที่ใช้ในการพัฒนา	57
10.5 การติดต่อไปยังเครื่องเซิร์ฟเวอร์	60
บทที่ 11 การทดสอบการทำงาน	65
11.1 วิธีการทดสอบ	65
11.2 ผลการทดสอบ	65
บทที่ 12 บทวิจารณ์และสรุป	75
12.1 สรุปผลการทดลอง	75
12.2 แนวทางในการพัฒนาต่อ	75
บรรณานุกรม	77
ภาคผนวก	78



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่	หน้าที่
รูปที่ 2.1 การเชื่อมต่อของ GPRS	8
รูปที่ 2.2 วิวัฒนาการของระบบ GSM	9
รูปที่ 3.1 ความสัมพันธ์ระหว่าง J2SE และ คลาสไลบรารีใน CDC และ CLDC	12
รูปที่ 3.2 โครงสร้างของ MIDP	14
รูปที่ 4.1 วงจรการทำงานของ MIDlet	19
รูปที่ 5.1 โครงสร้างลำดับชั้นของคลาส Displayable	23
รูปที่ 6.1 List แบบ Implicit- choice	27
รูปที่ 6.2 List แบบ exclusive- choice	27
รูปที่ 6.3 List แบบ multiple – choice	27
รูปที่ 6.4 โครงสร้างลำดับชั้นของคลาส Item	29
รูปที่ 9.1 การติดต่อจากไคลเอนต์ไปยังเซิร์ฟเวอร์เพื่อที่จะ โหลดและเซฟเกมส์	41
รูปที่ 9.2 การติดต่อจากเครือข่ายไร้สาย ไปยังเครือข่ายอินเทอร์เน็ต	42
รูปที่ 9.3 ฟิลด์ต่างๆที่ใช้ในการเก็บข้อมูล	43
รูปที่ 10.1 ภาพแผนที่ใหญ่เปรียบเทียบกับส่วนที่หน้าจอมองเห็นได้	51
รูปที่ 10.2 A:วิธีการวาดในหนึ่งหน้าจอ B:ภาพหลังจากการวาด	52
รูปที่ 10.3 ภาพในแต่ละเฟรม	53
รูปที่ 10.4 แอนิเมชันที่ปรากฏในเกม	53
รูปที่ 10.5 ท่าทางต่างๆของตัวละครชื่อนางเงือก	55
รูปที่ 10.6 A: ภาพการพูดคุย B: ภาพที่เห็นในเกม	56
รูปที่ 10.7 โปรแกรม Adobe PhotoShop	57
รูปที่ 10.8 โปรแกรม J2ME Wireless Toolkit	58
รูปที่ 10.9 อุปกรณ์จำลอง	58
รูปที่ 10.10 โปรแกรม IIS	59
รูปที่ 11.1 หน้าจอหลักเมื่อเริ่มต้นเข้าสู่เกมส์	65
รูปที่ 11.2 หน้าจอเมื่อเริ่มต้นเข้าสู่เกมส์	66
รูปที่ 11.3 หน้าจอเมื่อเลือก Instruction เพื่อดูวิธีการเล่น	66
รูปที่ 11.5 หน้าจอเมื่อต้องการสมัครตัวละครใหม่	66
รูปที่ 11.6 หน้าจอเมื่อมีการป้อนข้อมูลที่ไม่ถูกต้องตามเงื่อนไข	67
รูปที่ 11.7 หน้าจอเพื่อเลือกตัวละคร	67
รูปที่ 11.8 หน้าจอเพื่อป้อนชื่อของตัวละคร	68
รูปที่ 11.9 หน้าจอสถานะของตัวละคร	68
รูปที่ 11.10 หน้าจอปกติในเกมส์	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 11.11 หน้าจอบนแผนที่ปกติ	69
รูปที่ 11.12 หน้าจอเมื่อเข้าจากต่อสู้	70
รูปที่ 11.13 หน้าจอเมื่อสามารถเอาชนะศัตรูได้	70
รูปที่ 11.14 หน้าจอการซื้อของจาก NPC	71
รูปที่ 11.15 หน้าจอเมื่อจำนวนเงินไม่พอ	71
รูปที่ 11.16 หน้าจอเมื่อเราแพ้ต่อศัตรู	72
รูปที่ 11.17 หน้าจอการเลือก Continue	72
รูปที่ 11.17 หน้าจอการป้อนข้อมูลเมื่อ Continue	73
รูปที่ 11.18 หน้าจอสถานะของตัวละคร	73
รูปที่ 11.19 หน้าจอการดำเนินตามเนื้อเรื่องของแก้วเกศา	74



บทที่ 1

หลักการและเหตุผล

1. หลักการและเหตุผล

เนื่องจากในปัจจุบันนี้การพัฒนาซอฟต์แวร์นั้นได้มีการพัฒนาในรูปแบบด้านต่างๆมากมายเช่น ด้านโปรแกรมเพื่อการใช้งาน ด้านการติดต่อสื่อสาร ด้านโปรแกรมประเภทกราฟิกและแอนิเมชัน และในด้านความบันเทิง ซึ่งการพัฒนานั้นจะเกิดขึ้นบนเครื่องคอมพิวเตอร์ส่วนบุคคลที่มีหน่วยความจำที่มาก และส่วนประมวลผลที่มีความเร็วสูง

ดังนั้นจึงต้องการจะนำความสามารถต่างๆเหล่านั้นมาทำงานอยู่บนอุปกรณ์ที่มีขนาดเล็กซึ่งจำกัดไปด้วยหน่วยความจำที่น้อยและความเร็วของส่วนประมวลผลที่ยังช้าอยู่แต่มีข้อดีที่ว่าสามารถเคลื่อนที่ได้สะดวก พกพาได้ง่ายเนื่องจากมีขนาดเล็ก ซึ่งอุปกรณ์ที่จะเลือกพัฒนาในที่นี้คือ มือถือนั่นเอง

ซึ่งซอฟต์แวร์ที่ต้องการจะพัฒนาก็คือ ด้านความบันเทิง เหตุผลที่ต้องเป็นเพื่อความบันเทิงนั้นคือ มือถือนั้นเป็นอุปกรณ์ที่ต้องพกพาไปไหนมาไหนด้วยตลอดทั้งนั้น เมื่อมีเวลาว่างผู้เป็นเจ้าของมือถือก็สามารถที่จะผ่อนคลายได้ รวมไปถึงการพัฒนาซอฟต์แวร์ประเภทนี้นั้นสามารถใช้เป็นพื้นฐานขั้นต้นในการที่จะพัฒนาซอฟต์แวร์ขั้นสูงต่อไป

2. วัตถุประสงค์

1. เพื่อศึกษาการทำงาน โปรแกรมและการพัฒนาโปรแกรมด้วย J2ME บนโทรศัพท์มือถือ
2. เพื่อศึกษาทักษะในการเขียน โปรแกรมในด้านอัลกอริทึม
3. เพื่อพัฒนาบุคลากรทางด้านอุตสาหกรรมซอฟต์แวร์
4. เพื่อนำความรู้และทักษะที่ได้ นำไปสร้างประโยชน์แก่สังคมและประเทศชาติ
5. เพื่อศึกษาการเขียน โปรแกรมด้วย J2ME เพื่อนำไปใช้ในการพัฒนาเกมส์และโปรแกรมต่างๆ บนอุปกรณ์โทรศัพท์
6. เพื่อรู้จักใช้เวลาว่างให้เกิดประโยชน์ ทั้งทางด้านความรู้ และ การทำงานเป็นกลุ่มมีความรับผิดชอบและรู้จักการทำงานอย่างเป็นระบบ
7. เพื่อกระตุ้นคนไทยให้มีการรวมตัวกันเพื่อพัฒนาซอฟต์แวร์ เกมส์และโปรแกรมต่างๆมากขึ้น
8. เพื่อนำความรู้ที่ได้จากการเขียนเกมมาประยุกต์ให้เข้ากับการเขียน โปรแกรมประเภทอื่นๆต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ประโยชน์ที่คาดว่าจะได้รับ

ในส่วนของโปรแกรมที่สร้างสามารถแบ่งได้ออกเป็น 3 ส่วน

1. ส่วนของการติดต่อกับ network โดยส่วนของการติดต่อกับเครื่องเซิร์ฟเวอร์เพื่อที่จะเซฟสถานะของตัวละครเดิมของผู้เล่นนั้นสามารถนำส่วนนี้ไปใช้ประโยชน์เพื่อที่จะนำข้อมูลที่สำคัญต่างๆไปเก็บได้เพื่อที่จะนำข้อมูลที่สำคัญนั้นไปใช้จากที่อื่นได้โดยไม่มีข้อจำกัดว่าต้องเป็นเครื่องที่เราทำงานอยู่เท่านั้นจึงจะสามารถทำงานต่อได้เราสามารถดึงข้อมูลเหล่านี้มาใช้งานกับ Device อื่นที่มี Application เดียวกันได้

2. ส่วนของ High-Level API อินเทอร์เฟซระดับสูงซึ่งในส่วนนี้เป็นการติดต่อกับผู้เล่นเกมในตอนเริ่มแรกเพื่อที่จะป้อน ID และ Password ของตัวผู้เล่นเพื่อที่จะโหลดสถานะเดิมมาเพื่อเล่นต่อในส่วนนี้สามารถนำไปประยุกต์ใช้งานกับการป้อนข้อมูลต่างๆของผู้ใช้งาน application นั้นๆเพื่อนำข้อมูลเหล่านั้นไปใช้ประโยชน์ต่อไป

3. ส่วนของ Low-Level API อินเทอร์เฟซระดับต่ำเป็นซึ่งเป็นส่วนหลักของเกมในการทำ animation ต่างๆของเกมซึ่งในส่วนนี้จะสามารถนำไปประยุกต์ต่อไปในการที่จะทำด้านกราฟิกซึ่งในส่วนนี้ผู้เขียน application สามารถออกแบบทุกอย่างเองได้ตามใจชอบ

4. ขอบเขตของการพัฒนา

ซึ่งการพัฒนานั้นแบ่งออกเป็นส่วนต่างๆดังนี้

- 4.1 Client โดยในส่วนนี้การพัฒนานั้นจะพัฒนาในรูปแบบของแอปพลิเคชันในมือถือซึ่งการพัฒนานั้นใช้ภาษา J2ME (Java to Micro Edition) ซึ่งในส่วนนี้จะเป็นส่วนเกิดการเล่นเกมจริงๆและเป็นส่วนที่ติดต่อกับผู้ใช้ซึ่งส่วนแสดงผลคือหน้าจอมีถือนั่นเอง และส่วนที่ทำการติดต่อเพื่อรับคำสั่งจากผู้ใช้คือ ปุ่มกดต่างๆบนเครื่องโทรศัพท์นั่นเอง
- 4.2 Server ในส่วนนี้นั้นการพัฒนาจะใช้ภาษา ASP (Active Server Page) ในการเขียน Web Server โดยในส่วนของเซิร์ฟเวอร์จะทำงานในการติดต่อกับฐานข้อมูลเพื่อดึงข้อมูลต่างๆเพื่อให้ส่งให้กับ Client
- 4.3 Database จะใช้ Microsoft Access ในการเก็บข้อมูล
- 4.4 Connection ระหว่าง server กับ client นั้นใช้โปรโตคอล Http การติดต่อกับ Web Server ซึ่งใช้โปรโตคอล GPRS ในการส่งผ่านเครือข่ายไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

เทคโนโลยี WAP และ GPRS

2.1 เทคโนโลยี WAP

2.1.1 บทนำ

จากความเจริญเติบโตอย่างรวดเร็วของทั้ง อินเทอร์เน็ต และ การรวมเอา ความสามารถเชิงอินเทอร์เน็ต มาใช้กับเทคโนโลยี ของโทรศัพท์มือถือทำให้ ณ ปัจจุบันนี้ โทรศัพท์มือถือที่เราใช้กันอยู่ ทำอะไรๆ ได้มากกว่า แค่ใช้โทรออกหรือรับสาย เข้าเท่านั้นแล้ว ผู้คนเริ่มให้ความสนใจ และให้ความสำคัญมากขึ้น กับสิ่งที่จะ ช่วยอำนวยความสะดวกสบาย ในการที่จะ ติดต่อค้นหาข้อมูล และบริการอื่นๆ บนอินเทอร์เน็ต ซึ่งสามารถ ทำได้แม้แต่ในขณะที่กำลังเดินทางอยู่ โดยผ่านทางโทรศัพท์ มือถือเพียงเครื่องเดียว แต่จะอย่างไรล่ะ ถึงจะนำเอา ความสามารถต่างๆ ที่อยู่บน อินเทอร์เน็ต เหล่านั้นจับมาใส่ลง ในเครื่องโทรศัพท์มือถือได้ และ ยังต้องทำให้ใช้งานได้ง่ายด้วย เนื่องจากโทรศัพท์มือถือ นั้น มีข้อจำกัดในด้านการป้อนข้อมูล เข้า และการแสดงผล Wireless Application Protocol หรือ WAP จึงได้ถือกำเนิดขึ้น เพื่อช่วยให้ ผู้ใช้โทรศัพท์มือถือ สามารถใช้ความสามารถดังกล่าวข้างต้นเหล่านี้ได้

2.1.2 จุดกำเนิดของ WAP

WAP (Wireless Application Protocol) เป็นโปรโตคอลที่เป็นมาตรฐานสากล ที่เกิดจากความร่วมมือกันของ หลายบริษัท ผู้ผลิตโทรศัพท์มือถือ เพื่อนำเอาลูกเล่นหรือ ความสามารถ ต่างๆ ของ แอปพลิเคชันไร้สาย (Wireless Application) และ ของทางด้านอินเทอร์เน็ตให้มาใช้ได้ บนเครื่องโทรศัพท์มือถือ WAP จะทำให้ผู้ใช้โทรศัพท์มือถือ ใช้งานทางด้านอินเทอร์เน็ตทั่วๆ ไปได้ เหมือนกับใช้งานผ่านทาง เครื่องคอมพิวเตอร์ โดยที่ WAP นั้นไม่ต้องการ CPU ที่มีประสิทธิภาพ สูงๆ ไม่ต้องการหน่วยความจำมากๆ ไม่ต้องการแหล่งพลังงานมาก แต่ก็มีข้อจำกัดอยู่หลายเรื่องด้วยกันที่อาจทำให้ ผู้ใช้รู้สึกแปลกและ ตัดขาดบ้าง เช่น ข้อจำกัดด้านแบนด์วิดท์ (Bandwidth), มีจอแสดงผล (Display) ขนาดเล็ก และ มีส่วนของการป้อนข้อมูลเข้า (Input) ที่แตกต่างจากการใช้งาน บนคอมพิวเตอร์อยู่พอสมควร แต่จุดเด่นของ WAP นั้น ก็คือทำให้ ใช้งานได้ผ่านทางโทรศัพท์มือถือ ซึ่งสามารถพกพา หรือนำไปใช้งาน ณ ที่ไหนๆ ก็ได้ ซึ่งจุดนี้ ก็น่าจะหักล้างกับข้อจำกัดต่างๆ ได้

WAP นั้นถูกออกแบบมาเพื่อให้ใช้กับอุปกรณ์ไร้สายต่างๆ ไม่จำกัดเพียงแค่โทรศัพท์มือถือเท่านั้น หากยังรวมไปถึง วิทยุติดตามตัว (Pager), วิทยุรับส่งที่เรียกว่า Two-Way Radio, Smartphone และรวมไปถึงอุปกรณ์สื่อสารต่างๆ ตั้งแต่ระดับ Low-End จนถึง High-End เลยทีเดียว ซึ่ง ระบบเครือข่ายที่ใช้กับ WAP ได้นั้น ก็ใช้ได้หลากหลายรูปแบบ ทั้ง CDPD, CDMA, GSM, PDC, PHS, TDMA, FLEX, ReFLEX, iDEN, TETRA, DECT, DataTAC และ Mobitex. WAP เป็นโปรโตคอลสำหรับการสื่อสาร ซึ่งสามารถใช้งานได้กับระบบปฏิบัติการ (OS : Operating) ต่างๆ ได้หลากหลาย ทั้ง PalmOS, EPOS, Windows CE, FLEXOS, OS/9, JavaOS และอื่นๆ อีก WAP นั้นจะช่วยสนับสนุน Bearer หลักๆ ในการส่ง Message เช่น Short Message Service (SMS) , Circuit Switched Data, Unstructured Supplementary

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ทางการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Services Data (USSD) และ General Packet Radio Services (GPRS) อีกด้วย และ นอกจากนี้ ก็ยังมีการนำเอาเทคโนโลยีใหม่ๆ ต่างๆ มาใช้กับโทรศัพท์มือถือ มากขึ้น โดยอาศัย WAP และ เทคโนโลยีที่เกี่ยวกับการส่ง Message เข้ามาช่วยในการทำงานอีกด้วย เช่น HSCSD, EDGE และ WCDMA

2.1.3 มาตรฐานสำหรับ WAP

WAP Version 1.1 นั้น ได้ทำการแก้ไขปรับปรุงจาก Version 1.0 โดยที่ยังคง Function ต่างๆ ไว้เช่นเดิม เพียงแต่ Version 1.0 นั้น ยังเป็นเพียงมาตรฐานที่ใช้ทดลองกันภายในกลุ่มบริษัท ผู้พัฒนาเพื่อทำการแก้ไขมาตรฐานบางอย่าง หลังจากนั้นจึงวางเป็น มาตรฐาน Version 1.1 เพื่อใช้เป็นสากลต่อไป โดยที่จุดที่ทำการ แก้ไขนั้น หลักๆ ก็คือทำการแก้ไขในส่วนของ WML (Wireless Markup Language) เพื่อให้ใช้ได้กับ มาตรฐาน XHTML ที่กำลังจะเกิดขึ้นต่อไปนั่นเอง

สำหรับ WAP Version 1.2 นั้น ก็ยังคงใช้ Function อะไรๆต่างๆ เหมือนกับ Version 1.1 แต่ก็ได้ใช้ข้อมูล ความเห็นต่างๆของ ผู้ทดสอบผู้พัฒนาในกลุ่มบริษัท มาปรับปรุงแก้ไข และได้เพิ่มเติม ในส่วนของ WAP Push Architecture ซึ่งเป็นภาคขยายของ Wireless Telephony Applications (WTA) นอกจากนี้ก็ ยังได้ เพิ่มเติมให้รองรับกับเทคโนโลยีทางด้าน ระบบเครือข่ายให้มากขึ้นอีกด้วย

สำหรับอุปกรณ์ Wireless ที่รองรับ หรือใช้งานกับมาตรฐาน WAP ได้ นั้น ก็เริ่มทยอยออกมาสู่ท้องตลาดบ้างแล้ว โดยอุปกรณ์ หลักๆ ก็คือ โทรศัพท์เคลื่อนที่ ที่รองรับมาตรฐาน WAP หรือที่เรียกว่า WAP Phone ก็ได้เริ่มวางตลาดแล้วตั้งแต่ปลายๆ ปีที่แล้ว และ ก็คาดว่า ในอนาคตจะมีการจำหน่าย WAP Phone หลากหลาย รุ่นและยี่ห้อมากขึ้น

2.1.4 WAP กับมาตรฐานอินเทอร์เน็ต

WAP นั้นจริงๆ แล้วก็ได้พัฒนาขึ้นมาจากมาตรฐานอินเทอร์เน็ตนี่เอง ซึ่งก็ใช้พื้นฐานเดียวกับอินเทอร์เน็ตคือ XML, UPD และ IP ซึ่ง WAP Wireless Protocol นี้ ก็ยังได้พัฒนาโดย ใช้มาตรฐานของอินเทอร์เน็ตพวก HTTP และ TLS ด้วย เพียงแต่ ได้ทำการปรับปรุงแก้ไขบางส่วน เพื่อให้ใช้งานได้ด้วยอุปกรณ์ ไร้สาย (Wireless Device) ซึ่งมีข้อจำกัด ทั้งทางด้าน หน่วยแสดงผล (Display) ซึ่งมีขนาดเล็กหรือในส่วนของ การป้อนข้อมูลเข้าที่ยากกว่า การใช้เครื่อง PC ด้วยเช่นกัน WML จึงเป็นผลที่เกิดขึ้นจากการปรับปรุงแก้ไข มาตรฐานของการ เขียน Markup Language ซึ่งไม่สามารถใช้ HTML บนอุปกรณ์ไร้สาย จากเหตุผลข้างต้น

XML หรือ Extend Markup-Languaage นั้น เป็น Meta-Languaage ซึ่งกำหนดขึ้นโดย W3C (World Wide Web Consortium) ซึ่งหากจะมองว่าเป็นส่วนขยายของ SGML (Standard Generalized Makriup Language) ก็ได้ เพียงแต่ XML นี้ เป็นเพียงกฎ เงื่อนไข ต่างๆ ในการสร้างมาตรฐาน ภาษาที่ใช้ในการเขียนแอปพลิเคชันเท่านั้น คือยังไม่กำหนดรูปแบบ ต่างๆ แน่ชัด ส่วน WML หรือ HTML นั้น มีการกำหนดรูปแบบต่างๆ Tag คำสั่งต่างๆ เป็นที่แน่ชัดแล้ว และมีการเจาะจง ให้ใช้ในด้านใดด้านหนึ่ง ดังนั้นจึงมองได้ว่า WML หรือ HTML นั้น เป็น XML Application

นอกจากนี้ WML ก็ยังมี WMLScript ซึ่งก็อาศัยหลักการ และโครงสร้าง ที่คล้ายกับ JavaScript หรือ ECMAScript เพียงแต่ทำการ ปรับปรุงให้ใช้งานได้เหมาะสมกับระบบ อุปกรณ์ไร้สายด้วย โดยเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WMLScript นี้ จะใช้หน่วยความจำ และ CPU ต่ำมาก ช้ายัง ได้ตัดเอาฟังก์ชันต่างๆ ที่ไม่ เกี่ยวข้อง หรือไม่ มีความจำเป็น ต้องใช้กับในระบบนี้ออก ไปอีกไม่น้อย

นอกจากนี้ WAP ก็ยัง ได้รวมเอาความสามารถ เพื่อช่วย ในการรักษาความปลอดภัย ของข้อมูล ทั้ง ด้านการ การพิสูจน์ตน (Authentication) และ ด้านการ เข้ารหัสข้อมูล (Encryption) ซึ่งเรียกว่า WTLS อีก ด้วย

2.1.5 คำศัพท์ที่เกี่ยวข้อง

2.1.5.1 Micro-Browser

เป็น Software ทางฝั่งไคลเอนต์ (Client Software) ที่ฝังอยู่ในตัวซิม (SIM) ซึ่ง ออกแบบมา เพื่อให้เครื่องอุปกรณ์มือถือ สามารถใช้งานกับบริการด้านอินเทอร์เน็ตได้

2.1.5.2 HSCSD

หรือ High Speed Circuit Switched Data จะช่วยลดเวลาในการส่งถ่ายข้อมูล ระหว่าง โทรศัพท์มือถือกับอินเทอร์เน็ตและทำให้ช่วย การส่งถ่ายข้อมูลขนาดใหญ่ๆ นั้นทำได้แทบจะทันที พร้อมทั้งรับประกันความถูกต้องของข้อมูลที่มีการส่งถ่าย ออกไปอีกด้วย ด้วยความรวดเร็วในการส่งถ่ายข้อมูล ระดับเดียวกันกับ ISDN ขวกับความสะดวกสบายต่างๆ ก็น่าจะช่วยดึงดูดให้มีผู้มาใช้งานด้านนี้ มากขึ้น และแน่นอนเมื่อมีผู้ใช้งานมากขึ้น ก็จะทำให้บริษัทมีรายได้เพิ่มมากขึ้นด้วย

2.1.5.3 GPRS

หรือ General Packet Radio Service ซึ่งเป็นตัวสร้างการเชื่อมต่อโดยตรงระหว่างโทรศัพท์มือถือ และ อินเทอร์เน็ต เข้าด้วยกัน ซึ่ง GPRS นี้ จะทำให้สามารถ มีอัตราในการส่งถ่ายข้อมูล ได้มากที่สุดถึง 100 kbps เลยทีเดียว GPRS นั้น จะสร้าง IP ให้อย่างต่อเนื่อง และมี แบนด์วิดท์ กว้าง ทำให้สามารถนำเอา แอปพลิเคชันต่างๆ มาใช้ได้อย่างหลากหลาย อาทิ Multimedia Messaging, Wireless Corporate Intranet, Remote Control และ อื่นอีกมากมาย โดยผู้ใช้ จะเสียค่าใช้จ่าย เพียงแค่ในส่วนของจำนวนข้อมูลที่มีการส่ง ถ่ายเท่านั้น ทั้ง HSCSD และ GPRS จะทำงานในลักษณะเกี่ยวพันกัน เพื่อให้ทำงานได้อย่าง กว้างขวาง มากยิ่งขึ้น โดยแต่ละตัวจะสนับสนุน แอปพลิเคชัน ที่แตกต่างกัน ซึ่งแต่ละ แอปพลิเคชันนั้นก็เป็นส่วน เดิมเติมให้กันและกัน เพื่อนำไปสู่ยุคใหม่ (ยุคที่ 3) ของระบบ Cellular Radio (3G : 3rd Generation) ที่ กำลังจะมาถึงในอนาคตอันใกล้นี้

2.1.5.4 EDGE

หรือ Enhanced Data rates for Global Evolution ซึ่งคาดว่าจะมีใช้อย่าง แพร่หลายในปี 2001 จะ ช่วยให้โทรศัพท์มือถือสามารถให้บริการด้าน Multi-Media ต่างๆ ได้มากยิ่งขึ้น เช่น วีดีโอคอนเฟอเรนซ์ (VDO Conferent) , การถ่ายโอนข้อมูลขนาดใหญ่ๆ , การต่อเข้ากับ อินเทอร์เน็ต ผ่านทางเครื่อง โทรศัพท์มือถือเอง หรือผ่านทางเครื่อง โน้ตบุ๊ก หรือ เครื่องคอมพิวเตอร์ส่วนบุคคล ด้วย EDGE นี้ ผู้ใช้บริการจะสามารถจัดส่ง VDO-Postcard ให้แก่เพื่อนๆ หรือครอบครัวผ่านทางอินเทอร์เน็ตด้วย เครื่องโทรศัพท์ มือถือเพียงเครื่องเดียว ได้อย่างไม่ยากเย็น เทคโนโลยีของ EDGE นี้ ใช้ได้ทั้งระบบ GSM และ TDMA ซึ่งก็เป็นระบบดิจิทัล (Digital) ด้วยกันทั้งคู่ แต่แตกต่างกันเรื่องระบบ Cellular และ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างภายใน โดย EDGE นั้นสามารถใช้ได้กับในระบบเครือข่ายและความถี่ของสัญญาณที่ใช้ในปัจจุบันนี้ได้ เพียงแต่อาจต้องมีการแก้ไขฮาร์ดแวร์ (Hardware) ด้าน Network และ แก้ไข ซอฟต์แวร์ บ้างเล็กน้อย

2.1.5.5 WCDMA

หรือ Wideband Code Division Multiple Access โดยที่ Wideband นั้นจะมีความสามารถอยู่ระหว่าง Narrowband และ Broadband Narrowband คือความสามารถเชิงโทรคมนาคม ที่รองรับการส่งข้อมูลไม่กี่จำพวก เช่น ข้อมูลเสียง,รูปภาพที่ถอดแบบจากต้นฉบับ, Slow-Scan VDO และ ข้อมูลที่มีการส่งถ่ายด้วยความเร็วต่ำๆ ในระดับ Kilobit เท่านั้น โดยคุณภาพนั้นก็อยู่ใน ระดับที่ยอมรับได้ ต่างกับ Broadband ที่สามารถจัดส่งข้อมูลได้ในระดับ Megabit ซึ่งทำให้สามารถส่งข้อมูลพวก Full-Color Image, หรือ Full-Motion VDO ได้ โดยที่ความเร็วในการส่งถ่ายโดยรวมจะเร็วกว่า แบบ Narrowband ตั้งแต่ 100 ถึง 1,000 เท่าเลยทีเดียว โดยสรุป Wideband นั้นก็จะสามารถรองรับการส่งถ่าย ข้อมูลได้ถึงระดับเม็กกะบิต ซึ่งก็สามารถถ่ายโอนข้อมูลจำพวกภาพเคลื่อนไหวได้ หรือ ทำให้การดาวน์โหลดข้อมูล ข่าวสารต่างๆ เป็นไปได้อย่างรวดเร็ว WCDMA นั้นกล่าวโดยรวมก็คือ เป็นโหมดในการเข้าถึงข้อมูล ที่ใช้สำหรับ Radio Interface ของ 3G (3rd Generation ของ Cellular Radio) ลักษณะพิเศษของมันคือ จะมีความจุของข้อมูลสูง , มี Cell Radius ขนาดเล็ก และ มีการส่งถ่ายของ Spectrum Radio แบบกระจายเป็นวงกว้าง

2.2 เทคโนโลยี GPRS

2.2.1 บทนำ

หลังจากที่วงการ โทรศัพท์เคลื่อนที่ ได้มีการพัฒนาด้านการสื่อสารข้อมูลผ่าน โทรศัพท์มือถือและบริการที่ไม่เกี่ยวกับเสียงการสนทนา (None Voice Application) อย่างต่อเนื่องเพื่อให้สามารถสื่อสารได้ทุกรูปแบบอย่างไร้ขีดจำกัดในระหว่างเคลื่อนที่ ไม่ว่าจะเป็นการสื่อสารด้วยเสียงหรือข้อมูล ดังนั้นผู้ให้บริการเครือข่ายโทรศัพท์เคลื่อนที่จึงได้พัฒนาและนำเทคโนโลยีอย่างที่เราเห็นกันทุกวันนี้เป็นอย่างเป็นขั้นเป็นตอน เช่น

1. Short Message Service(SMS)
 - การใช้เทคโนโลยี SMS หรือการส่งข้อความที่กำลังได้รับความนิยมกันทั่วไปมากขึ้นทุกวันในบ้านเราขณะนี้
 - Sim Tool Kit โดยใช้ Sim Card ที่ทางผู้ให้บริการ โทรศัพท์เคลื่อนที่ ได้พัฒนาและเพิ่มเติมบริการไว้ให้ใช้งานและบริการต่าง ๆ ง่ายมากยิ่งขึ้น
2. Circuit Switched Data (CSD)
 - WAP หรือ Wireless Application Protocol ที่สามารถ Connect กับโลกของข่าวสารข้อมูลกับ Wap Site ต่าง ๆ ได้ทั่วโลกแม้กระทั่งในรูปแบบของอินเทอร์เน็ตไร้สาย (Wireless Internet)

แต่อย่างไรก็ตามทางผู้ให้บริการ โทรศัพท์เคลื่อนที่ก็ยังสังเกตเห็นว่า การโอนถ่ายสื่อสารข้อมูลของ โทรศัพท์มือถือเคลื่อนที่ ยังมีข้อจำกัดในด้านความเร็วการรับส่ง และรวมไปถึงปริมาณข้อมูลที่สามารถทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับจึงได้เริ่มพัฒนาแก้ไขเพื่อที่จะเพิ่มเติมบริการตรงส่วนบกพร่องนี้ให้ดีขึ้น จึงได้เริ่มนำเทคโนโลยีที่เรียกว่า GPRS (General Packet Radio Service)

2.2.2 GPRS คืออะไร

GPRS ย่อมาจาก " General Packet Radio Service" ซึ่งเป็นการส่งข้อมูลต่าง ๆ ในรูปแบบแพ็กเก็ตเกิดต่าง ๆ การเชื่อมต่อแบบใหม่ที่ใช้ระบบGPRSเข้ามา ก็จะเป็นการเชื่อมต่อและวิธีการส่งข้อมูลที่มีลักษณะเช่นเดียวกับอินเทอร์เน็ตก็คือ เมื่อต้องการข้อมูลหรือส่งข้อมูลอะไรก็ได้แล้วแต่ ก็จะเป็นการส่งข้อมูลลักษณะนั้น เข้าไปในเครือข่ายโทรศัพท์เคลื่อนที่เท่านั้นไม่จำเป็นต้องจองเวลาไว้ตลอดเวลา จึงทำให้วิธีการใช้งานของ GPRS ในแบบใหม่นี้ จะเห็นได้ว่าจะมีการพูดถึงการเก็บเงินที่เป็นจำนวนข้อมูลที่รับและส่งออกมามากกว่าวิธีการติดต่อสื่อสารจากวิธีเดิมที่คิดจำนวนเวลาในการติดต่อสื่อสารแต่ละครั้ง

การติดต่อด้วยระบบ GPRS ยังสามารถติดต่อสื่อสารด้วยเสียง ในขณะที่เราสามารถติดต่อสื่อสารผ่านโลกอินเทอร์เน็ตในขณะเดียวกัน ซึ่งก็คือ เราสามารถติดต่อสื่อสารทั้ง 2 ระบบ ภายในช่วงเวลาเดียวกัน แต่ทั้งนี้ต้องขึ้นอยู่กับโทรศัพท์เคลื่อนที่ในแต่ละรุ่นที่ผลิตออกมา แต่เท่าที่ทราบในขณะนี้ โทรศัพท์เคลื่อนที่แต่ละรุ่นยังไม่สามารถติดต่อสื่อสารพร้อม ๆ กันได้

2.2.3 การเชื่อมต่ออินเทอร์เน็ตด้วย GPRS

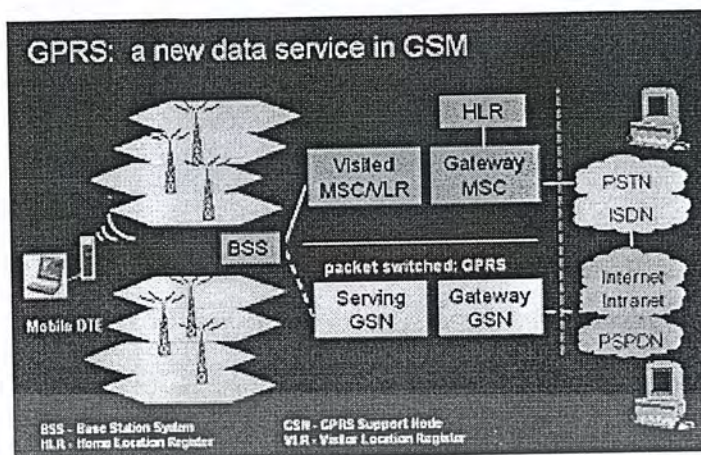
GPRS ไม่ได้เป็นลักษณะที่จะสามารถให้บริการได้ด้วยตัวของระบบเอง แต่ตัวมันเองเป็นเพียงแค่ Bearer ให้กับ Application ต่าง ๆ ที่ต้องการใช้ความเร็วที่เพิ่มมากกว่าปกติในระบบ GSM ที่เคยรองรับอยู่เดิมมาก่อน และระบบ GPRS จะต้องต่อไปยัง Packet Data Network ที่เป็น IP Network อีกคือหนึ่ง

ดังนั้นผู้ให้บริการเครือข่ายโทรศัพท์เคลื่อนที่ ที่จะเปิดใช้ในระบบ GPRS ได้นั้นจะต้องทำการติดตั้งระบบเครือข่าย ที่ประกอบด้วยหน่วยหลัก ๆ 2 หน่วยด้วยกันคือ

1. SGSN (Serving GPRS Supports Node)
2. GGSN (Gateway GPRS Supports Node)

โดยทั้งสองหน่วยหลักขององค์ประกอบนี้จะถูกเชื่อมต่อเข้าด้วยกันโดยมีอุปกรณ์อื่น ๆ เป็นตัวช่วยเพื่อไปร่วมใช้ Radio Interface จาก Base Station โดยผ่านตัวควบคุม ที่เรียกว่า PCU (Packet Control Unit) ที่ติดตั้งไว้ที่ BSC(Base Station Controller) อันทั้งนี้อาจมองไป Network เป็นอีก Network หนึ่ง ซึ่งเข้ากับ Mobile Phoneผ่านทาง Radio Interface ของระบบ GSM Network เดิมโดยเป็นบริการที่เกี่ยวข้องกับการรับส่งข้อมูลเป็น Packet โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 การเชื่อมต่อของ GPRS

จากรูปที่ 2.1 จะให้เห็นว่าระบบ GPRS นั้นจะเป็นการติดตั้งอุปกรณ์เพิ่มลงบน network เดิมของระบบ GSM โดยส่วนที่เพิ่มขึ้นจะเป็นส่วนของ Packet switch และ Gateway GSN ที่เชื่อมต่อ เข้าสู่ระบบ Internet

2.2.4 ประโยชน์ของ GPRS

2.2.4.1 GPRS เป็นระบบที่ทำให้สามารถรับส่งข้อมูลความเร็วสูง (non voice) ด้วยความเร็วสูงสุดถึง 40 Kbps ทำให้การใช้บริการ internet การใช้ wap และ mms สามารถทำงานได้รวดเร็ว เป็นผลให้เกิดการพัฒนา เทคโนโลยีการบริการ non voice กันมากขึ้น เช่น การใช้บริการ wap เพื่อการไหลรูปภาพ เสียงเรียกเข้า การรับส่งข้อความแบบ mms และการใช้งานระบบอินเทอร์เน็ต ผ่านโทรศัพท์เคลื่อนที่กับคอมพิวเตอร์ โน้ตบุ๊ก และเครื่อง PDA เพื่อใช้งาน Web browsing และ E-mail

2.2.4.2 ระบบ GPRS เป็นระบบที่มีความสามารถในการใช้งานแบบ always on คือในการเชื่อมต่อ สามารถเชื่อมต่อเข้ากับ Gateway ของระบบเครือข่ายได้โดยอัตโนมัติ ไม่จำเป็นต้องทำการ โทรออกเหมือนกับระบบ CSD ทำให้การเชื่อมต่อสามารถทำได้อย่างรวดเร็ว

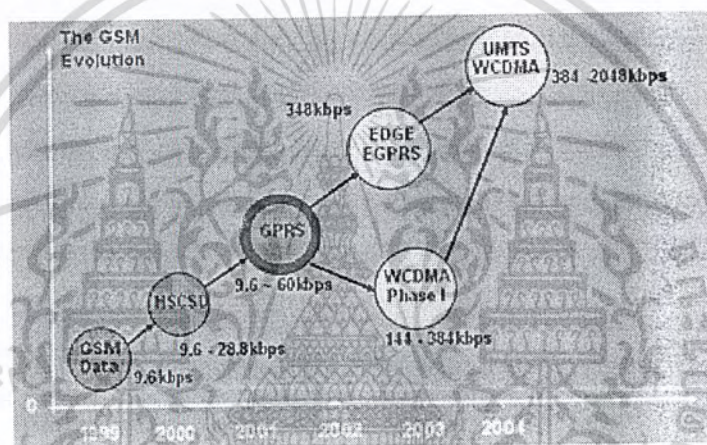
2.2.4.3 ไม่สูญเสียการติดต่อขณะใช้งาน ระบบ GPRS อยู่เราจะไม่สูญเสียการติดต่อกับผู้อื่น เนื่องจาก GPRS ไม่มีการ Dial up ดังนั้นขณะใช้งานอยู่สามารถมีผู้โทรเข้าหาเราและรับสายสนทนาได้ทันที และขณะเดียวกันเราก็สามารถโทรออกเพื่อสนทนาได้ทันทีเช่นเดียวกัน

2.2.4.4 ค่าใช้จ่ายในการใช้งานที่ถูกลง และมีความเป็นธรรม เพราะการคิดค่าใช้จ่าย จะคิดตามปริมาณของข้อมูล ที่เรารับและส่ง ผ่านเครือข่าย เรียกว่าสามารถเชื่อมต่อทั้งไว้ทั้งวันได้โดยไม่เสียเงินหากไม่มีการรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.5 ความเร็วของ GPRS

ในอดีตระบบ GSM นั้นสามารถทำการส่งข้อมูล data ผ่านระบบ CSD (Circuit Switch Data) ที่ระดับความเร็ว 9.6 Kbps ซึ่งสำหรับผู้ที่ใช้งานระบบอินเทอร์เน็ต จะทราบกันดีว่าความเร็วในระดับ 9.6 Kbps นี้ถือว่าช้ามากในการใช้งานอินเทอร์เน็ตจนทำให้การเข้าชมหน้าเว็บ ต่าง ๆ ใช้เวลานานมาก และค่าใช้จ่ายที่เกิดขึ้น ก็จะเป็นการคิดเป็นนาที ทำให้เสียค่าใช้จ่ายในการใช้งานที่สูง ดังนั้นระบบ GPRS จึงเกิดขึ้นมาเพื่อรองรับบริการ Non voice services ด้วยความสามารถในการรับส่งข้อมูลความเร็วสูง ในทางทฤษฎีสามารถ ทำความเร็วสูงสุดได้ถึง 171.2 Kbps แต่หมายความว่าระบบ GPRS นี้จะต้องใช้งานเป็นช่วงของเวลา (Timeslot) ของระบบ GSM ทั้งหมด 8 timeslot ซึ่งเป็นไปไม่ได้ในทางปฏิบัติ ในปัจจุบันความเร็วของระบบ GPRS ที่ใช้งานจะอยู่ที่ประมาณ 40 Kbps ซึ่งจะเร็วหรือช้ากว่านี้ก็จะขึ้นอยู่กับปริมาณของผู้ใช้งาน ในพื้นที่นั้นด้วย



รูปที่ 2.2 วิวัฒนาการของระบบ GSM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

จาวาสำหรับอุปกรณ์ขนาดเล็กและอุปกรณ์ไร้สาย

3.1 บทนำ

หลังจากที่บริษัทซัน ไมโครซิสเต็มส์ได้ออก จาวา 2 ไมโครเอ디션 (Java 2 Micro Edition : J2ME) ซึ่งมีความยืดหยุ่นสูง เพื่อใช้พัฒนาแอปพลิเคชันรุ่นใหม่สำหรับอุปกรณ์อิเล็กทรอนิกส์และอุปกรณ์เคลื่อนที่ขนาดเล็กซึ่งส่วนมากได้แก่ โทรศัพท์เคลื่อนที่และเพจเจอร์รับส่งข้อมูล

3.2 ภาพรวมของจาวา

ภาษาจาวาได้กลายเป็นแพลตฟอร์มการพัฒนาแอปพลิเคชันแบบเชิงวัตถุ (Object-oriented) ในอุปกรณ์และงานต่างๆ อย่างเต็มตัว นับตั้งแต่แอปพลิเคชันเซิร์ฟเวอร์ระดับองค์กร เครื่องคอมพิวเตอร์แบบตั้งโต๊ะทั่วไป จนถึงแอปพลิเคชันฝังตัวสำหรับอุปกรณ์ขนาดเล็กอีกด้วย

แพลตฟอร์มจาวา 2 ที่ใช้งานอยู่ในขณะนี้ มี 3 รุ่นด้วยกัน โดยที่แต่ละรุ่นนั้นก็จะเป็นนำไปใช้กับเฉพาะกลุ่มแอปพลิเคชันดังนี้

- จาวา 2 เอ็นเทอร์ไพรส์เอ디션 (Java 2 Enterprise Edition : J2EE) ใช้งานกับแอปพลิเคชันบนเครื่องเซิร์ฟเวอร์สำหรับองค์กรที่รองรับระบบงานใหญ่ๆ และ โคลเอนด์จำนวนมาก
- จาวา 2 สแตนดาร์ดเอ디션 (Java 2 Standard Edition : J2SE) ใช้งานกับแอปพลิเคชันบนเครื่องคอมพิวเตอร์แบบตั้งโต๊ะทั่วไป
- J2ME ใช้งานกับแอปพลิเคชันรุ่นใหม่ซึ่งเน้นกลุ่มผู้ใช้อุปกรณ์อิเล็กทรอนิกส์และอุปกรณ์ฝังตัว (Embedded)

3.3 J2ME คืออะไร

J2ME เป็นแพลตฟอร์มจาวาที่ออกแบบมาเพื่อใช้กับแอปพลิเคชันที่ทำงานบนอุปกรณ์ขนาดเล็ก เช่น โทรศัพท์มือถือ พีดีเอ โทรศัพท์พร้อมจอภาพที่ต่อกับอินเทอร์เน็ต โทรศัพท์ดิจิทัลขนาดเล็ก อุปกรณ์บันเทิงและระบบนำทางในรถยนต์ สวิตช์ในระบบเครือข่าย คอมพิวเตอร์ของเครื่องอำนวยความสะดวกภายในบ้าน ฯลฯ โดย J2ME ได้นำโครงสร้างแบบโมดูลที่มีความยืดหยุ่นสูงเข้ามาใช้ เพื่อให้สามารถสนับสนุนการทำงานอุปกรณ์หลากหลายประเภท J2ME กำหนดชั้นของซอฟต์แวร์ (Software) ไว้ 3 เลเยอร์ (Layer) ด้วยกัน โดยเลเยอร์ทั้งหมดจะอยู่บนเนื้อชั้นระบบปฏิบัติการของอุปกรณ์ ดังนี้

3.3.1 เลเยอร์จาวาเวอร์ชวลแมชชีน (Java Virtual Machine) เป็นเลเยอร์ของจาวาเวอร์ชวลแมชชีนปรับแต่งให้เข้ากับระบบปฏิบัติการของอุปกรณ์ และรองรับแต่ละ คอนฟิกูเรชัน (Configuration) ของ J2ME และเวอร์ชวลแมชชีน ของ J2ME ได้แก่ (Connected Virtual Machine : CVM) และ (Kilobyte Virtual Machine : KVM) ทั้ง CVM และ KVM ต่างสนับสนุนยูทิลิตี้

JavaCodeCompact หรือคลาส prelinker preloader และ ROmnizer ยูทิลิตี้นี้จะโยง จาวาคลาสเข้าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับเวอร์ชวลแมชชีน ช่วยลดระยะเวลาเริ่มใช้งาน VM ได้มากขึ้น ตัวอย่างเช่น เรียก KVM และ CLDC ขึ้นมาก่อนโหลดไปยังหน่วยความจำแบบ ROM ของอุปกรณ์ คลาส CLDC จะถูกโยกเข้าไปใน KVM โดยตรง เพื่อปรับปรุงประสิทธิภาพการทำงานเราเรียกขั้นตอนนี้ว่าการโหลดเข้าหน่วยความจำแบบ ROM (ROMized)

3.3.2 เลขอร์คอนฟิเจอร์ชัน เป็นเลขอร์ของคอนฟิเจอร์ชันของ J2ME ซึ่งกำหนดคลาสไลบรารีอุปกรณ์ ใช้งานทั่วไปหรือกลุ่มอุปกรณ์ที่มีความต้องการหน่วยความจำและหน่วยประมวลผลใกล้เคียงกันคอนฟิเจอร์ชันในจาวาสำหรับอุปกรณ์ขนาดเล็ก มี 2 ประเภทด้วยกัน คือ CDC (Connected Device Configuration) และ CLDC (Connected Limited Device Configuration)

3.3.3 เลขอร์โพรไฟล์ (Profile) เป็นเลขอร์ที่สร้างเหนือเลขอร์คอนฟิเจอร์ชัน โดยกำหนดคลาสไลบรารีเพื่อสนองต่อความต้องการขอลาดเฉพาะกลุ่ม ตัวอย่างของโพรไฟล์ในเลขอร์นี้ได้แก่ PDA Profile , MID Profile , Foundation Profile และ Personal Profile

3.4 คอนฟิเจอร์ชันใน J2ME

คอนฟิเจอร์ชันและโพรไฟล์เป็นองค์ประกอบหลักของ J2ME โดยมีจุดสำคัญ คือ เพื่อปรับแต่งเวอร์ชวลแมชชีน และคลาสไลบรารีให้เหมาะสมกับอุปกรณ์แต่ละประเภท คอนฟิเจอร์ชัน คือ ชุดของคุณสมบัติขั้นต่ำของ จาวาเวอร์ชวลแมชชีน และจาวาคลาสสำหรับอุปกรณ์แต่ละประเภท เป็นตัวแทนของอุปกรณ์ที่มีลักษณะคล้ายคลึงกัน อาจกล่าวได้ว่า คอนฟิเจอร์ชันเป็นตัวกำหนดคุณสมบัติหรือไลบรารีขั้นต่ำของแพลตฟอร์มจาวาที่นักพัฒนาคาดว่าจะต้องมีในทุกอุปกรณ์ ขณะที่คลาสไลบรารีที่กำหนดในคอนฟิเจอร์ชันจะมีในทุกอุปกรณ์ที่อยู่ในกลุ่มเดียวกัน

ปัจจุบันคอนฟิเจอร์ชันใน J2ME แบ่งเป็น 2 ประเภทคือ CDC (Connected Device Configuration) และ CLDC (Connected Limited Device Configuration) ดังนี้

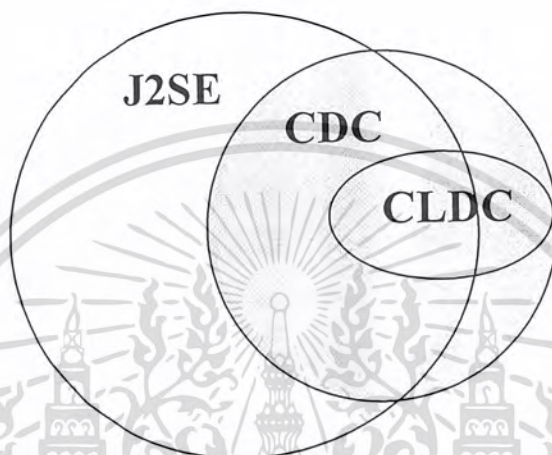
3.4.1 CDC เป็นอุปกรณ์ใช้ร่วมกับอุปกรณ์อื่น ติดตั้งตายตัว และใช้เชื่อมต่อข้อมูล โดยปรกติมักมีส่วนติดต่อกับผู้ใช้หลากหลายแบบ มีหน่วยความจำประมาณ 2-16 เมกกะไบต์ ใช้หน่วยประมวลผลแบบ 32 บิตหรือมากกว่า เชื่อมต่อเครือข่ายที่มีแบนด์วิดท์ (Bandwidth) สูงอย่างต่อเนื่องโดยอาศัยพอร์ต TCP/IP ตัวอย่างอุปกรณ์ประเภทนี้ ได้แก่ โทรศัพท์มือถือขนาดเล็ก อินเทอร์เน็ตทีวี โทรศัพท์พร้อมจอภาพที่เชื่อมต่อกับอินเทอร์เน็ต อุปกรณ์สื่อสารที่มีความซับซ้อนสูง อุปกรณ์บันทึกและระบบนำทางในรถยนต์

3.4.2 CLDC เป็นอุปกรณ์ส่วนบุคคล พกพาได้ และใช้เชื่อมต่อข้อมูล โดยปรกติมักมีส่วนติดต่อกับผู้ใช้แบบง่ายเมื่อเทียบกับระบบบนเครื่องคอมพิวเตอร์ตั้งโต๊ะ มีหน่วยความจำประมาณ 128 กิโลไบต์ – 1 เมกกะไบต์ ใช้หน่วยประมวลผลแบบ 16 หรือ 32 บิต เชื่อมต่อกับเครือข่ายที่มีแบนด์วิดท์ต่ำเป็น ระยะเวลานั้นๆ โดยไม่อาศัยพอร์ต TCP/IP ตัวอย่างอุปกรณ์ประเภทนี้ ได้แก่ โทรศัพท์มือถือแบบไม่ซับซ้อนมากนัก เพจเจอร์รับส่งข้อความ เครื่องปาล์มโอ

เอกสารแนบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในชั้นของคอนฟิเจอร์ชัน มีคลาส 2 ประเภทด้วยกัน คือ คลาสที่นำมาจาก J2SE และที่ออกแบบเฉพาะอุปกรณ์ขนาดเล็ก คลาสที่นำมาจาก J2SE จะมีคุณสมบัติอย่างเดียวกันกับคลาสใน J2SE หรือเป็นชั้นคลาสของ J2SE เช่น แพคเกจ `java.io` และ `java.util` จากรูป 2.1 แสดงความสัมพันธ์ระหว่าง J2SE และคลาสไลบรารีของ CDC และ CLDC จากภาพจะเห็นว่าคลาสใน CLDC ส่วนใหญ่ทำงานร่วมกับคลาสใน CDC ได้ดีเช่นเดียวกับกรณีของคลาสใน CLDC และคลาสใน J2SE



รูปที่ 3.1 ความสัมพันธ์ระหว่าง J2SE และ คลาสไลบรารีใน CDC และ CLDC

คลาสที่ไม่ได้นำมาจาก J2SE แลและออกแบบเพื่อใช้เฉพาะอุปกรณ์มักทำงานร่วมกับ J2SE ได้ไม่ค่อยดีใน CLDC คลาสเหล่านี้จะอยู่ในกลุ่มกรอบการติดต่อสื่อสารทั่วไป (Generic Connection Framework) โดยระบุไว้ในแพคเกจ `javax.microedition.io`

คอนฟิเจอร์ชันยังระบุคุณสมบัติของจาวาเวอร์ชวลแมชชีน ในเลเยอร์ด้านล่างอีกด้วย ในโครงสร้างปัจจุบัน CDC และ CLDC มีเวอร์ชวลแมชชีนที่ปรับแต่งมาเฉพาะตัวอยู่แล้ว เวอร์ชวลแมชชีนของ CDC คือ C Virtual Machine (CVM) มีคุณสมบัติครบถ้วนเหมือน Java 2 Virtual Machine แต่ขนาดเล็กกว่าออกแบบมาสำหรับอุปกรณ์ที่สลับซับซ้อน CVM มีความต้องการหน่วยความจำ 256 กิโลไบต์ ขณะที่หน่วยความจำแบบ ROM ของ CDC มีขนาด 1 เม็กกะไบต์ เวอร์ชวลแมชชีนของ CLDC คือ K Virtual Machine (KVM) แม้จะมีขนาดเล็กแต่มีความสามารถในการทำงานสูง ออกแบบมาเพื่อใช้งานกับอุปกรณ์ที่มีทรัพยากรจำกัด K ใน KVM หมายถึง กิโล โดยเรียกตามหน่วยความจำที่นับเป็นกิโลไบต์ ส่วนคอมพิวเตอร์แบบตั้งโต๊ะนั้นหน่วยนับเป็นเม็กกะไบต์ KVM เหมาะจะนำมาใช้กับหน่วยประมวลผล RISC/CISC แบบ 16/32 บิต ซึ่งมีหน่วยความจำทั้งหมดไม่กี่ร้อยกิโลไบต์เท่านั้นประมาณ 128 กิโลไบต์ ปัจจุบัน KVM มีความต้องการหน่วยความจำอยู่ในช่วง 40 – 80 กิโลไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

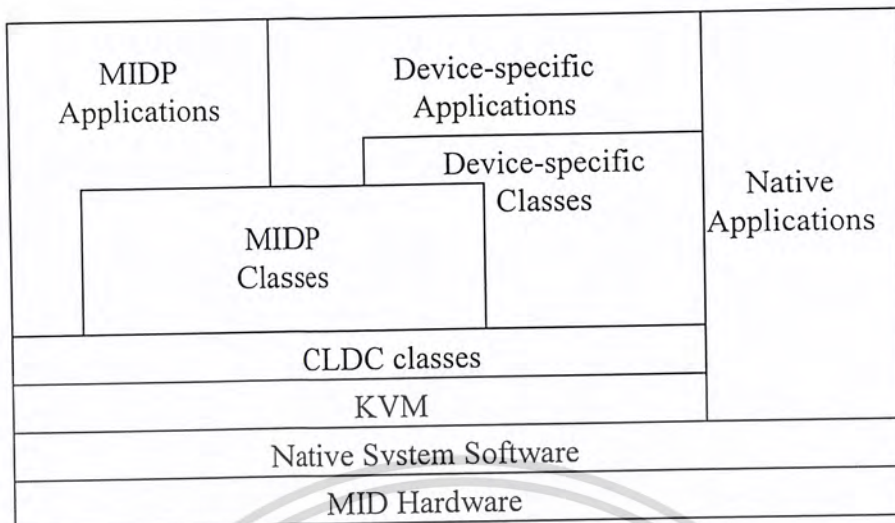
3.5 โพรไฟล์ของ J2ME

โพรไฟล์จะกำหนดชุดของ API ที่ต้องใช้เพิ่มเติม ตลอดจนกำหนดคุณสมบัติที่เป็นที่ต้องการของตลาดเฉพาะกลุ่มหรืออุปกรณ์เฉพาะประเภท คลาสไลบรารีในโพรไฟล์ช่วยให้นักพัฒนาสร้างฟังก์ชันเฉพาะอุปกรณ์ เช่น ส่วนติดต่อกราฟิกกับผู้ใช้การเชื่อมต่อเครือข่าย หน่วยเก็บข้อมูล Persistent Storage ฯลฯ ตามปรกติแล้วจะไม่สามารถนำคลาสไลบรารีที่สร้างเพื่อใช้งานกับโพรไฟล์หนึ่งไปใช้กับโพรไฟล์อื่นได้ ขณะนี้มีบางโพรไฟล์ได้ถูกกำหนดแล้ว แต่ก็มีหลายโพรไฟล์ที่ยังอยู่ในขั้นตอนการดำเนินการอยู่ โพรไฟล์ Foundation และ RMI ซึ่งสร้างบน CDC กำลังเผยแพร่อยู่ ส่วนที่สร้างบน CLDC มีโพรไฟล์เดียวคือ MID Profile (Mobile Information Device Profile : MIDP) มันจะเตรียมส่วนติดต่อผู้ใช้และหน่วยเก็บข้อมูล Persistent Storage ความสามารถด้านเครือข่าย แบบจำลอง API สำหรับแอปพลิเคชัน ไว้ให้อุปกรณ์ไร้สาย เช่น โทรศัพท์ที่ไม่ซับซ้อนมากนักและเพจเจอร์รับส่งข้อความ ส่วนโพรไฟล์ พีดีเอ (PDAP) กำลังได้รับการวิจัยอยู่

อุปกรณ์หนึ่งๆอาจมีโพรไฟล์ใช้งานมากกว่า 1 ชนิด และบางโพรไฟล์ใช้งานเฉพาะบางอุปกรณ์หรือแอปพลิเคชันเท่านั้น ตัวอย่างเช่น โพรไฟล์บน CDC ส่วนใหญ่ เช่น RMI และ Personal จะสร้างไว้เหนือ Profile Foundation หากไม่มี Profile Foundation และ CDC รองรับแอปพลิเคชันที่เขียนก็ไม่สามารถทำงานได้

3.6 J2ME สำหรับอุปกรณ์ไร้สาย

J2ME ได้ให้กำเนิดแอปพลิเคชันยุคใหม่บนอุปกรณ์ไร้สาย ช่วยให้เกมส์แบบหลายผู้เล่นที่ทำงานผ่านอินเทอร์เน็ตการทำธุรกรรมทางโทรศัพท์มือถือ แอปพลิเคชันสำหรับองค์กรทั้งไคลเอนต์และเซิร์ฟเวอร์ เกิดขึ้นได้บนโทรศัพท์มือถือและเพจเจอร์รับส่งข้อความ MIDP CLDC และ KVM ได้ กลายมาเป็นรากฐานในการพัฒนาจาวาแอปพลิเคชันบนอุปกรณ์ไร้สายยุคใหม่ มี 3 ปีจจัยในการสร้างแพลตฟอร์มสำหรับ แอปพลิเคชันไร้สาย คือ MIDP CLDC KVM



รูปที่ 3.2 โครงสร้างของ MIDP

โครงสร้างของเลเยอร์ต่างๆจากล่างขึ้นบน ดังนี้

1. เลเยอร์ฮาร์ดแวร์ MID หมายถึง ตัวโทรศัพท์มือถือ
2. เลเยอร์ซอฟต์แวร์ของระบบที่ติดตั้งมากับอุปกรณ์ หมายถึง ระบบปฏิบัติการและไลบรารีของระบบที่บริษัทผู้ผลิตให้มา
3. เลเยอร์ KVM เป็นส่วนที่เตรียม Runtime Environment ไว้ให้แอปพลิเคชันบนอุปกรณ์ไร้สาย
4. เลเยอร์ CLDC เป็นส่วนที่เตรียม API หลักของจาวาให้แอปพลิเคชันบนอุปกรณ์ไร้สาย
5. เลเยอร์ MIDP เป็นส่วนที่เตรียมไลบรารีสำหรับส่วนติดต่อกราฟิกกับผู้ใช้ และหน่วยเก็บข้อมูล Persistent Storage ระบบเครือข่าย และ ไทม์เมอร์

นอกจากคลาสไลบรารีสำหรับ MIDP ผู้ผลิตอาจเตรียมคลาสไลบรารีเฉพาะอุปกรณ์ไว้ให้นักพัฒนา เพื่อดึงความสามารถของฟังก์ชันที่มีอยู่แล้วไปใช้งานให้เกิดประโยชน์สูงสุด อาทิ การโทรศัพท์ การแชร์ข้อมูลกับแอปพลิเคชันที่ติดตั้งมาในเครื่อง จำพวก ปฏิทิน สมุดจดที่อยู่ การตรวจสอบข้อมูลอุปกรณ์ เช่น อายุแบตเตอรี่ ความแรงของสัญญาณ ฯลฯ แม้ว่าการนำคลาสเฉพาะอุปกรณ์ที่ผู้ผลิตเตรียมไว้มาใช้งาน จะช่วยเพิ่มความสามารถแก่แอปพลิเคชันบนอุปกรณ์ไร้สาย แต่ไม่สามารถเคลื่อนย้ายไปสู่อุปกรณ์ อื่นที่ใช้ MIDP ได้ เนื่องจากคลาสที่นำมาใช้อยู่นอกเหนือขอบเขตของ MIDP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ความต้องการของระบบ

3.7.1 อุปกรณ์ไร้สายจะทำงานสนับสนุน J2ME ได้ดีเมื่อมีคุณสมบัติตรงตามข้อกำหนด หากต้องการให้ KVM ทำงานได้อย่างมีประสิทธิภาพ โดยใช้ไลบรารี CLDC จะต้องมีคุณสมบัติของระบบขั้นต่ำดังนี้

- มีหน่วยความจำ 160 – 512 กิโลไบต์ สำหรับสร้างแพลตฟอร์มจาวา
- มีหน่วยประมวลผลแบบ 16-32 บิตความเร็ว 25 เม็กกะเฮิร์ซ
- ใช้พลังงาน โดยมากมักทำงานโดยใช้แบตเตอรี่
- เชื่อมต่อกับเครือข่ายได้ในช่วงสั้นๆ อาศัยระบบไร้สายเป็นส่วนใหญ่ แบนด์วิดท์ค่อนข้างจำกัด ความเร็ว 9600 ไบต์ต่อวินาทีหรือน้อยกว่า
- มีหน่วยความจำชั่วคราวขนาด 32 กิโลไบต์ สำหรับเก็บจาวา รันไทม์และหน่วยความจำของอ็อบเจกต์

3.7.2 การใช้งาน MIDP ฮาร์ดแวร์จะต้องมีคุณสมบัติดังนี้

3.7.2.1 การแสดงผล

- หน้าจอขนาด 96*54
- ความลึกของสี 1 บิต
- ตัดส่วนของภาพ 1:1

3.7.2.2 การรับข้อมูลเข้า

- ใช้กลไกการป้อนข้อมูลอย่างไรอย่างหนึ่ง อาทิ แป้นพิมพ์ หรือจอสัมผัส

3.7.2.3 หน่วยความจำ

- หน่วยความจำถาวรขนาด 128 กิโลไบต์สำหรับเก็บคอมไพล์ของ MIDP
- หน่วยความจำถาวรขนาด 8 กิโลไบต์สำหรับเก็บข้อมูลที่แอปพลิเคชันสร้าง

ขึ้น

- หน่วยความจำชั่วคราวขนาด 32 กิโลไบต์ สำหรับเก็บจาวา รันไทม์ เช่น จา

วา ฮีป

3.7.2.4 เครือข่าย

- เครือข่ายรับส่งข้อมูลผ่านระบบไร้สาย เชื่อมต่อได้ในช่วงสั้นๆ และมีแบนด์

วิดท์จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

หลักการเขียนโปรแกรมบนอุปกรณ์ไร้สายด้วยจาวา

4.1 บทนำ

ในบทนี้จะแสดงให้เห็นภาพรวมของหลักการพัฒนาแอปพลิเคชันบน MIDP ด้วย J2ME โดยชี้ให้เห็นว่าใน MIDP มี จาวาคลาสอะไรบ้าง ตลอดจนแสดงถึงวิธีการเขียนโปรแกรม MIDP เพื่อใช้งานใน MIDP

สภาพแวดล้อมการทำงานของจาวาใน MIDP มักมาพร้อมกับ KVM ไลบารีของ CLDC และ MIDP และซอฟต์แวร์จัดการแอปพลิเคชัน (Application Management Software – AMS) ไลบารีจาวาใน CLDC และ MIDP จะเป็นรากฐานรองรับการเขียนแอปพลิเคชันบนอุปกรณ์ไร้สาย

4.2 ไลบารีของ CLDC

ไลบารีของ CLDC เป็นไลบารีของเครือข่ายและระบบชั้นสูงที่ไม่จำกัดเฉพาะอุปกรณ์ใด อุปกรณ์หนึ่งประกอบด้วย คลาส 2 ประเภท คือ คลาสที่เป็นซับเซตของ J2SE และคลาสดูเกี่ยวกับกรอบการติดต่อสื่อสารทั่วไป

4.2.1 ซับคลาสของ J2SE คลาสประเภทแรกของ CLDC เป็นซับคลาสของไลบารี J2SE กำหนดไว้ในแพ็คเกจ `java.lang.java.util` และ `java.io` โดยคลาสระบบและคลาสชนิดข้อมูลเข้ากันได้กับ J2SE และ J2EE และเพื่อให้เข้ากันได้เคลื่อนย้ายข้ามแพลตฟอร์มได้สะดวก คลาสประเภทนี้จะใช้ชื่อคลาสและชื่อแพ็คเกจตามชื่อคลาสอย่างเดียวกันใน J2SE หรือตามชื่อซับคลาสใน J2SE ที่เกี่ยวเนื่องกัน โดยไม่มีการเพิ่มเมธอด `public` หรือ `protected` หรือฟิลด์ซึ่งไม่มีในคลาสที่เกี่ยวข้องกัน ใน J2SE ความหมายของคลาสและเมธอดจึงไม่เปลี่ยนแปลง

4.2.2 คลาสที่มีเฉพาะใน CLDC คลาสประเภทที่สองของ CLDC กำหนดไว้ในแพ็คเกจ `javax.microedition.io` คลาสเหล่านี้มีเฉพาะใน CLDC จึงไม่สามารถทำงานร่วมกับไลบารีของ J2SE ได้ เป็นคลาสชั้นสูงเกี่ยวกับเรื่องทั่วไปของเครือข่าย กำหนดไว้ในแพ็คเกจ `java.io` และ `java.net` คลาสประเภทนี้จะเตรียมกรอบการติดต่อสื่อสารผ่านเครือข่ายให้กับอุปกรณ์ที่สนับสนุน J2ME โดยกรอบการสื่อสารนี้มักเรียกกันว่า Generic Connection Framework คลาสส่วนใหญ่ได้แก่อินเทอร์เฟซที่แทนการสื่อสารรูปแบบต่างๆ เช่น ซ็อกเก็ต ดาต้าแกรมซีเรียล และ http ขึ้นอยู่กับว่าผู้ผลิตอุปกรณ์หรือผู้ให้บริการเครือข่ายจะติดตั้งอินเทอร์เฟซการเชื่อมต่อเหล่านี้เพียงบางตัวหรือครบทั้งหมด เมื่อพิจารณาจากความสามารถของอุปกรณ์และเครือข่าย

ข้อจำกัดของ CLDC CLDC และ KVM ออกแบบมาเพื่อใช้กับอุปกรณ์ที่มีทรัพยากรจำกัด จึงไม่สามารถสนับสนุนคุณสมบัติและฟังก์ชันได้หลากหลายเท่า J2SE คุณสมบัติบางอย่างที่มีใน J2SE ถูกดึงออกจาก CLDC และ KVM เพื่อลดขนาดและปรับปรุงประสิทธิภาพในการทำงาน โดยจะต้องศึกษาข้อจำกัดของ J2ME ก่อนที่จะเริ่มออกแบบและพัฒนาแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 ประเภทของข้อมูลพื้นฐานของ CLDC สนับสนุนเฉพาะชนิดของชนิดข้อมูลที่นำมาจาก J2SE ซึ่งได้แก่ byte short int long char และ boolean แต่ไม่สนับสนุนข้อมูลประเภท float และ double เนื่องจากสาเหตุสองประการ คือ อุปกรณ์เป้าหมายที่นำ CLDC ไปใช้ ส่วนใหญ่ไม่มีฮาร์ดแวร์รองรับตัวเลขทศนิยม และหากจะใช้ซอฟต์แวร์เข้าช่วยก็จะต้องเสียค่าใช้จ่ายสูงมาก นอกจากนี้ CLDC ยังกำหนดคลาส type wrapper สำหรับข้อมูลทุกประการ ได้แก่

```
java.lang.Boolean
java.lang.Byte
java.lang.Character
java.lang.Integer
java.lang.Long
java.lang.Short
```

คลาส type wrapper กำหนดไว้ในจาวา เนื่องจากจาวามีระบบย่อยหลายระบบซึ่งทำงานได้เฉพาะกับอ็อบเจกต์เท่านั้นในกรณีนี้ คุณสามารถสร้างอ็อบเจกต์โดยใช้คลาส wrapper ซึ่งเก็บชนิดของข้อมูลที่นำมาจาก J2SE เอาไว้

4.3 ไลบรารีของ MIDP

ไลบรารีของ CLDC ช่วยสร้างฟังก์ชันที่ไม่จำกัดเฉพาะอุปกรณ์บางชนิด ไลบรารีของ MIDP กลับตรงกันข้ามได้แก่ การจัดการแอปพลิเคชันบนอุปกรณ์ ส่วนการติดต่อกราฟิกกับผู้ใช้ทั้งแบบพื้นฐานและซับซ้อน พื้นที่เก็บข้อมูลแบบคงตัว และความสามารถเพิ่มเติมด้านเครือข่าย

4.3.1 คลาสโปรแกรมจัดการแอปพลิเคชัน เป็นคลาสที่ติดต่อกับโปรแกรมจัดการแอปพลิเคชันบนอุปกรณ์ถูกกำหนดไว้ในแพ็คเกจ `java.microedition.midlet` แอปพลิเคชันทั้งหลายที่เขียนใน MIDP จะต้องขยายคลาส `MIDlet` ที่อยู่ในแพ็คเกจออกไป และจะต้องนำเมธอดทั้ง 3 ซึ่งได้แก่ `startApp()` `pauseApp()` และ `destroyApp()` เข้ามาใช้

4.3.2 คลาสของส่วนติดต่อกับกราฟิกกับผู้ใช้ (GUI Class) เนื่องจากชุดเครื่องมือ `Abstract Windowing Toolkit` ใน J2SE ออกแบบมาเพื่อใช้งานกับแอปพลิเคชันบนเครื่องคอมพิวเตอร์แบบตั้งโต๊ะ และไม่สามารถนำมาใช้กับอุปกรณ์ไร้สายได้ เนื่องจากบริบททรัพยากรประเภทหน่วยความจำค่อนข้างมาก MIDP มีวิธีที่ต่างกันในกาหนดไลบรารีสำหรับแพ็คเกจส่วนติดต่อกับกราฟิกกับผู้ใช้ ด้วยการใช้ API ชั้นสูงซึ่งเน้นความสามารถในการเคลื่อนย้ายข้ามอุปกรณ์ และ API ชั้นพื้นฐานซึ่งเน้นองค์ประกอบกราฟิกเฉพาะอุปกรณ์ และ input event ทั่วไป คลาสที่เชื่อมต่อกับส่วนติดต่อกับกราฟิกกับผู้ใช้ และ event-handling กำหนดไว้ในแพ็คเกจ `java.microedition.lcdui` `Screen` ถือเป็นซูเปอร์คลาสของคอมโพเนนต์ส่วนติดต่อกับผู้ใช้โดยใช้ API ชั้นสูง ประกอบไปด้วย `Alert` , `Form` , `List` , `Textbox` ฯลฯ `Canvas` และ `Graphic` เป็นคลาสหลักของ API ระดับต่ำ (Low Level) แอปพลิเคชันเกมส์เป็นแอปพลิเคชันที่ใช้

API ระดับต่ำในการสร้างส่วนติดต่อกับผู้ใช้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 คลาสของพื้นที่เก็บข้อมูลแบบคงตัว (Persistent Storage Class) ในบางครั้งแอปพลิเคชันที่เขียนบน MIDP จำเป็นต้องเก็บข้อมูลไว้บนอุปกรณ์อย่างถาวร คลาสที่กำหนดไว้ในแพ็คเกจ `java.microedition.rms` ให้กลไกการเก็บข้อมูลถาวรที่เรียกว่า `RecordStore` โดยยอมให้แอปพลิเคชันเขียนลบและปรับปรุงบันทึกข้อมูลในหน่วยเก็บข้อมูลถาวรในอุปกรณ์

4.3.4 คลาสของเครือข่าย แม้ว่ากรอบการติดต่อสื่อสารทั่วไปที่กำหนดไว้ใน CLDC จะประกอบไปด้วยชุดของอินเทอร์เฟซการเชื่อมต่อเครือข่าย แต่ก็ไม่มีโปรโตคอลอยู่เบื้องหลังอินเทอร์เฟซการเชื่อมต่อจริงๆ หากแต่ปล่อยไว้ให้เป็นหน้าที่ของ MIDP ในบรรดาอินเทอร์เฟซการเชื่อมต่อเครือข่ายเหล่านี้ `HttpConnection` ถือเป็นอินเทอร์เฟซหลักที่ต้องมีใน MIDP เสมอ คลาสของอินเทอร์เฟซเหล่านี้กำหนดในแพ็คเกจ `java.microedition.io`

4.4 MIDlet

MIDlet หมายถึงแอปพลิเคชันบน MIDP MIDlet มีส่วนคล้ายกับจาวาแอปเพล็ต แม้จะไม่มีเมธอด `main()` แต่ MIDlet ก็นำคลาส `java.microedition.midlet.MIDlet` ตลอดจนเมธอด `startApp()` `pauseApp()` และ `destroyApp()` เข้ามาใช้ นอกจากนี้ MIDlet ยังกำหนด constructor แบบ `public` ที่ไม่มีอาร์กิวเมนต์ใดๆ อีกด้วย เราสามารถให้คำนิยามของคลาส `javax.microedition.midlet.MIDlet` ได้ดังนี้

```
Public abstract คลาส MIDlet extends Object
```

```
{
```

```
protected MIDlet()
```

```
protected abstract void startApp() throws MIDletStateChangeException
```

```
protected abstract void pauseApp()
```

```
protected abstract void destroyApp(boolean unconditional)
```

```
throws MIDletStateChangeException
```

```
public final String getAppProperty(String key)
```

```
public final void notifyDestroyed()
```

```
public final void notifyPaused()
```

```
public final String getAppProperty(String key)
```

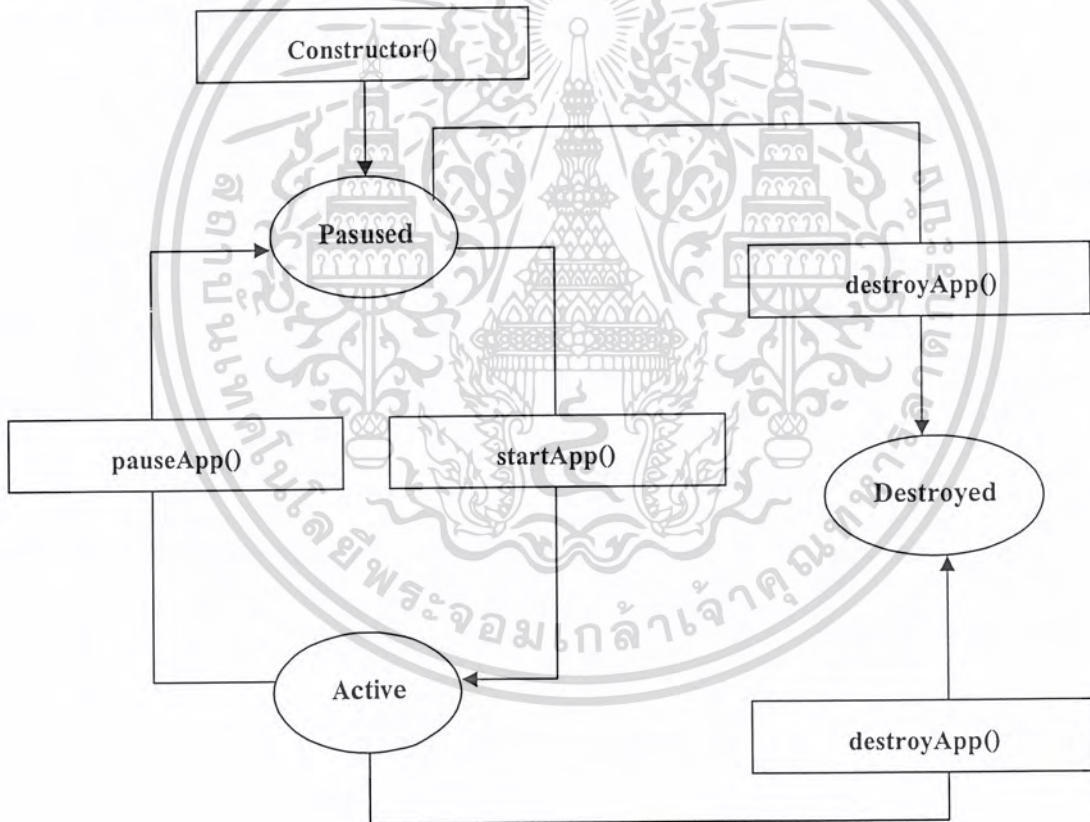
```
public final void resumeRequest()
```

```
}
```

คลาส MIDlet ระบุเมธอดที่สามารถเรียกใช้โดยซอฟต์แวร์จัดการแอปพลิเคชัน (AMS) เพื่อสั่งให้แอปพลิเคชัน MIDlet เริ่มต้นและหยุดทำงาน

4.4.1 วงจรการทำงานของ MIDlet

การกระทำของ MIDlet ประกอบไปด้วย 3 สถานะ คือกำลังทำงาน หยุดชั่วคราวและถูกทำลาย อาศัยซอฟต์แวร์จัดการแอปพลิเคชัน เป็นตัวควบคุมการเปลี่ยนสถานะหนึ่งไปยังอีกสถานะหนึ่งควบคุมด้วยเมธอด `startApp()` `pauseApp()` และ `destroyApp()` ที่มาพร้อมกับ MIDlet จากรูปที่ 4.1 แสดงให้เห็นจุดเปลี่ยนระหว่างสถานะทั้ง 3 โดยการเรียกใช้เมธอดข้างต้น เมื่อ MIDlet พร้อมสั่งกระทำการ ซอฟต์แวร์จัดการแอปพลิเคชันจะสร้างตัวอย่าง MIDlet ขึ้นมาก่อนโดยใช้ constructor แบบ public ที่ไม่มีอาร์กิวเมนต์ใดๆ โดย MIDlet จะอยู่ในสถานะหยุดชั่วคราว จากนั้นซอฟต์แวร์จัดการแอปพลิเคชันจะเรียกเมธอด `startApp()` ขึ้นมา และ MIDlet จะเข้าสู่สถานะกำลังทำงานเปิดรับทรัพยากรที่ต้องการและเริ่มดำเนินการทำงาน ในสถานะนี้ MIDlet จะทำงานและดึงทรัพยากรที่ต้องการไว้ใช้งาน เมื่อซอฟต์แวร์จัดการแอปพลิเคชันไม่ต้องการให้ MIDlet ทำงานต่อไป ก็จะเรียกเมธอด `pauseApp()` จากนั้น MIDlet จะหยุดทำงานและเข้าสู่สถานะหยุดชั่วคราว คืนทรัพยากรที่ดึงมาใช้งานและเข้าสู่ภาวะไม่ทำงาน MIDlet สามารถกลับไปอยู่ที่สถานะกำลังทำงาน ได้เมื่อซอฟต์แวร์จัดการแอปพลิเคชันเรียกเมธอด `startApp()` ขึ้นมา



รูปที่ 4.1 วงจรการทำงานของ MIDlet

ท้ายที่สุดเมื่อซอฟต์แวร์จัดการแอปพลิเคชันไม่ต้องการเรียกใช้งาน MIDlet หรือต้องการเคลียร์หน่วยความจำเพื่อให้โปรแกรมอื่นได้ใช้งาน ก็จะส่งสัญญาณเตรียมทำลาย MIDlet ึ่งด้วยการเรียกเมธอด `destroyApp()` และเข้าสู่สถานะถูกทำลาย ซึ่ง MIDlet จะปล่อยทรัพยากรทั้งหมด ทำการจัดเก็บข้อมูลถาวรต่างๆ และหยุดการทำงานทั้งหมด หาก MIDlet อยู่ระหว่างสั่งกระทำขั้นตอนสำคัญอยู่ ก็อาจร้องขอเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่เข้าสู่สถานะถูกทำลายได้โดยเรียกใช้ `MIDletStateChangeException` ใดๆก็ตามที่ซอฟต์แวร์จัดการแอปพลิเคชันอาจปฏิเสธหรือยินยอมตามคำร้องขอก็ได้ ตัวแปรบูลีน `unconditional` ในลายเซ็นของเมธอด `destroyApp()` จะเป็นตัวกำกับว่าการร้องขอนี้สมควรหรือไม่ ถ้าตัวบ่งชี้มีค่าเท่ากับ `true` การร้องขอเป็นอันตกไป ตรงกันข้ามหากมีค่าเท่ากับ `false` ก็จะได้รับคำตอบรับและจะเรียกเมธอด `destroyApp()` ขึ้นมาใหม่ในคราวต่อไป หาก `MIDlet` ต้องการเข้าสู่สถานะหยุดชั่วคราวหรือถูกทำลายด้วยตัวเอง ก็สามารถทำได้โดยเรียกเมธอด `notifyPause()` หรือ `notifyDestroy()` ตามลำดับ โดยเมธอดทั้ง 2 จะแจ้งให้ซอฟต์แวร์จัดการแอปพลิเคชันทราบว่า `MIDlet` ได้เข้าสู่ สถานะหยุดชั่วคราว/ถูกทำลายแล้ว ในกรณีนี้ซอฟต์แวร์จัดการแอปพลิเคชันจะไม่เรียกเมธอด `pauseApp()` หรือ `destroyApp()` อีก `MIDlet` จะเข้าสู่สถานะหยุดทำงานได้ก็ต่อเมื่อผ่านสถานการณ์ทำงานมาก่อนแล้ว ในทางตรงกันข้าม หากต้องการเข้าสู่สถานะถูกทำลาย สามารถเข้าได้โดยตรงจากสถานะหยุดทำงานชั่วคราวและสถานะกำลังทำงานนอกจากนี้ยังสามารถเข้าสู่สถานะหยุดการทำงานได้ในขณะที่กำลังทำงาน หรือเมื่อได้รับคำสั่งจากซอฟต์แวร์จัดการแอปพลิเคชัน

4.5 ซอฟต์แวร์จัดการแอปพลิเคชัน (Application Management Software)

ซอฟต์แวร์แอปพลิเคชันที่มาพร้อมกับ MIDP ทำหน้าที่ควบคุมการติดตั้ง สั่งกระทำการ และลบ `MIDlet` บางครั้งเราเรียกซอฟต์แวร์จัดการแอปพลิเคชันว่าเป็น ซอฟต์แวร์จัดการ `MIDlet` (`MIDlet Management Software`) หรือ โปรแกรมบริหารจาวาแอปพลิเคชัน (`Java Application Manager`)

การติดตั้งซอฟต์แวร์จัดการแอปพลิเคชันแตกต่างกันออกไปตามแต่ประเภทของอุปกรณ์ แต่ว่าให้บริดการขั้นพื้นฐานเหมือนกัน ได้แก่

- ยอมให้ผู้ใช้งานติดตั้งและถอดถอน `MIDlet` จากอุปกรณ์ไร้สาย ทั้งผ่านสายเคเบิลที่เชื่อมต่อเข้ากับเครื่องคอมพิวเตอร์ส่วนบุคคล หรือ จากระยะไกลโดยผ่านอินเทอร์เน็ต
- เตรียมสภาพแวดล้อม สั่งกระทำการให้แก่ `MIDlet` หลังจาก `MIDlet` เริ่มทำงานซอฟต์แวร์จัดการแอปพลิเคชันจะจัดเตรียมทรัพยากรของระบบ เช่น คลาสของ `CLDC` และ `MIDlet` และ `KVM` ไว้ให้ `MIDlet` ใช้งาน นอกจากนี้ยังเตรียมคลาสไฟล์ทรัพยากรและไฟล์ `descriptor` ของ แอพลิเคชันไว้ให้ `MIDlet` ได้ใช้เป็นรันไทม์ในการทำงานอีกด้วย
- ซอฟต์แวร์จัดการแอพลิเคชันยังจัดการแก้ไขข้อผิดพลาดทั้งหมดที่เกิดขึ้นระหว่างการติดตั้งและสั่งกระทำการแอปพลิเคชัน โดยไม่ทำให้ระบบล่มอีกด้วย

4.6 ข้อจำกัดต่าง

4.6.1 ข้อจำกัดในการจัดการข้อผิดพลาด `CLDC` นั้นรองรับความผิดพลาดในการทำงานในวงจำกัด โดยกำหนดคลาสแสดงข้อผิดพลาดไว้เพียง 2 คลาส คือ `java.lang.VirtualMachineError` และ `java.lang.OutOfMemoryError` ขณะที่คลาสดังกล่าวส่วนใหญ่ออกไปเนื่องจาก

4.6.1.1 ในระบบอุปกรณ์ฝังตัว การคืนสู่ภาวะทำงานผิดพลาด เป็นความสามารถเฉพาะบางอุปกรณ์ เท่านั้น ไม่ควรคาดหวังให้ผู้เขียนแอปพลิเคชันจัดการกับข้อบกพร่องเฉพาะอุปกรณ์เหล่านี้

4.6.1.2 ความผิดปกติประเภท Error มักไม่สามารถแก้ไขได้ การบรรลุความสามารถในการจัดการข้อผิดพลาดอย่างเต็มรูปแบบทำให้ต้องเสียค่าใช้จ่ายมาก และยังคงกำหนดลำดับความสำคัญไว้สูงทั้งที่อุปกรณ์ CLDC มีทรัพยากรค่อนข้างจำกัด

4.6.2 ข้อจำกัดของ KVM เนื่องจาก KVM เป็นเลเยอร์ Java Virtual Machine ที่รองรับไลบรารีของ CLDC คุณสมบัติใดที่ CLDC ไม่สนับสนุน ก็จะถูกดึงออกจาก KVM ด้วยเช่นกัน อาทิ ตัวเลขทศนิยม finalization และการจัดการข้อผิดพลาดเฉพาะอุปกรณ์

4.6.2.1 ไม่สนับสนุน Java Native Interface (JNI) ด้วยสาเหตุ 2 ประการคือในแบบจำลองความปลอดภัยของ CLDC ผู้เขียนแอปพลิเคชันไม่สามารถดาวน์โหลดไลบรารีใหม่ๆที่ยังมีฟังก์ชันเดิมของอุปกรณ์อยู่ และไม่สามารถเข้าถึงฟังก์ชันเดิมของอุปกรณ์ที่ไม่มีในไลบรารีของจาวาได้นอกจากนี้ การนำJNI มาใช้ยังเพิ่มภาระให้กับหน่วยความจำบนอุปกรณ์ CLDC

4.6.2.2 ไม่สนับสนุนโปรแกรมบรรจุคลาสที่ผู้ใช้งานกำหนดขึ้นเอง และโปรแกรมบรรจุคลาสใน KVM ก็ไม่อนุญาตให้ผู้ใช้งานลบล้าง เขียนทับ หรือกำหนดค่าใหม่ด้วย

4.6.2.3 ไม่สนับสนุน Reflection, RMI และ การเรียกลำดับอ็อบเจกต์ ทำให้โปรแกรม CLDC ไม่สามารถตรวจสอบเนื้อหาในคลาสอ็อบเจกต์ เมธอด ฯลฯ

4.6.2.4 ไม่สนับสนุน Thread Group หรือ Daemon คือ KVM สนับสนุนแอปพลิเคชันแบบทำงานคู่ขนาน (Multi-Thread) แต่ไม่รองรับกลุ่มงานย่อย (Thread Object) หรืองานเบื้องหลัง (Daemon Thread)

4.7 ระบบความปลอดภัยใน J2ME

วัตถุประสงค์สำคัญของการรักษาความปลอดภัยบนระบบคอมพิวเตอร์คือ การป้องกันทรัพยากรของระบบจากการเจาะเข้ามาทั้งที่ไม่ประสงค์ดีและโดยไม่ได้ตั้งใจ การดาวน์โหลดเนื้อหาและแอปพลิเคชันตามข้อกำหนดของ CLDC ระบุไว้ ทำให้การรักษาความปลอดภัยบนระบบเครือข่ายทวีความสำคัญมากขึ้น จาวานั้นออกแบบมาโดยคำนึงถึงความปลอดภัยมาตั้งแต่ต้น โดย J2SE เองมีคุณสมบัติด้านความปลอดภัยที่หลากหลาย เช่น การตรวจไบต์โค้ด และโปรแกรมจัดการความปลอดภัย อย่างไรก็ตาม เนื่องจากรหัสที่ J2SE ใช้ในการรักษาความปลอดภัยมีขนาดใหญ่เกินกว่าหน่วยความจำของอุปกรณ์ไร้สายที่มีทรัพยากรจำกัด จึงต้องลดทอนคุณสมบัติดังกล่าวลงให้เหมาะสมกับความต้องการระบบของอุปกรณ์ไร้สาย

4.8 การเวอร์ิฟายและฟรีเวอร์ิฟายไฟล์คลาส

เนื่องจากทรัพยากรที่มีจำกัดบนอุปกรณ์ไร้สายการเวอร์ิฟายคลาสของ KVM จึงต้องแยกกัน

กระทำ โดยบางส่วนกระทำนอกตัวอุปกรณ์นั้น เรียกว่า ฟรีเวอร์ิฟิเคชัน เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

คอมโพเนนต์หลักของส่วนติดต่อกับผู้ใช้ในภาษาจาวา

5.1 บทนำ

J2ME สามารถทำงานบนอุปกรณ์ที่หลากหลาย ตั้งแต่สมาร์ตการ์ดและโทรศัพท์เคลื่อนที่ ไปจนถึงอุปกรณ์ set-top-box สำหรับโทรทัศน์และเครื่องใช้ไฟฟ้าในบ้าน อุปกรณ์เหล่านี้มีความสามารถในการแสดงผลและทรัพยากรที่แตกต่างกันค่อนข้างมาก บริษัทชน ไมโครซิสเต็มส์ตระหนักถึงปัญหานี้จึงไม่ได้กำหนดคลาสส่วนติดต่อกับผู้ใช้ซึ่งสมบูรณ์แบบใน J2ME หากแต่กำหนด API ของส่วนติดต่อกับผู้ใช้ในลักษณะโพรไฟล์ของอุปกรณ์เคลื่อนที่ (Mobile Information Device Profile : MIDP) หรือถ้าเป็น API ส่วนติดต่อกับผู้ใช้สำหรับพีดีเอ ก็จะระบุไว้ในโพรไฟล์ของพีดีเอ (Personal Digital Assistant Profile:PDAP) เป็นต้น

API ของส่วนติดต่อกับผู้ใช้ใน J2SE กำหนดไว้ในชุดเครื่องมือ Abstract Window Toolkit(AWT) ซึ่งออกแบบสำหรับเครื่องคอมพิวเตอร์ตั้งโต๊ะที่มีอุปกรณ์การชี้ (Pointer Device) จอภาพขนาดใหญ่พร้อมหน่วยความจำที่พอเพียง ขณะที่หน้าจอแสดงผลไร้สายมีขนาดเล็กกว่า และใช้เป็นกดเพื่อป้อนข้อมูลเข้าแทนที่จะเป็นตัวชี้ ดังนั้นกลุ่มผู้เชี่ยวชาญการออกแบบ MIDP จึงได้กำหนด API ของส่วนติดต่อกับผู้ใช้ขึ้นมาใหม่ทั้งหมดโดยไม่ได้อ้างอิงซบเซตของ AWT เลขซึ่ง API ของส่วนติดต่อกับผู้ใช้เหล่านี้กำหนดไว้ในแพ็คเกจ javax.microedition.lcdui

API ของส่วนติดต่อกับผู้ใช้ใน MIDP มี 2 กลุ่มหลักๆ ด้วยกันคือ API ระดับสูงและระดับล่าง API ระดับสูงออกแบบให้เป็น Abstract เน้นความยืดหยุ่นในการโยกย้ายไปใช้งานในระบบอื่น แอปพลิเคชันที่ใช้ API ระดับสูงจะควบคุมวิธีแสดงผลได้น้อยมากและเข้าถึงได้เฉพาะอีเวนต์ระดับสูงเท่านั้น ในทางตรงกันข้ามการใช้ API ระดับล่างจะสามารถควบคุมการแสดงผลได้เต็มที่ สามารถเข้าถึงอุปกรณ์ป้อนข้อมูล และจัดการอีเวนต์พื้นฐานที่เกิดจากการตอบโต้ของผู้ใช้ได้ อย่างไรก็ตาม โปรแกรมที่ใช้ API ระดับล่างนี้จะขาดความยืดหยุ่น ขัดตักกับอุปกรณ์มากจนเกินไป โดยจะต้องพิจารณาถึงข้อดีข้อเสียในการออกแบบโปรแกรมด้วย

5.2 Displayable และ Display

ฟังก์ชันหลักอย่างหนึ่งของส่วนติดต่อกับผู้ใช้ ก็คือ การแสดงข้อมูลบนหน้าจอใน J2ME MIDP ออบเจกต์ Displayable จะเก็บข้อมูลที่ต้องการนำเสนอ และออบเจกต์ Display จะนำออบเจกต์ Displayable นั้น ไปแสดงบนจอให้ผู้ใช้เห็น

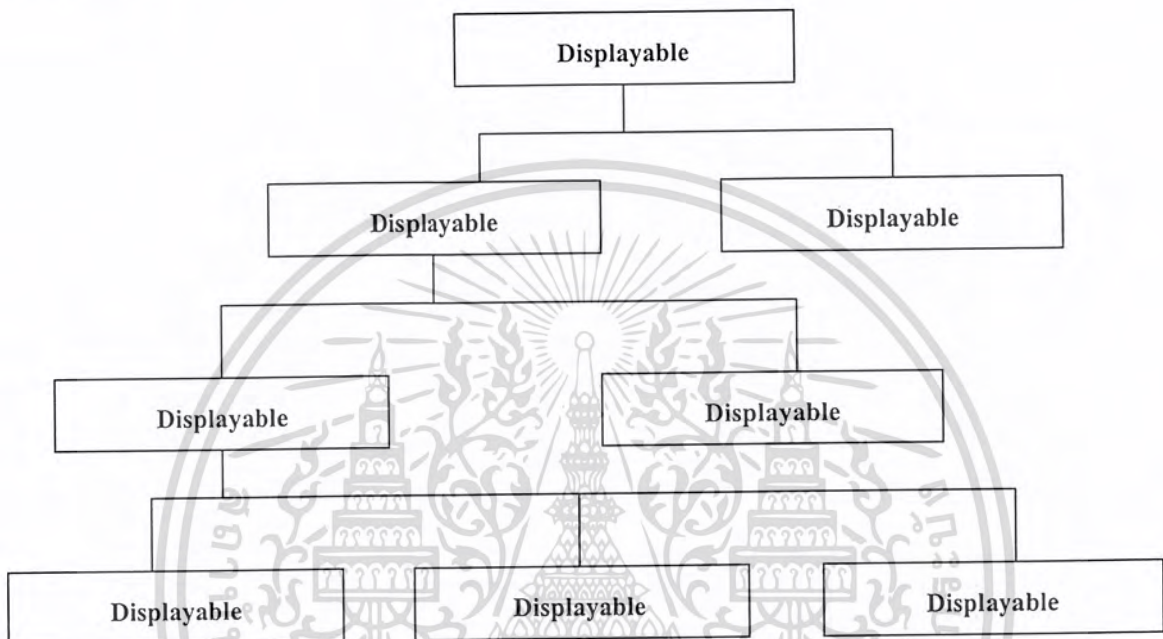
5.2.1 Displayable

เป็นคลาสหลักของส่วนติดต่อกับผู้ใช้ของ MIDP สามารถเก็บภาพกราฟิกที่สร้างเฉพาะอุปกรณ์ และนำไปแสดงบนจอ ตัว MIDP กำหนด API ของส่วนติดต่อกับผู้ใช้ออกเป็น 2 กลุ่มคือ API ระดับสูงและระดับล่าง คลาส Screen เป็นคลาสย่อยของ Displayable ที่ใช้ API ระดับสูง ส่วนคลาส Canvas เป็นคลาสย่อยของ Displayable ที่ใช้ API ระดับล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดและรูปร่างของหน้าจออุปกรณ์จะแตกต่างกันออกไป เนื่องจาก API ระดับสูงจะจัดการกับความแตกต่างของขนาดหน้าจอ ดังนั้น ในการเขียนแอปพลิเคชันจึงไม่ต้องพะวงเรื่องความแตกต่างของขนาดหน้าจอมากนัก แต่หากเป็นแอปพลิเคชันที่ใช้ API ระดับล่างจะต้องระวังเรื่องความแตกต่างของหน้าจอแสดงผลเหล่านี้

Displayable ทุกตัวต้องรีจิสเตอร์ Command และ CommandListener เพื่อที่จะสามารถรับอีเวนต์ของคอมมานด์ได้



รูปที่ 5.1 แสดงโครงสร้างลำดับชั้นของคลาส Displayable

5.2.1.1 Screen

เป็น subclass ของ Displayable ซึ่งนำ API ระดับสูงมาใช้ประกอบด้วยชุปเปอร์คลาส Alert, Form, List และ TextBox อาจมีชื่อหรือแถบตัวอักษรวิ่ง (ticker tape)

กำกับหรือจะไม่มีก็ได้

Screen เป็นคลาสแบบ Abstract เมธอดซึ่งกำหนดวิธีการแสดงผลและการโต้ตอบกับเนื้อหาในออบเจกต์ก็เป็นเมธอดแบบ Abstract ด้วย นำมาใช้โดย subclass ของ Screen มีเพียงเมธอดสำหรับปรับปรุงหน้าจอเวลาเมื่อต้องการแสดงผลเท่านั้นที่อยู่ในคลาส Screen อย่างไรก็ตาม ไม่สามารถเพิ่มเติมอะไรในคลาส Screen ได้เอง เนื่องจากใช้เมธอดแบบ Abstract ซึ่งไม่สามารถเข้าถึงได้จากภายนอกแพ็คเกจ javax.microedition.lcdui

5.2.1.2 Canvas

เป็น subclass ของ Displayable ซึ่งใช้ API ระดับล่างแอปพลิเคชันสามารถควบคุมได้ว่า จะแสดงผลอะไรและอย่างไรบ้าง และยังสามารถเข้าถึงอีเวนต์ระดับล่างเช่น การกดปุ่ม ได้ด้วย Canvas เป็นคลาส Abstract แอปพลิเคชันต้องสร้าง subclass ย่อยเพื่อใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2 Display

เป็นตัวจัดการการแสดงผล และใช้เมธอดเหล่านี้สำหรับอ่านค่าพารามิเตอร์ของอุปกรณ์และนำออบเจกต์ไปแสดงผลบนหน้าจอ

ในโปรแกรม MIDlet หนึ่งๆ จะมีออบเจกต์ของ Display ได้เพียงตัวเดียวเท่านั้น แอปพลิเคชันสามารถอ้างอิงออบเจกต์ Display ได้ด้วยเมธอด `getDisplay()` และสามารถเรียกเมธอดนี้ได้ทุกเมื่อตั้งแต่เริ่มเรียกใช้ `MIDlet.startApp()` ไปจนถึง `MIDlet.destroyApp()` ออบเจกต์ Display ทั้งหมดที่ได้จาก `getDisplay()` จะยังคงอยู่ตลอด จนกว่าจะเสร็จสิ้นโปรแกรม MIDlet

ในแอปพลิเคชันหนึ่งๆ จะมีออบเจกต์ `Displayable` ที่จะแสดงผลบนอุปกรณ์หรือมีการทำงานโต้ตอบกับผู้ใช้ได้เพียงออบเจกต์เดียวเท่านั้น โดยเรียก `Displayable` นี้ว่า `Displayable` ปัจจุบัน ผู้ใช้จะป้อนข้อมูลหรือโต้ตอบได้เฉพาะกับ `Displayable` ปัจจุบันเท่านั้น คลาส `Displayable` มีเมธอด `getCurrent()` สำหรับดึง `Displayable` ปัจจุบัน และมีเมธอดไว้สำหรับแสดงผลคือ `setCurrent(Displayable nextDisplayable)`

แอปพลิเคชัน สามารถควบคุม `Displayable` ปัจจุบัน และ เรียก เมธอด `Display.setCurrent()` ได้ตลอดเวลา เรดของ MIDlet สามารถกำหนดหรือดึงค่า `Displayable` ปัจจุบัน ด้วย เมธอด `Display.setCurrent()` และ `Display.getCurrent()`

แอปพลิเคชัน หน้าฉากและหลังฉาก แม้สามารถสั่งให้แอปพลิเคชันหลายตัวทำงานพร้อมกันบนอุปกรณ์หนึ่งๆ อุปกรณ์ก็จะแสดงผลได้เฉพาะ `Displayable` ปัจจุบันเท่านั้น ดังนั้นแอปพลิเคชันใดที่มี `Displayable` ปัจจุบันแสดงผลอยู่บนจอจะเรียกว่าเป็นแอปพลิเคชัน หน้าฉาก (foreground) ซึ่งสามารถรับ อีเวนต์การป้อนข้อมูลเข้าจากผู้ใช้ได้ แอปพลิเคชันที่เหลือซึ่งไม่ได้แสดงผลจะเรียกว่าแอปพลิเคชันหลังฉาก (background)

MIDlet สามารถวางตัวเองให้อยู่หลังฉากได้โดยใช้ `Display.setCurrent(null)` แต่การใช้เมธอดนี้ไม่ได้กำหนด `Displayable` ปัจจุบันให้เป็น null จริงๆ สมมติว่า `Displayable` ปัจจุบันคือ cD หลังจากแอปพลิเคชันเรียก `Display.setCurrent(null)` แล้ว การเรียกเมธอด `getCurrent()` ก็ ยังได้ค่าเป็น cD แอปพลิเคชัน จะกลับคืนสู่หน้าฉากโดยใช้ `Display.setCurrent(Display.getCurrent())`

แม้ว่า `Displayable` ปัจจุบันของแอปพลิเคชันจะไม่ได้ว่าอะไรบนจอก็ตามแอปพลิเคชันก็ยังรับรู้ ว่า `Displayable` ใดเป็น `Displayable` ปัจจุบัน ซึ่งสำคัญมากแม้แต่กับแอปพลิเคชันหลังฉาก เนื่องจากเมื่อแอปพลิเคชันหลังฉากกลายเป็นแอปพลิเคชัน หน้าฉาก ก็จะสามารถแสดงผล `Displayable` ที่ ถูกต้องได้

โปรแกรม MIDlet แต่ละตัวจะมี `Displayable` ปัจจุบันของตัวเอง เมธอด `getCurrent()` จะให้ `Displayable` ปัจจุบัน โดยใช้เมธอด `setCurrent()` ก็ไม่มีผลอะไรกับ `Displayable` ปัจจุบันของ MIDlet อื่นๆ

System Screen โดยปรกติแล้ว จะสามารถมองเห็นการแสดงผลบนหน้าจอปัจจุบันของ MIDlet ที่ทำงานหน้าฉากได้ อย่างไรก็ตามในบางสถานการณ์ระบบอาจจะสร้างหน้าจอชั่วคราวขึ้นมาบังหน้าจอปัจจุบันของแอปพลิเคชันเอาไว้ คือ หน้าจอของระบบ (system screen) โดยจะแสดงขึ้นมาเมื่อระบบต้องการแสดงเมนูเพิ่มเติม นอกเหนือไปจากการแก้ไขฟิลด์ข้อความต่างๆ ใน Form

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แม้ว่าหน้าจอของระบบ จะโผล่ขึ้นมาบังหน้าจอของแอปพลิเคชัน Displayable ปัจจุบันก็ยังเป็นอันเดิมไม่เปลี่ยนแปลง การเรียกใช้เมธอด `getCurrent()` ในขณะที่ยังมองเห็นหน้าจอของระบบ ก็ยังได้ค่า `Displayable` ปัจจุบันของแอปพลิเคชัน และไม่ใช้หน้าจอของระบบถ้าจะดูว่า `Displayable` ปัจจุบัน ถูกบังด้วยหน้าจอของระบบหรือไม่ สามารถตรวจสอบ ได้จาก `currentDisplayable.isShown()`

5.3 รูปภาพ (Image)

คลาส `Image` ใช้เก็บข้อมูลรูปภาพ ออบเจกต์ `Image` จะอยู่ในหน่วยความจำที่แยกจากส่วนแสดงผล ออบเจกต์ `Image` สามารถวางลงบน `Canvas` หรือวางใน `Form` , `Alert` , `List` หรือ `ChoiceGroup` ก็ได้ `Image` อาจเป็นรูปภาพที่แก้ไขได้ หรือแก้ไขไม่ได้ ขึ้นอยู่กับวิธีการสร้าง

5.3.1 รูปที่แก้ไขไม่ได้ มักมีที่มาจากภาพจากไฟล์ จากแหล่งอื่นๆที่ใส่มาในแพ็คเกจ หรือจากเครือข่าย เมื่อสร้างรูปเสร็จแล้ว จะไม่สามารถแก้ไขได้อีก รูปที่ใส่ใน `Alert` , `Choice` หรือ `Item` จะต้องเป็นรูปที่แก้ไขไม่ได้เท่านั้น

5.3.2 รูปที่แก้ไขได้ จะสร้างในหน่วยความจำที่อยู่นอกจอ เสมือนกับว่าเป็นหน้าจอหรือ `Canvas` อีกอันหนึ่ง แอปพลิเคชันสามารถวาดรูปที่แก้ไขนี้ได้จากที่สร้างขึ้นมาแล้ว ด้วยเมธอด `Image.createImage(int width,int height)` โดยรูปที่แก้ไขได้จะใช้ในเทคนิคแบบบัพเพอร์ 2 ชั้น

พารามิเตอร์ของรูป ออบเจกต์ของ `image` จะมีพารามิเตอร์ 3 แบบด้วยกันคือ `height` `width` และ `isMutable` แอปพลิเคชันสามารถอ่านค่าพารามิเตอร์ได้โดยใช้เมธอด `getHeight()` `getWidth()` และ `isMutable()` ตามลำดับ ถ้ารูปนั้นแก้ไขได้ แอปพลิเคชันจะสามารถสร้างออบเจกต์ `Graphics` เพื่อสร้างรูปโดยใช้ `getGraphics()`

5.4 อีเวนต์และการจัดการอีเวนต์

ส่วนติดต่อกับผู้ใช้ประกอบด้วย 2 ส่วนหลักคือ หน้าจอที่ทำหน้าที่แสดงผล และตอบสนองการโต้ตอบของผู้ใช้ ส่วนติดต่อกับผู้ใช้ในอีเวนต์จะทำหน้าที่ประมวลผลอีเวนต์ที่เกิดขึ้น อีเวนต์และตัวจัดการอีเวนต์มีทั้งระดับสูงและระดับล่างเช่นเดียวกันกับ API ของ `Displayable`

บทที่ 6

การใช้ API ระดับสูงของภาษาจาวา

6.1 บทนำ

API ระดับสูงเน้นความสามารถในการโยกย้ายไปใช้กับอุปกรณ์อื่นๆ ชั้นคลาสของ Screen มี 2 กลุ่ม กลุ่มแรก ได้แก่ List TextBox และ Alert ซึ่งมีโครงสร้างที่กำหนดไว้ล่วงหน้าแล้ว กับกลุ่มที่สองได้แก่คลาส Form ซึ่งกำหนดโครงสร้างโดยแอปพลิเคชัน

6.2 List และ Choice

เมื่อเริ่มการทำงาน แอปพลิเคชันมักจะแสดงเมนูซึ่งประกอบด้วยฟังก์ชันต่างๆ ให้คุณได้เลือกตามต้องการ List และ ChoiceGroup ต่างแสดงรายการทางเลือก โดยทั้งคู่ใช้อินเตอร์เฟซ Choice ที่กำหนดไว้ในแพ็คเกจ javax.microedition.lcdui

6.2.1 List

เป็น Screen ซึ่งมีโครงสร้างที่กำหนดไว้แล้วโดยใช้อินเตอร์เฟซ Choice API ของแพ็คเกจ javax.microedition.lcdui กำหนด constructor ของ list ไว้ 2 แบบ โดยตัวแรกจะสร้าง List เปล่าๆ ขึ้นมา ซึ่งสามารถเพิ่มรายการของ choice เข้าไปได้ทีหลัง constructor ตัวที่สองจะสร้าง List ที่มีค่าเริ่มต้นอยู่ข้างใน อาร์เรย์ stringElements ต้องไม่เป็น null และสมาชิกทุกตัวในอาร์เรย์ต้องไม่เป็น null ด้วยเช่นกัน ความยาวของอาร์เรย์ stringElements จะกำหนดจำนวนรายการใน List อาร์เรย์ imageElements อาจมีค่าเป็น null ซึ่งหมายความว่ารายการใน List นั้นไม่มีภาพอยู่ ถ้าอาร์เรย์ imageElements ไม่เป็น null ก็จะต้องมีความยาวเท่ากับอาร์เรย์ stringElements รายการแต่ละรายการในอาร์เรย์ imageElements อาจจะเป็น null ได้ซึ่งหมายความว่าไม่มีภาพในรายการของ List ที่ตรงกัน

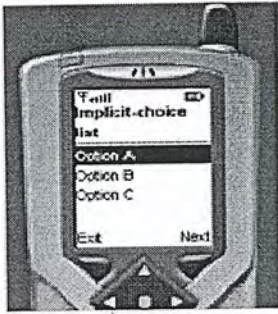
6.2.2 Choice

ออบเจกต์ Choice มี 3 ประเภทได้แก่ implicit-choice (ใช้ได้กับ List เท่านั้น) exclusive-choice และ multiple-choice

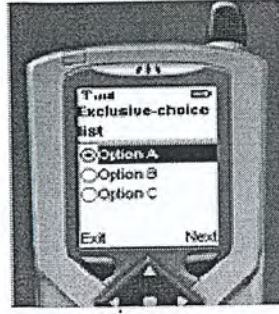
6.2.2.1 Exclusive-choice จะต้องเลือก รายการใดรายการหนึ่ง ในแต่ละครั้งเว้นเสียแต่ว่าไม่มีรายการใดๆ ให้เลือก

6.2.2.2 Implicit-choice เป็นกรณีพิเศษของ exclusive-choice เมื่อรายการที่อยู่ในโฟกัสถูกเลือกโดยอัตโนมัติเมื่อเริ่มต้น Command โดย choice ประเภทนี้ใช้ได้กับ List เท่านั้น

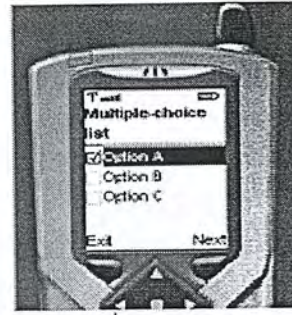
6.2.2.3 Multiple-choice สามารถเลือกหลายรายการพร้อมกันได้ในแต่ละครั้งหรือจะไม่เลือกเลยก็ได้



รูปที่ 6.1 List แบบ

Implicit- choice

รูปที่ 6.2 List แบบ

exclusive- choice

รูปที่ 6.3 List แบบ

multiple - choice

ออบเจกต์ Choice จะแสดงรายการทางเลือกแบบต่างๆ ให้ผู้ใช้เลือก แต่ละรายการประกอบด้วยสตริงข้อความและรูปซึ่งอาจจะมีหรือไม่มีก็ได้

จำนวนรายการในออบเจกต์ Choice สามารถดูได้โดยใช้เมธอด `size()` แต่ละรายการสามารถอ้างอิงได้ตามดัชนีซึ่งเป็นตัวเลขจำนวนเต็มมีค่าตั้งแต่ 0 จนถึง `size() - 1` โดยที่ 0 อ้างถึงรายการแรกและ `size() - 1` อ้างถึงรายการสุดท้าย

ภาพในรายการ Choice อาจเว้นว่างได้ถ้าแอปพลิเคชันนั้นไม่ได้กำหนดไว้ หรือแม้จะกำหนดไว้ก็อาจเลือกไม่แสดงภาพเลย หากอุปกรณ์นั้นๆ ไม่สามารถแสดงผลได้ ถ้าเลือกที่จะแสดงภาพภาพจะติดอยู่กับสตริงข้อความและถือเสมือนว่าทั้งภาพและข้อความนั้นเป็นหน่วยเดียวกัน ภาพไม่สามารถเปลี่ยนได้ แอปพลิเคชันสามารถกำหนดภาพให้กับรายการ `elementNum choice` ได้โดย `Void set(int elementNum, String stringPart, Image imagePart)` และอ่านภาพจาก `elementNum choice` ได้โดย `Image getImage(int elementNum)`

6.3 Textbox

เป็น Screen ที่มีโครงสร้างกำหนดไว้แล้ว ผู้ใช้งานสามารถป้อนหรือแก้ไขข้อความใน TextBox ได้ การสร้างออบเจกต์ TextBox ทำได้โดยใช้ constructor ดังนี้

```
TextBox(String title, String text, int maxSize, int constraints)
```

ข้อความใน TextBox จะเก็บเป็นอาร์เรย์ของ char ดังนั้นจึงมีขนาด (ความจุ) ในการเก็บกลุ่มตัวอักษรไว้ในออบเจกต์อย่างจำกัด โดยสามารถกำหนดขนาดสูงสุดของ TextBox ในขณะสร้างออบเจกต์หรืออาจใช้เมธอด `setMaxSize()` กำหนดหลังจากที่สร้างออบเจกต์ TextBox แล้ว การดูขนาดความจุทำได้โดยใช้เมธอด `getMaxSize()` เนื่องจากอุปกรณ์ไร้สายมักมีหน่วยความจำจำกัด จึงจำกัดสูงสุดที่แท้จริงขึ้นอยู่กับค่าสูงสุดที่ MIDP กำหนดไว้ ดังนั้นขนาดสูงสุดจริงของ TextBox จะจำกัดอยู่ภายในกรอบที่กำหนดโดย MIDP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบขนาดความสูงสุดเกิดขึ้นเมื่อสร้างออบเจกต์ TextBox เมื่อผู้ใช้งานแก้ไขข้อความภายใน TextBox และเมื่อ โปรแกรมเรียกใช้เมธอดเพื่อแก้ไขเนื้อหาของ TextBox เมื่อใดก็ตามที่ข้อความใน TextBox มีขนาดเกินกว่าที่กำหนดไว้ จะเกิด IllegalArgumentException ขึ้น

ข้อความใน TextBox อาจมีความยาวเกินกว่าจะแสดงผลได้หมดในครั้งเดียว ในกรณีนี้ผู้ใช้สามารถเลื่อนหน้าจอขึ้น-ลง เพื่อดูหรือแก้ไขข้อความส่วนต่างๆ การเลื่อนหน้าจอนี้ไม่ทำให้เกิดอีเวนต์ application-visible แต่อย่างไร

6.4 Alert

เป็น Screen ที่มีโครงสร้างกำหนดไว้แล้ว ใช้สำหรับแสดงข้อมูล (ข้อความและภาพ) ให้ผู้ใช้และจะคอยเป็นเวลาครู่หนึ่ง (กำหนดโดย timeout) ก่อนที่จะแสดงหน้าจอถัดไป มี Constructor 2 เมธอด ดังนี้

```
Alert(String title)
```

```
Alert(String title, String alertText, Image alertImage, AlertType alertType)
```

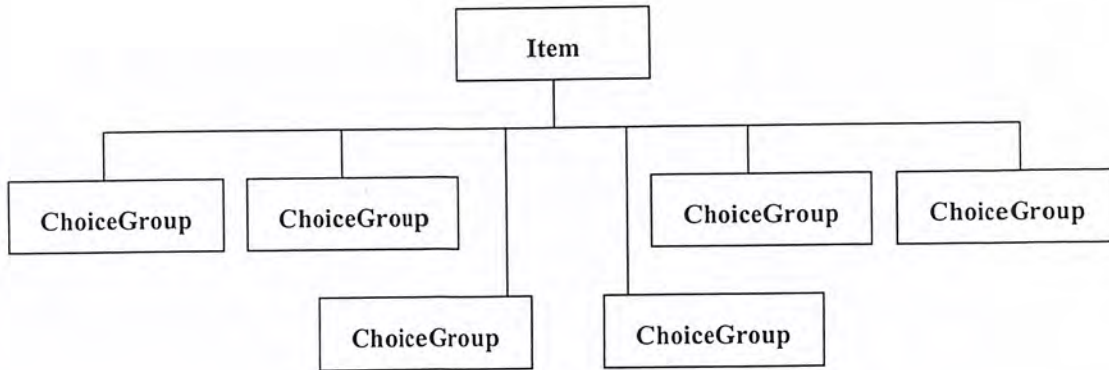
6.5 Form และ Item

Form เป็นออบเจกต์ Screen ซึ่งสามารถบรรจุ Item ต่างๆเช่น ภาพ ข้อความ หรือ choice จำนวนเท่าใดก็ได้ การสร้างออบเจกต์ Form ทำได้ 2 แบบ คือ สร้าง Form เปล่าที่ไม่มี Item ใดๆโดยใช้ Form(String title) หรือสร้าง Form ที่มี Item เริ่มต้นโดยใช้ Form(String title, Item[] item)

สามารถดูจำนวนของ Item ใน Form ได้โดยใช้เมธอด size() การอ้างอิง Item ทำได้โดยใช้ดัชนีตัวเลข ซึ่งมีค่าตั้งแต่ 0 ถึง size() - 1 อ้างอิงตัวสุดท้าย สามารถดึงออบเจกต์ Item จาก Form ได้โดยใช้ get(int itemNum)

Item หนึ่งจะอยู่ได้ใน Form เดียวเท่านั้น หากแอปพลิเคชัน โปรแกรมพยายามใส่ Item ที่มีแล้วใน Form นี้หรือ ในฟอร์มอื่นๆ ลงไปในอีก Form หนึ่งก็จะเกิด IllegalStateException ขึ้น ดังนั้นจะต้องย้าย Item ออกจาก Form ที่อยู่เดิมก่อนจะสามารถใส่เข้าไปใน Form ใหม่ได้

6.5.1 Item เป็นซูปเปอร์คลาสสำหรับคอมโพเนนต์ต่างๆที่สามารถใส่ลงไปใน Form ได้ แต่ละ Item จะมีป้ายข้อความกำกับ ซึ่งเป็นฟิลด์สตริงข้อความที่ติดอยู่กับ Item โดยปกติจะแสดงไว้ใกล้ๆ กับคอมโพเนนต์ที่ปรากฏในหน้าจอ



รูปที่ 6.4 โครงสร้างลำดับชั้นของคลาส Item

6.5.1.1 **ChoiceGroup** คือ Item ที่แสดงผลแบบ Choice มีลักษณะคล้ายกับ List แต่ที่ไม่มีแบบ implicit-choice การแสดงผลแบบ exclusive หรือ multiple-choice ของ ChoiceGroup จะเหมือนกับ List การสร้าง ChoiceGroup ทำได้ดังนี้ `ChoiceGroup(String label, int choiceType)`

6.5.1.2 **ImageItem** อาจมีออบเจกต์ Image ซึ่งสามารถเปลี่ยนแปลงได้ การสร้างออบเจกต์ ImageItem ทำได้ดังนี้ `ImageItem(String label, Image img, int layout, String altText)`

6.5.1.3 **StringItem** ประกอบด้วยป้ายและสตริงข้อความที่แสดงผลอย่างเดียวกันซึ่งผู้ใช้ไม่สามารถแก้ไขค่าใน StringItem ได้โดยตรง ต้องทำผ่านแอปพลิเคชันเท่านั้นโดยใช้เมธอด `setText()` การสร้างออบเจกต์ของ StringItem ทำได้ดังนี้ `StringItem(String label, String text)`

6.5.1.4 **Gauge** เป็น Item ที่แสดงผลแบบกราฟแท่งโดยมีค่าระหว่าง 0 ถึง `maxValue` การสร้างออบเจกต์ Gauge ทำได้ดังนี้ `Gauge(String label, Boolean interactive, int maxValue, int initialValue)`

6.5.1.5 **DateField** คือ Item ที่แก้ไขได้ ใช้สำหรับแสดงค่าวัน เดือน ปี และ เวลา สามารถสร้างออบเจกต์ DateField โดยใช้ `DateField(String label, int mode, TimeZone timeZone)`

6.5.1.6 **TextField** เป็น Item ที่มีสตริงข้อความซึ่งผู้ใช้สามารถแก้ไขได้ การสร้างออบเจกต์ TextField ทำได้ดังนี้ `TextField(String label, String text, int maxSize, int constraints)`

ข้อแตกต่างระหว่าง TextField และ TextBox คือ TextBox เป็น subclass ของ Screen ดังนั้นจึงสามารถแสดงผลบนหน้าจอได้ทันที ทว่า TextField เป็น subclass ของ Item จะสามารถแสดงผลได้ก็ต่อเมื่อนำไปไว้ใน Form

6.5.2 **Form** เป็นคอมโพเนนต์ที่สำคัญที่สุดในการพัฒนาส่วนติดต่อกับผู้ใช้สำหรับอุปกรณ์ MIDP เพราะช่วยให้ผู้พัฒนาสามารถสร้างหน้าจอได้อย่างยืดหยุ่น

ในการนำ MIDP มาใช้งาน ได้ระบุการวางโครงร่างของ Item ใน Form สำหรับอุปกรณ์ต่างๆ ไปไว้ว่าต้องเป็นแนวตั้ง Item ที่โฟกัสได้เช่น TextField, DateField, Gauge หรือ ChoiceGroup จะขึ้นบรรทัดใหม่ เสมอ สำหรับ StringItem และ ImageItem ซึ่งไม่เกี่ยวข้องกับคำตอบของผู้ใช้จะวางใน

แนวอนนอกเสียจากว่าสตริงหรือคำสั่ง Layout ของ ImageItem จะกำหนดให้ขึ้นบรรทัดใหม่ ข้อความเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ใดๆ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน StringItem หรือ ภาพใน ImageItem จะถูกตัดขึ้นบรรทัดใหม่(กรณีข้อความ) หรือถูกคลิบบางส่วนออก (กรณีภาพ) เพื่อให้พอดีกับความกว้างของหน้าจอ หากจำเป็นต้องเลื่อนจอ ทำให้เฉพาะในแนวตั้งเท่านั้น

6.6 อีเวนต์ระดับสูงและการจัดการอีเวนต์

โมเดลการจัดการอีเวนต์ระดับสูงเป็นแบบ delegation-based เหมือนกับใน AWT ของ J2SE ในโมเดลนี้ประกอบด้วย 2 ส่วนคือ แหล่งกำเนิดอีเวนต์ และตัว event listener แหล่งกำเนิดอีเวนต์จะสร้างอีเวนต์แล้วส่งไปยัง event listener จากนั้นเมื่อกู้ดที่จัดการอีเวนต์ของ listener ก็จะประมวลผลอีเวนต์นั้น

ออบเจกต์ของส่วนติดต่อกับผู้ใช้ใน MIDP จะใช้อีเวนต์ระดับสูง 2 ตัวคือ อีเวนต์ command และ ItemStateChanged ดังนั้นจึงมีอีเวนต์ listener 2 ตัวคือ commandListener และ ItemStateListener ตามลำดับ

6.6.1 Command เป็นโครงสร้างที่เก็บความหมายของแอคชันหรืออีเวนต์ แอคชันจริง (หรือการจัดการอีเวนต์) ซึ่งเกิดเมื่อมี command เกิดขึ้นได้กำหนดไว้ในอินเทอร์เฟซของ CommandListener เมื่อ command เริ่มต้นจะเกิดอีเวนต์แล้วส่งไปยังออบเจกต์ CommandListener ที่รีจิสเตอร์ไว้ ออบเจกต์ Command มีข้อมูล 3 อย่างคือ เลขเบส , ชนิด และ ลำดับความสำคัญ

6.6.2 ItemStateChanged เมื่อ Form แสดงผลบนจอ ผู้ใช้สามารถเลื่อนโฟกัสจาก Item หนึ่งไปยังอีก Item หนึ่งภายใน Form ได้ โดยไม่เกิดอีเวนต์ให้แอปพลิเคชันสามารถตรวจเห็นได้แต่หากผู้ใช้แก้ไขเปลี่ยนแปลงสถานะของ Item ที่โต้ตอบได้ซึ่งอยู่ภายใน Form จะทำให้เกิด ItemStateChanged ขึ้น แต่ถ้าเป็นการเปลี่ยนแปลงค่า Item โดยแอปพลิเคชันด้วยเมธอด setString() ก็จะไม่ทำให้เกิดอีเวนต์ ItemStateChanged

บทที่ 7

หลักการเขียนโปรแกรมเครือข่ายของ J2ME

7.1 บทนำ

การเขียนโปรแกรมเครือข่าย นำเอาจุดเด่นเรื่องความสามารถในการเชื่อมต่อของอุปกรณ์เหล่านี้มาใช้ มีบทบาทสำคัญในการพัฒนาแอปพลิเคชันสำหรับอุปกรณ์ไร้สาย

7.2 การเขียนโปรแกรมเครือข่ายด้วย J2SE และ J2ME

การเขียนโปรแกรมสำหรับเครือข่ายด้วย J2SE จะง่ายและตรงไปตรงมา ซึ่งใน J2SE ได้ให้ไลบรารีของเครือข่ายซึ่งมีฟังก์ชันให้ใช้มากมาย ในแพ็คเกจ `java.io` มีคลาสที่สนับสนุนการนำเข้า – ส่งออกข้อมูลประมาณ 60 คลาส ขณะที่อีกประมาณ 20 คลาสในแพ็คเกจ `java.net` สนับสนุนการทำงานด้านเครือข่าย

อย่างไรก็ดี คลาสส่วนใหญ่ ในแพ็คเกจทั้งสองออกแบบมาสำหรับระบบคอมพิวเตอร์ทั่วไปที่มีหน่วยประมวลผล หน่วยความจำ และพื้นที่เก็บข้อมูลที่เพียงพอ ไฟล์คลาสเหล่านี้มีขนาดคงที่ทั้งสิ้น ประมาณ 200 กิโลไบต์ ซึ่งใหญ่เกินกว่าที่จะใส่ลงไปในอุปกรณ์ไร้สาย ซึ่งมีทรัพยากรต่างๆ จำกัดและมีพื้นที่เก็บข้อมูลเพียงไม่กี่ร้อยกิโลไบต์

ไม่เฉพาะขนาดของไฟล์เท่านั้นที่เป็นอุปสรรคในการใช้งานบนอุปกรณ์ไร้สาย Java 2 Micro Edition จะต้องรองรับอุปกรณ์เคลื่อนที่ได้หลากหลายประเภท การติดต่อเครือข่ายและไลบรารีของไฟล์ I/O ในแต่ละอุปกรณ์แตกต่างกันมาก เนื่องจากความต้องการขั้นต่ำในการเชื่อมต่อและความสามารถของ I/O ที่แตกต่างกัน ความแตกต่างระหว่างเครือข่ายทำให้ต้องกำหนดสาระสำคัญของการสื่อสาร (abstraction) ในจาวาไลบรารี 2 รูปแบบคือ การเชื่อมต่อแบบค่าโปรแกรมสำหรับเครือข่ายสลับกลุ่มข้อมูลและคือ การเชื่อมต่อแบบซ็อกเก็ตสำหรับเครือข่ายแบบสลับวงจร ทั้งนี้ ผู้ผลิตซึ่งสนับสนุนค่าโปรแกรมอาจไม่สนใจจะสนับสนุนซ็อกเก็ตก็ได้ และเช่นเดียวกันในกรณีของผู้ผลิตซึ่งสนับสนุนซ็อกเก็ต

การเข้าถึงไฟล์ I/O ในอุปกรณ์ไร้สายก็เป็นเรื่องเฉพาะอุปกรณ์และมีวิธีต่างๆ ในการนำไปใช้เช่นกัน ผู้ผลิตที่สนับสนุนกลไก I/O ประเภทหนึ่งแล้วมักจะไม่นับสนับสนุนกลไกประเภทอื่นๆ อีกเนื่องจากข้อจำกัดด้านหน่วยความจำ

7.3 กรอบการติดต่อสื่อสารทั่วไป (The Generic Connection Framework)

นำกรอบการติดต่อสื่อสารทั่วไปเข้ามาใช้ใน CLDC ของ J2ME ก็เพื่อที่จะสะท้อนให้เห็นถึงความต้องการเครือข่ายขนาดเล็กและไฟล์ I/O สำหรับอุปกรณ์เคลื่อนที่ประเภทต่างๆ

กรอบการติดต่อสื่อสารทั่วไปสร้างฟังก์ชันเครือข่ายและคลาสไฟล์ I/O จากแพ็คเกจ `java.io` และ `java.net` ของ J2SE หรือเป็นซับเซตของคลาสใน J2SE แต่มีขนาดเล็กกว่า คลาสและอินเตอร์เฟสทั้งหมดของ J2ME รวมอยู่ในแพ็คเกจเดียว คือ `javax.microedition.io`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้กรอบการติดต่อสื่อสารทั่วไปมีความยืดหยุ่นและสนับสนุนอุปกรณ์ได้หลากหลายที่ต้องการ จะนำสาระสำคัญในการติดต่อสื่อสารรูปแบบต่างๆมาใช้ ได้แก่ อินเทอร์เน็ตการติดต่อ 7 ลักษณะ คือ Connection, ContentConnection, DatagramConnection, InputConnection, OutputConnection, StreamConnection และ StreamConnectionNotifier

7.4 การเขียนโปรแกรมเครือข่ายบนอุปกรณ์ไร้สายโดยการเชื่อมต่อแบบซ็อกเก็ต

ซ็อกเก็ตเป็นกลไกการติดต่อสื่อสารระหว่างโปรแกรมที่ทำงานบนเครือข่ายปลายทาง 2 แห่ง อาทิ อุปกรณ์ไร้สายกับเซิร์ฟเวอร์ระยะไกล หรือระหว่างอุปกรณ์ไร้สาย 2 อุปกรณ์ มีประสิทธิภาพน่าเชื่อถือความสามารถในการเชื่อมต่อแบบซ็อกเก็ตในอุปกรณ์เคลื่อนที่บางประเภททำให้เกิดแอปพลิเคชันที่เขียนด้วย J2ME บนไคลเอนต์และเซิร์ฟเวอร์ต่างๆ การเชื่อมต่อแบบซ็อกเก็ตกับเซิร์ฟเวอร์ SMTP และ POP3 ทำให้ไคลเอนต์อีเมลบนอุปกรณ์ไร้สายสามารถรับและส่งอีเมลข้อความผ่านระบบอินเทอร์เน็ตได้ตามปกติ

ซ็อกเก็ตช่วยนักพัฒนา J2ME สามารถพัฒนาแอปพลิเคชันทุกชนิดเพื่อใช้งานบนอุปกรณ์ไร้สายได้ อย่างไรก็ตามผู้ผลิตอุปกรณ์บางรายไม่สามารถสนับสนุนการเชื่อมต่อแบบซ็อกเก็ตในอุปกรณ์ MIDP ซึ่งหมายถึงว่าแอปพลิเคชันที่พัฒนาโดยใช้ซ็อกเก็ตอาจจะจำกัดเฉพาะในอุปกรณ์ไร้สายบางชนิด และเคลื่อนย้ายข้ามเครือข่ายอุปกรณ์ไร้สายต่างๆ ได้น้อยลง

ในการนำซ็อกเก็ตมาใช้ ก่อนอื่นผู้รับผู้ส่งจะต้องสร้างการเชื่อมต่อระหว่างซ็อกเก็ตเสียก่อน โดยฝ่ายหนึ่งจะคอยฟังคำร้องขอเชื่อมต่อ ขณะที่อีกฝ่ายหนึ่งขอเชื่อมต่อเครือข่าย เมื่อซ็อกเก็ตทั้ง 2 ด้านเชื่อมต่อเข้าด้วยกัน ก็จะส่งผ่านข้อมูลไปยังอีกด้านหนึ่ง

การรับข้อมูลจากเครื่องเซิร์ฟเวอร์ระยะไกลจะต้องสร้าง InputConnection ขึ้นมาเพื่อรับ InputStream จากการเชื่อมต่อนั้น การส่งข้อมูลไปยังเครื่องเซิร์ฟเวอร์ระยะไกลจะต้องสร้าง OutputConnection ขึ้นมาเพื่อรับ OutputStream จากการเชื่อมต่อนั้น ใน J2ME การเชื่อมต่อทั้ง 3 ประเภทที่กำหนดไว้เพื่อจัดการกับ Stream ข้อมูลขาเข้า / ขาออก ได้แก่ InputStream OutputStream และ StreamConnection โดยมีความหมายตามชื่อเรียก คือ InputConnection กำหนดให้ input stream สามารถรับส่งข้อมูลได้ ขณะที่ OutputConnection ก็กำหนดให้ output stream สามารถส่งข้อมูลได้ การจะนำการเชื่อมต่อประเภทใดมาใช้เมื่อไหร่ขึ้นอยู่กับการรับ - ส่ง หรือทั้งรับและส่งข้อมูล

การเขียนโปรแกรมแบบเครือข่ายโดยใช้ซ็อกเก็ตใน J2ME นั้นเขียนอย่างตรงไปตรงมาโดยมีขั้นตอนการทำงานดังนี้

1. ปิดการเชื่อมต่อแบบซ็อกเก็ตกับเซิร์ฟเวอร์ระยะไกลหรืออุปกรณ์ไร้สายอื่นโดยใช้ Connector.open()
2. สร้าง InputStream และ OutputStream จากเปิดการเชื่อมต่อแบบซ็อกเก็ตเพื่อใช้ในการรับ - ส่ง กลุ่มข้อมูล
3. รับ - ส่งข้อมูลจากเซิร์ฟเวอร์ระยะไกลผ่านการเชื่อมต่อแบบซ็อกเก็ต ด้วยโอเปอเรชันการอ่าน

และเขียนในออบเจกต์ InputStream หรือ OutputStream

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เปิดการเชื่อมต่อแบบซ็อกเก็ตและ input / output stream ก่อนออกจากโปรแกรม

7.5 การเขียนโปรแกรมเครือข่ายไร้สายโดยใช้ดาต้าแกรม

ดาต้าแกรม คือ สารที่มีเนื้อหาอยู่ในตัวสามารถส่งผ่านข้ามเครือข่ายโดยไม่อาศัยปัจจัยอื่น ๆ เข้าช่วย ทว่าไม่สามารถกำหนดชัดได้ว่าสารจะถึงมือผู้ใช้งานเมื่อไรและเวลาใด เป็นกลไกการสื่อสารที่ใช้กลุ่มข้อมูล มีข้อมูล มีข้อแตกต่างจากการสื่อสารแบบ Stream ตรงที่ไม่มีการเปิดพอร์ตเพื่อเชื่อมต่อระหว่างผู้รับและผู้ส่ง

เครือข่ายไร้สายแบบสลับกลุ่มข้อมูลมักสนับสนุนดาต้าแกรมบนอุปกรณ์ MIDP ขณะที่เครือข่ายไร้สายแบบสลับวงจรอาจไม่สนับสนุนดาต้าแกรมทำให้แอปพลิเคชันบนอุปกรณ์ไร้สายที่พัฒนาโดยใช้ดาต้าแกรมจำกัดอยู่เฉพาะในบางอุปกรณ์ และอาจไม่สามารถโยกย้ายข้ามเครือข่ายต่างๆ ได้

ดาต้าแกรมจะอาศัยพอร์ต UDP ในการเชื่อมต่อ ผู้ส่งจะสร้างกลุ่มข้อมูลดาต้าแกรมโดยกำหนดข้อมูลปลายทาง (ที่อยู่ในอินเทอร์เน็ตและหมายเลขพอร์ต) และส่งออกไป เลขอร์เครือข่ายระดับล่างจะไม่สามารถรับประกันการส่งข้อมูลไปยังปลายทาง เซิร์ฟเวอร์อาจจะไม่ได้รับดาต้าแกรมที่คุณส่งออกมาเลย หรือแม้ว่าเซิร์ฟเวอร์จะได้รับกลุ่มข้อมูลดังกล่าว อุปกรณ์ไร้สายของคุณก็อาจไม่ได้รับการตอบรับใด ๆ ก็ได้ เนื่องจากโปรโตคอลของพอร์ต UDP ไม่สามารถรับประกันการส่งได้ จึงไม่เหมาะสำหรับแอปพลิเคชันประเภท FTP (File Transfer Protocol) ซึ่งต้องอาศัยการส่งข้อมูลที่เชื่อถือได้ อย่างไรก็ตาม พอร์ต UDP มีประโยชน์ในกรณีดังต่อไปนี้

- เมื่อความเร็วในการสื่อสารมีความสำคัญมากกว่าจะต้องส่งข้อมูลทุกบิตให้ครบถ้วนถูกต้อง ตัวอย่างเช่น แอปพลิเคชันเกี่ยวกับเสียง / สื่อวีดิโอตามเวลาจริง (real time) บนโทรศัพท์เคลื่อนที่ เมื่อกลุ่มข้อมูลหายเสียง / รูปเคลื่อนที่ก็จะหยุดนิ่ง ซึ่งดีกว่าเมื่อเทียบการเชื่อมต่อข้อมูลแบบซ็อกเก็ตที่เสียงจะขาดหายไปเป็นช่วงๆ
- เมื่อต้องส่งข้อมูลเป็นประจำ และการบริการไม่ได้รับผลกระทบต่อมากนัก หากกลุ่มข้อมูลหายในบางครั้ง
- เมื่อไม่รองรับการเชื่อมต่อข้อมูลแบบซ็อกเก็ตเลย ซึ่งมักเกิดขึ้นกับเครือข่ายไร้สายแบบสลับกลุ่มข้อมูล

ขั้นตอนทั่วไปในการนำการเชื่อมต่อแบบดาต้าแกรมมาใช้ในแอปพลิเคชัน MIDlet

1. สร้างการสื่อสารแบบดาต้าแกรม
2. สร้างและส่งออกเจ็ทดาต้าแกรมพร้อมเนื้อหาของสารและที่อยู่ปลายทาง
3. ส่งสารดาต้าแกรมไปตามการเชื่อมต่อที่สร้างไว้แล้ว
4. สร้างออบเจกต์รับดาต้าแกรมด้วยบัพเฟอร์ที่จัดสรรไว้แล้ว
5. รอรับสารตามการเชื่อมต่อที่สร้างไว้แล้วโดยใช้บัพเฟอร์ดาต้าแกรมที่จัดสรรไว้แล้ว
6. ยกเลิกการเชื่อมต่อเมื่อเสร็จงานแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.6 การเขียนโปรแกรมเครือข่ายสำหรับอุปกรณ์ไร้สายโดยใช้ HttpURLConnection

การเขียนโปรแกรมเครือข่ายที่อาศัย HTTP สามารถทำได้ง่ายและ ตรงไปตรงมามากยิ่งขึ้น ตัวอย่างเช่น เมธอดที่ช่วยให้นักพัฒนาอ่านข้อมูลส่วนหัวของ HTTP ได้ง่ายกว่าเดิม การนำ HttpURLConnection มาใช้ในการติดต่อสื่อสารในแอปพลิเคชันที่มีข้อดีหลายประการด้วยกัน อาทิ

- อุปกรณ์ MIDP บางอุปกรณ์ ไม่สนับสนุนการติดต่อสื่อสารแบบซ็อกเก็ตและคาล์วแกรม ขณะที่ทุกอุปกรณ์สนับสนุนการติดต่อสื่อสารแบบ HTTP
- การติดต่อสื่อสารแบบซ็อกเก็ตและคาล์วแกรมต้องอาศัยเครือข่ายค่อนข้างมาก ขณะที่บางเครือข่ายอาจเลือกใช้รูปแบบการติดต่อสื่อสารเพียงแบบใดแบบหนึ่งและไม่สนับสนุนอื่นๆ ทำให้โยกย้ายแอปพลิเคชันสำหรับอุปกรณ์ไร้สายได้น้อยลง
- การกำหนดให้อุปกรณ์ MIDP สนับสนุนโปรโตคอล HTTP ทำให้แอปพลิเคชันบนอุปกรณ์ไร้สายมีโปรโตคอลระดับสูง (HttpURLConnection) เป็นมาตรฐาน และไม่ต้องอาศัยเครือข่ายไว้ใช้งาน นอกจากนี้ยังสามารถโยกย้ายข้ามเครือข่ายไร้สายต่างๆ ได้ดีอีกด้วย
- สามารถเก็บข้อมูลประเภทต่างๆ ลงในการร้องขอใช้ HTTP ได้ง่ายโดยเฉพาะอย่างยิ่งหากนักพัฒนานำ XML มาใช้ในแอปพลิเคชันด้วย

การร้องขอ HttpURLConnection ไปยังเว็บเซิร์ฟเวอร์ทำได้ 3 ทางด้วยกัน คือ GET HEAD และ POST โดยโปรแกรมนำเมธอด GET มาใช้เพื่ออ่านเนื้อหาของเอกสารในเว็บจาก URL ที่กำหนดไว้ การตอบรับของเว็บเซิร์ฟเวอร์ประกอบด้วยข้อมูลส่วนหัว และ เอกสารในเว็บข้อมูลแบบ MIME เกี่ยวกับเนื้อหาข้อมูลและตัวเนื้อหาข้อมูล และนำเมธอด HEAD มาใช้เพื่ออ่านข้อมูลเกี่ยวกับเอกสารในเว็บแทนที่จะเป็นเนื้อหาของเอกสารนั้นๆ เมื่อเว็บเซิร์ฟเวอร์ได้รับการร้องขอ HEAD จะมีเพียงข้อมูลส่วนหัวของ HTTP เท่านั้นที่จะตอบกลับมา (ไม่มีเนื้อหาข้อมูล) และนำเมธอด POST มาใช้เพื่อส่งข้อมูลเกี่ยวกับลักษณะไปยัง URL ของโปรแกรม CGI โดยทั้ง POST และ GET ต่างถูกนำมาใช้เพื่อส่งข้อมูลไปยังโปรแกรม CGI ข้อแตกต่างอยู่ที่ POST ส่งข้อมูลแบบ stream ในขณะที่เมธอด GET ส่งข้อมูลผ่าน environment ของตัวแปรที่ฝังตัวในสตรีมการสอบถาม

7.6.1 การใช้เมธอด GET กับ HttpURLConnection

เมธอดโดยปริยายที่ HttpURLConnection ใช้ในการร้องขอ คือ GET โดยว่าถือข้อมูลทั้งหมดเป็นเสมือนส่วนหนึ่งของสตรีม URL ในโค้ดตัวอย่างต่อไปนี้ จะส่งการร้องขอ GET ไปยังเซิร์ฟเวอร์ 64.28.105.110

<http://64.28.105.110/servlets/webyu/Chapter9Servlet?request=gettimestamp>

Java Servlet Chapter9Servlet จะรับคำร้องขอ ดูเวลาที่ท้องถิ่นปัจจุบัน แล้วส่งกลับมายังไคลเอ็นต์ ในการร้องขอ GET สตรีมการสอบถามจะฝังตัวเป็นส่วนหนึ่งของสตรีม URL ตัวอย่างเช่น หาก เรียก getQuery() ขึ้นมาที่ HttpURLConnection ค่าของ request=gettimestamp จะถูกส่งคืนกลับมา เมื่อการร้องขอเอกสารนี้เป็นเอกสารที่ส่งมาสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GET ส่งไปยัง Java Servlet Environment ของตัวแปร request และค่าของตัวแปรจะส่งผ่านไปยังโปรแกรมบนฝั่งเซิร์ฟเวอร์เช่นกัน

7.6.2 การใช้เมธอด HEAD กับ HttpURLConnection

เซิร์ฟ HTTP กำหนดเนื้อหาส่วนหัวของ HTTP (อยู่ส่วนบนก่อนถึงคำตอบรับ) ไว้ค่อนข้างมาก ในตัวอย่างนี้แสดงส่วนหัวทั่วไปของ HTTP ที่คืนค่าโดยเว็บเซิร์ฟเวอร์ Apache ที่ทำงานบนชั้น โซลาริส

```
HTTP 1.1 200 OK
Data: Mon, 18 Oct 1999 20:06:45 GMT
Server: Apache/1.2.4 (Unix) PHP/3.0.6
Last-Modified: Mon, 18 Oct 1999
Accept-Ranges: bytes
Content-Length: 35259
Content-Type : text/html
```

โดยมากส่วนหัวของ HTTP จะบอกรวมถึงประเภทเนื้อหาของหน้าเว็บที่ร้องขอ ความยาวของเนื้อหา และชุดตัวอักษรที่เนื้อหาใช้เข้ารหัส วันและเวลาของการตอบรับ เวลาปรับปรุงล่าสุดของหน้าที่ร้องขอ วันหมดอายุสำหรับการเก็บแคช

เมื่อมีการร้องขอ HEAD ถูกส่งไปยังเว็บเซิร์ฟเวอร์ จะคืนกลับมาเฉพาะข้อมูลส่วนหัว การร้องขอประเภทนี้ใช้เพื่อประเมินว่า สามารถนำตัวอย่างแคชมาใช้ซ้ำได้อีกหรือไม่ หรือจะต้องแทนที่ด้วยข้อมูลที่ใหม่กว่าโดยดูจากค่าคุณสมบัติที่อ่านมาจากฟิลด์ส่วนหัว

7.6.3 การใช้เมธอด POST กับ HttpURLConnection

การใช้เมธอด POST ส่งการร้องขอ HTTP จะต้องได้รับทั้ง InputStream และ OutputStream จาก HttpURLConnection โดยใช้ InputStream เพื่ออ่านการตอบรับจากเว็บเซิร์ฟเวอร์ และ OutputStream เพื่อส่งข้อมูลแยกผ่าน stream

คล้ายคลึงกับการร้องขอแบบ GET แต่เมื่อ Java Servlet Chapter9Servlet จะรับคำร้องขอ คุณเวลาท้องถิ่นปัจจุบัน แล้วส่งกลับมายังไคลเอ็นต์ ฟังสังเกตว่า ในการร้องขอ POST นี้ ข้อมูลที่ส่งมาคือ request=gettimestamp ไม่ได้เป็นส่วนหนึ่งของ URL อีกแล้วหากแต่จะถูกแยกส่งไปยังเว็บเซิร์ฟเวอร์ต่างหากเมื่อสร้างการเชื่อมต่อแบบ HTTP

บทที่ 8

หลักการเขียน ASP (Active Server Page)

8.1 บทนำ

ASP หรือ Active Server Pages เป็นโปรแกรมตีความภาษา (Interpreter) ที่ใช้ในการตีความเว็บเพจที่เขียนขึ้นมาโดยใช้ไวยากรณ์หรือ syntax ของภาษา VBScript (ซึ่ง VBScript ก็อาศัยโครงสร้างของภาษา Visual Basic อีกที) แล้วสร้างเว็บเพจผลลัพธ์ขึ้นมา จากนั้นก็จะส่งไปให้ web server เพื่อที่จะให้ web server ส่งต่อไปยัง browser อีกที เนื่องจาก ASP จะต้องทำงานโดยการร้องขอของ web server ดังนั้นจึงจะต้องมีโปรแกรม ASP ติดตั้งที่ web server ด้วย โดยที่ปัจจุบัน เมื่อพูดถึง ASP มักจะหมายถึงเป็น ASP ที่ทำงานในวินโดวส์ NT หรือ วินโดวส์ 95 ,98 (ใช้กับธุรกิจหรืองานที่ปริมาณการติดต่อไม่มากนัก หรือใช้ในการทดสอบเพื่อการพัฒนาต่อไปสู่ระบบใหญ่ต่อไป)

8.2 ความรู้เบื้องต้นของภาษา HTML

เป็นภาษาหลักในการเขียน WEB PAGE โดยการเขียนภาษา HTML นั้นสามารถแทรกด้วย VB SCRIPT , JAVA SCRIPT , ASP ได้ โดยโครงสร้างของ WEB PAGE มี 2 ส่วนดังต่อไปนี้

Header : หรือส่วนหัวของเว็บเพจ

ในส่วนนี้นั้นจะบรรจุข้อมูลต่างๆ ที่ใช้อธิบายภาพรวม ควบคุมการทำงาน และควบคุมการแสดงผลของเว็บเพจ ซึ่งข้อมูลเหล่านี้มักจะถูกบราวเซอร์นำไปใช้งาน

Body : หรือส่วนเนื้อหาของเว็บเพจ

ส่วนนี้จะเป็นส่วนที่บรรจุข้อมูลต่างๆ ที่เป็นเนื้อหาของเว็บเพจนั้นไว้ซึ่งสามารถใส่ได้ทั้งข้อความ รูปภาพ หรือ ภาพเคลื่อนไหว และส่วนนี้อาจจะมีส่วนที่เชื่อมโยงไปยังเว็บเพจอื่น ๆ ได้ด้วย

โดยภาษา HTML นั้น จะมี TAG ต่าง ๆ เป็นตัวควบคุมการแสดงผลต่าง โดยอยู่ในเครื่องหมาย < และ >

โดยมี TAG ที่ควรรู้ดังต่อไปนี้

- <HTML> และ </HTML> เป็น TAG ที่ใช้กำหนดควบคุมขอบเขตของเว็บเพจ
- <HEAD> และ </HEAD> เป็น TAG ใช้ครอบส่วนหัวของเอกสาร (header) โดยส่วนใหญ่ จะแสดงข้อความบอกเนื้อหาโดยรวมของเว็บเพจนี้ให้ผู้อ่านได้ทราบ โดยแสดงให้เห็นที่ Title Bar
- <BODY> และ </BODY> เป็น TAG ที่ใช้ครอบส่วนที่เป็นเนื้อหาที่เหลือของเว็บเพจ ซึ่งภายในสามารถจะบรรจุข้อมูลในรูปแบบต่าง ๆ และคำสั่งที่ใช้ควบคุมการแสดงผลของเว็บเพจได้
- <TITLE> และ </TITLE> เป็น TAG ที่อยู่ในส่วน ของ <HEAD> ทำหน้าที่แสดงข้อความที่

ปรากฏบน Title Bar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- และ เป็น TAG ที่ทำหน้าที่กำหนดขนาด สี ตัวอักษรต่าง ๆ ได้
- และ เป็น TAG ที่ทำหน้าที่นำรูปภาพมาแสดงภายในหน้าเว็บเพจ ซึ่งสามารถกำหนดรูปภาพ และขนาดของรูปได้
-
 เป็นการขึ้นบรรทัดใหม่
- <P> เป็นการขึ้นบรรทัดใหม่ที่ใหญ่กว่า

- <HR> เป็นการตีเส้นกั้นบรรทัดนั้น
- <A> และ เป็น TAG ที่ทำหน้าที่ในการสร้าง link ต่างๆ โดยมีลักษณะดังต่อไปนี้ คือ ข้อความหรือรูปภาพ

8.3 ความรู้เบื้องต้นของ VBScript

ในส่วนของ VB SCRIPT นั้น จะต้องครอบด้วย TAG <SCRIPT> </SCRIPT> ทุกครั้ง ดังตัวอย่างต่อไปนี้

```
<SCRIPT>
```

```
</SCRIPT>
```

- โดยในการกำหนดตัวแปรนั้น ใช้วิธีดังนี้

```
Dim <ชื่อตัวแปร>,<ชื่อตัวแปร>,...
```

- ในการกำหนดค่าคงที่นั้น กำหนดได้โดย

```
Const <ชื่อตัวแปร> = <ค่าที่กำหนด>
```

- ในการกำหนดตัวแปรแบบ อาร์เรย์ ทำได้โดย

```
Dim <ชื่อตัวแปรอาร์เรย์> (อินเด็กซ์)
```

หรือ

```
Dim <ชื่อตัวแปรอาร์เรย์> (<ขอบเขตล่าง> To <ขอบเขตบน>)
```

ฟังก์ชันที่พร้อมใช้งานใน VB SCRIPT นั้นมีอยู่ มากมาย จะยกตัวอย่างที่ใช้บ่อยๆดังต่อไปนี้

Date	บอกวันเดือนปีปัจจุบัน
Time	บอกเวลาปัจจุบัน
Now	บอกทั้งวันที่ และ เวลาปัจจุบัน
Day	บอกเฉพาะวันที่
Month	บอกเฉพาะเดือน
Year	บอกเฉพาะปี
Hour	บอกเฉพาะชั่วโมงในขณะนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Minute	บอกเฉพาะนาที่ในขณะนั้น
Weekday	บอกว่าเป็นวันไหนของสัปดาห์
Lcase	แปลงข้อความให้เป็นตัวอักษรเล็กทั้งหมด
Ucase	แปลงข้อความให้เป็นตัวอักษรใหญ่ทั้งหมด
Msgbox	ใช้แสดงข้อความตอบรับกับผู้ใช้
InputBox	เป็นการรับข้อมูลจากผู้ใช้

วิธีการเขียน Sub และ Function โดยทั้งสองอย่างนั้นมีความแตกต่าง กันดังนี้

Sub เป็นโปรแกรมย่อยที่เมื่อจบการทำงานแล้ว จะไม่มีการส่งผลการทำงานกลับมารให้เราทราบ เราจึงมักเขียน Sub ในงานที่มีต้องการเน้นผลการการทำงานกลับมาให้ โดยมีรูปแบบการเขียนดังต่อไปนี้

Sub <ชื่อของ Sub> (พารามิเตอร์ที่ต้องใช้)

...

End Sub

Function เป็นโปรแกรมย่อยที่เมื่อจบการทำงานแล้ว จะต้องส่งผลการการทำงานกลับมาให้ผู้ที่เรียกใช้งานฟังก์ชันนั้น (เหมือนกับที่เราเรียกใช้ฟังก์ชันที่มีอยู่แล้ว) ซึ่งในการเขียนสคริปต์จึงมักกำหนดตัวแปรมา 1 ตัวเพื่อคอยรับผลการทำงานของฟังก์ชัน โดยมีรูปแบบดังนี้

Function <ชื่อของ Function> (พารามิเตอร์ที่ต้องใช้)

...

<ชื่อของ Function> = <ค่าที่คืนกลับไป>

End Function

- การใช้ If ... Then ... Else มีวิธีการดังนี้

If (เงื่อนไข) Then

...

Else

...

End if

โดยถ้าเรามี If คร่อมกันหลาย ๆ ตัว เราจะเป็นต้องมี End If หลาย ๆ ตัวด้วยจึงจะใช้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การวนซ้ำแบบ For ... Next มีวิธีการใช้ดังต่อไปนี้
 For <ตัวแปรที่ใช้วนรอบ> = <ค่าเริ่มต้น> to <ค่าสุดท้าย>
 ...
 Next
- การวนซ้ำแบบตรวจสอบเงื่อนไข While ... Loop มีวิธีการใช้ดังนี้
 While <ตรวจสอบเงื่อนไขการวนซ้ำ>
 ...
 Loop

8.4 ความรู้เบื้องต้นของ ASP

การเขียนภาษา ASP นั้น สามารถทำให้สามารถเขียนเว็บเพจที่มีความซับซ้อนและมีความปลอดภัยสูงกว่า การใช้ HTML อย่างเดียวเพราะว่า ASP นั้นจะรันที่เซิร์ฟเวอร์ไม่ได้รันที่ไคลเอนต์จึงมีความปลอดภัยมากกว่า โดย ต้องเซฟ เป็น .asp

การเขียน ASP script จะทำโดยการฝังหรือ embedded ส่วนที่เป็น script ลงไปในเว็บเพจ กล่าวคือ หากไม่มีการฝัง ASP script เลยเว็บเพจนั่นก็คือเว็บเพจธรรมดาทั่วไปนั่นเอง การตีความโดย ASP ก็จะทำให้การตีความไต่ลงมาจากต้นไฟล์ไปที่ละบรรทัด บรรทัดไหนมีส่วนของ ASP script อยู่ก็จะทำการตีความก่อนแล้วจะทยอยส่งผลลัพธ์ออกมาเรื่อยๆ หากเกิดข้อผิดพลาดที่รุนแรงก็จะหยุดการทำงาน ส่วนที่เป็น ASP script จะขึ้นต้นหรือปิดด้วย tag โดยใช้เครื่องหมาย <% และลงท้ายหรือปิดด้วย %>

ยกตัวอย่างการเขียน ASP script เช่นหากเราต้องการแสดงวัน-เวลาปัจจุบัน ก็จะใช้ ฟังก์ชัน now ก็จะเขียนดังนี้

วัน-เวลา ขณะนี้คือ

```
<%=now%>
```

โดยที่บรรทัดที่ 1 ก็ จะแสดงข้อความตามปกติ แต่บรรทัดที่ 2 จะมีการนำค่า วัน-เวลา ปัจจุบัน ผลลัพธ์ก็จะเป็นดังนี้

วัน-เวลา ขณะนี้คือ 2/2/2004 6:12:18 PM

การจะเขียน ASP script เพื่อทำงานอะไร ก็ต้องศึกษารูปแบบคำสั่งของภาษา VBScript หรือ JScript และต้องมีพื้นฐานของการเขียนภาษา HTML ด้วย เพราะจะต้องใช้ร่วมกัน ดังตัวอย่างที่ผ่านมาหากจะให้มีการแสดงเป็นตัวเข้ม ขนาดโตขึ้นไปจากปกติ 2 ระดับและแสดงข้อความสีเขียว ทั้งหมดที่ว่ามาจะเป็นกำหนดที่ ภาษา HTML ครับ ดังนี้

```
<font size=+2 color=green>วัน-เวลา ขณะนี้คือ
<%=now%></font>
```

ผลลัพธ์คือ

วัน-เวลา ขณะนี้คือ 2/2/2004 6:12:18 PM

โดยในส่วนที่ ติดต่อกับ Data Base นั้นมีดังนี้
เราต้องสร้าง ไฟล์ .udl ขึ้นมาก่อนเพื่อทำการเชื่อมโยงกับ Data Base แล้วจึงใส่คำสั่งพวกนี้ลงใน Code ที่ได้ทำไว้

```
DataBase="File name=" & request.servervariables("APPL_PHYSICAL_PATH") & "connect.udl"
Set Cn=Server.CreateObject("ADODB.Connection")
Cn.Open(DataBase)
```

โดยคำสั่งที่ใช้ในการติดต่อ Data Base มีดังนี้

1. การดึงข้อมูลจาก Data Base ใช้คำสั่งดังนี้
Select <Column ที่ต้องการ> From <ชื่อของตาราง>
2. การใส่ข้อมูลลงใน Data Base ใช้คำสั่งดังนี้
Insert into <ชื่อตาราง> values <ข้อมูลที่ต้องการนำเข้า>
3. การเปลี่ยนแปลงข้อมูลใน Data Base ใช้คำสั่งดังนี้
Update <ชื่อตาราง> SET <ชื่อ Column> = <ข้อมูลที่ต้องการเปลี่ยน>
4. การลบข้อมูลออกจาก Data Base ใช้คำสั่งดังต่อไปนี้
Delete from <ชื่อตาราง>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

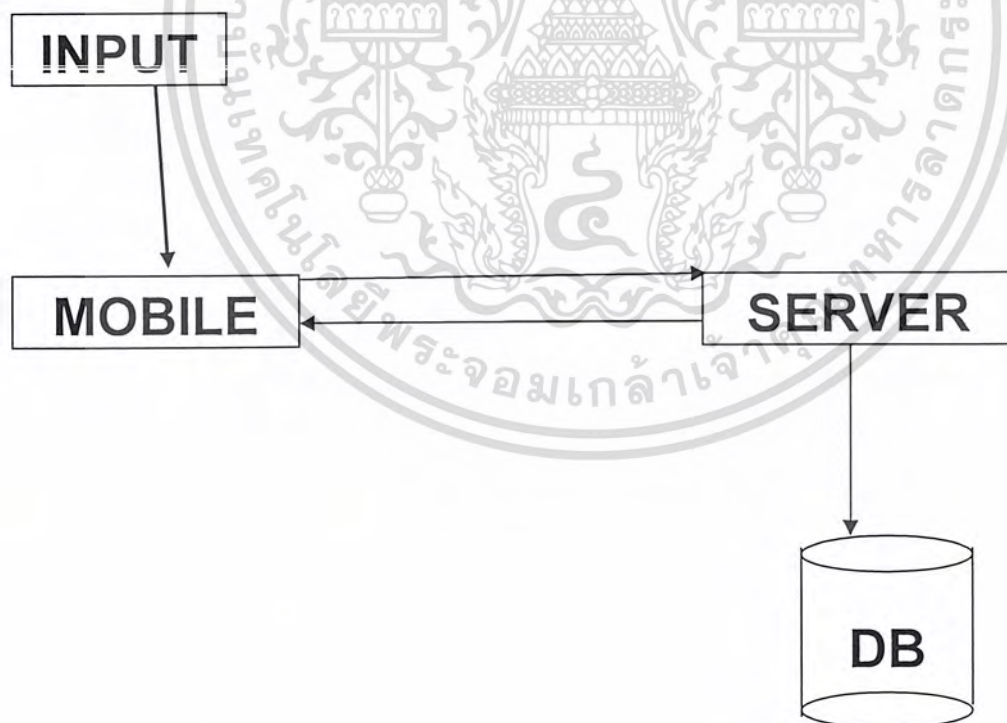
บทที่ 9

การออกแบบเซิร์ฟเวอร์

9.1 การทำงาน

ลักษณะการทำงานในฝั่งเซิร์ฟเวอร์นั้นจะทำงานเพื่อทำการติดต่อกับฐานข้อมูลที่มี เพื่อจะทำการดึงข้อมูลที่ไคลเอนต์ทำการร้องขอเพื่อที่จะทำการโหลดในตอนที่จะเริ่มเล่นเกมส์หรือเซฟในตอนที่จะเล่นเกมส์เพื่อที่จะสามารถนำมาเล่นในคราวต่อไป เนื่องจากเกมส์เป็นแนว RPG คือลักษณะเกมเป็นเนื้อเรื่องที่ผู้เล่นต้องดำเนินตามเนื้อเรื่องแต่สามารถดำเนินไปในลักษณะที่แตกต่างกันออกไปได้ในแต่ละตัวละครหรือในบางส่วนของการเล่นเนื้อเรื่อง ดังนั้นการที่เกมส์มีลักษณะดังนี้จะต้องมีการดำเนินเนื้อเรื่องที่ยาวพอสมควรจึงจำเป็นต้องมีการเซฟเกมส์ไว้เพื่อหยุดพักแล้วสามารถที่จะมาเล่นต่อจากของเดิมเมื่อไรก็ได้ อีกทั้งยังสามารถทำการโหลดหรือเซฟข้อมูลในการที่จะเล่นเกมส์จากโทรศัพท์มือถือใดก็ได้ที่มีแอปพลิเคชันของเกมส์เดียวกันนี้ โดยเพียงแค่ติดต่อไปยังเซิร์ฟเวอร์ที่ต้องการให้ถูกต้อง

การทำงานของเซิร์ฟเวอร์จะทำงานในรูปแบบของเว็บเซิร์ฟเวอร์ที่เขียนด้วยภาษา ASP (Active Sever Page) ซึ่งภาษาที่รันที่ฝั่งเซิร์ฟเวอร์ การที่ไคลเอนต์จะทำการติดต่อมาที่เซิร์ฟเวอร์นั้นจะต้องกระทำผ่านโปรโตคอล HTTP เพื่อที่จะติดต่อผ่านอินเทอร์เน็ต ไปยังเซิร์ฟเวอร์ที่เราต้องการ



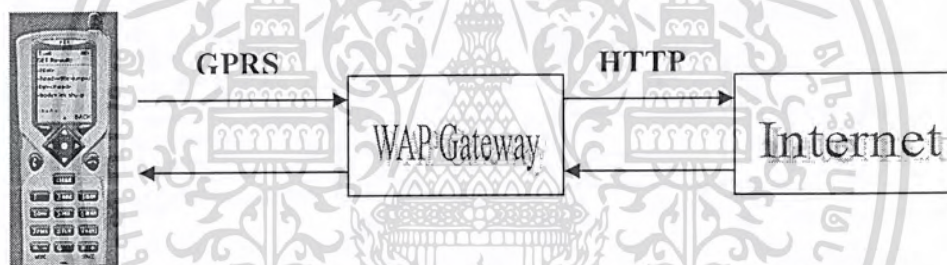
รูปที่ 9.1 การติดต่อจากไคลเอนต์ไปยังเซิร์ฟเวอร์เพื่อที่จะโหลดและเซฟเกมส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะเห็นได้ว่าลักษณะการทำงานจะเริ่มขึ้นโดยการที่ผู้ใช้จะทำการเล่นเกมสับบนโทรศัพท์มือถือนั้นก็คือไคล์แอนดนี่เองโดยการป้อน INPUT (การกดปุ่มต่างๆบนโทรศัพท์มือถือ) แล้วทางไคล์แอนดนี่ก็จะทำการติดต่อผ่านโปรโตคอล HTTP มายังเซิร์ฟเวอร์เพื่อร้องขอข้อมูลต่างๆมายังฐานข้อมูลหรือทำการนำข้อมูลที่ต้องการเซฟมาเก็บไว้ที่ฐานข้อมูล

9.2 ลักษณะของการติดต่อ

การติดต่อนั้นเริ่มจากโทรศัพท์มือถือจะกระทำการติดต่อผ่านโปรโตคอล GPRS ในเครือข่ายไร้สายไปยัง wap gateway ของผู้ให้บริการขอโทรศัพท์นั้นเพื่อที่จะให้สามารถเข้าไปยังเครือข่ายอินเทอร์เน็ตเมื่อผ่านเข้าไปยังเครือข่ายอินเทอร์เน็ตแล้วจากใช้โปรโตคอล HTTP เพื่อที่จะทำการติดต่อร้องขอข้อมูลจากเว็บเซิร์ฟเวอร์ในรูปแบบของการเรียกหน้าเว็บเพจตามปกติซึ่งคำสั่งและรูปแบบของการทำงานต่างๆที่โทรศัพท์มือถือซึ่งเขียนด้วยภาษา J2ME จะกล่าวโดยละเอียดอีกทีในการออกแบบแอปพลิเคชันในฝั่งไคล์แอนดนี่



รูปที่ 9.2 การติดต่อจากเครือข่ายไร้สายไปยังเครือข่ายอินเทอร์เน็ต

9.3 การออกแบบฐานข้อมูล

ระบบฐานข้อมูลที่ใช้เก็บข้อมูลต่างๆคือ Microsoft Access เนื่องจากข้อมูลที่จะนำมาเก็บนั้นมีลักษณะเป็นรายละเอียดต่างๆของตัวละครดังนั้นฐานข้อมูลก็จะถูกออกแบบให้มีฟิลด์ต่างๆที่เก็บข้อมูลรายละเอียดของตัวละครนั้นๆว่าเป็นของใคร ซึ่งตัวละครของผู้เล่นแต่ละคนก็จำเป็นจะต้องป้อนความลับเวลาที่ผู้เล่นจะเล่นเกมคือด้วยตัวละครของตัวเองจึงจำเป็นจะต้องใส่ ID กับ PASSWORD เพื่อเป็นการบ่งบอกว่าตัวละครของใครเป็นของใครถ้าไม่ใช่ของตัวเองคนอื่นจะไม่สามารถนำออกมาเล่นได้ จากรูป 9.3 จะเห็นได้ว่าตารางของฐานข้อมูลจะมีแค่ตารางเดียวเนื่องมาจากข้อมูลของแต่ละตารางเป็นเอกเทศกันไม่เกี่ยวข้องกันเลยจึงไม่มีความซ้ำซ้อนเกิดขึ้นดังนั้นจึงมีแค่ตารางเดียวก็พอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	real_id	cus_id	cus_password	cus_name	lv	human_hp	expna	red	white	m	l	k	cha	moneyna	job
/	121	Suppachai	1234	Thom	1	100	100	3	3	2	13	15	1	25000	1
*	(AutoNumber)				0	0	0	0	0	0	0	0	0	0	0

รูปที่ 9.3 ฟิลด์ต่างๆที่ใช้ในการเก็บข้อมูล

รายละเอียดของฟิลด์ต่างๆ

Real_id	หมายถึง ID ของตัวละครแต่ละตัวที่เก็บไว้ (Primary Key)
Cus_id	หมายถึง ID ของผู้เล่น
Cus_password	หมายถึง PASSWORD ของผู้เล่น
Cus_name	หมายถึง ชื่อตัวละครของผู้เล่น
Lv	หมายถึง ระดับของตัวละครมีตั้งแต่ระดับ 1 ถึง 99
Human_hp	หมายถึง พลังชีวิตของตัวละคร
Expna	หมายถึง ค่าประสบการณ์ในการต่อสู้ของตัวละคร
Red	หมายถึง ของที่ใช้ในการใช้ท่าพิเศษของแต่ละตัวละคร
White	หมายถึง ของที่ใช้ในการเพิ่มพลังชีวิตของตัวละคร
M	หมายถึง MAP ที่ตัวละครอยู่
L	หมายถึง ตำแหน่งในแนวแกน y ของตัวละครใน MAP นั้นๆ
K	หมายถึง ตำแหน่งในแนวแกน x ของตัวละครใน MAP นั้นๆ
Cha	หมายถึง บทบาทตัวละครที่เราทำการเลือกเล่น -“0” คือ พระอภัยมณี -“1” คือ ศรีสุวรรณ -“2” คือ แก้วเกสา -“3” คือ นางเงือก
Money	หมายถึง จำนวนเงินที่ตัวละครมีอยู่
Job	หมายถึง ตัวละครมีความสามารถในระดับใด -“1” คือ ไม่สามารถใช้ท่าพิเศษได้ -“2” คือ สามารถใช้ท่าพิเศษได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.4 การออกแบบหน้าเว็บเพจ

เนื่องจากการติดต่อของไคลเอนต์ที่ติดต่อมานั้นเป็นลักษณะของ HTTP connection ซึ่งเป็นการที่เรียกหน้าเว็บเพจขึ้นมา ดังนั้นการทำงานของเซิร์ฟเวอร์เช่น การคำนวณต่างๆ การติดต่อฐานข้อมูลนั้น จึงต้องอาศัยหน้าเว็บเพจแต่ละหน้าเพื่อทำงานในลักษณะที่แตกต่างกันออกไปเพื่อความเป็นระเบียบและเข้าใจง่าย ซึ่งสิ่งที่ต้องคำนึงถึงในการออกแบบนั้นก็คือมีการทำงานอะไรอยู่บ้างสามารถแบ่งการทำงานออกเป็นหลายๆย่อยแต่ละส่วนได้อย่างไรแล้วนำการทำงานเหล่านั้นไปใส่บนหน้าเว็บเพจแต่ละหน้าซึ่งการทำงานนั้นสามารถเลือกใช้ภาษาอะไรก็ได้ตามความถนัดแล้วเหมาะสมซึ่งในที่นี้เลือกใช้ภาษา ASP ในที่นี้การทำงานของส่วน CODE ที่ทำการติดต่อ DATABASE คือ

```
DataBase="File name=" & request.servervariables("APPL_PHYSICAL_PATH") & "connect.udl"
set Cn = Server.CreateObject("ADODB.Connection")
```

ซึ่งเป็นคำสั่งที่ใช้ติดต่อไปยังไฟล์ (.UDL) ซึ่งไฟล์ (.UDL) ก็คือไฟล์ที่ทำการบรรยายรายละเอียดในการเชื่อมต่อกับฐานข้อมูลนั้นๆซึ่งเราก็สามารถนำไฟล์ (.UDL) นี้มาเรียกใช้เมื่อเว็บเพจต้องการเข้าถึงข้อมูล

9.5 การทำงานของเว็บเพจแต่ละหน้า

9.5.1 NEWID.ASP

หน้านี้จะทำงานในส่วนของการที่ผู้เล่นต้องการจะสมัครเพื่อที่จะเล่นตัวละครใหม่ซึ่งการทำงานของหน้านี้ก็คือ เมื่อผู้เล่นเปิดเกมขึ้นมาเพื่อที่จะทำการเล่นนั้นแล้วกดเลือก NEW ทางไคลเอนต์จะทำการให้ป้อน ID และ PASSWORD เมื่อผู้เล่นทำการป้อนแล้วกดตกลงทางไคลเอนต์จะทำการส่ง ID มาเซิร์ว่าเคยมีคนใช้ ID นี้แล้วรึยังถ้ามีคนใช้แล้วก็จะฟ้องขึ้นมาว่ามีคนเคยใช้แล้วการสมัครก็จะไม่สมบูรณ์ต้องเปลี่ยน ID จนเมื่อใส่ ID ไม่ซ้ำกับคนอื่นที่เคยมีแล้วทางเซิร์ฟเวอร์จึงจะทำการนำ ID และ PASSWORD ไปเก็บในฐานข้อมูลโดยการเพิ่มตัวละครใหม่ให้หลังจากนำข้อมูลไปเก็บแล้วไคลเอนต์ก็จะได้หมายเลขของตัวละครจากหน้าเว็บเพจเพื่อที่จะนำไปใช้ในการที่จะเซฟข้อมูลหลังจากเลิกเล่นเกม

ตัวอย่าง CODE ของหน้า NEWID.ASP

```
<%
```

```
dim id
dim pass
id = request("id")
pass = request("pass")
inid = replace(id,"","'")
inpass = replace(pass,"","'")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DataBase="File name=" & request.servervariables("APPL_PHYSICAL_PATH") &
"connect.udl"
set Cn = Server.CreateObject("ADODB.Connection")
Cn.Open(DataBase)
sql = "select * from idtable where cus_id = "& inid &""
set rs = Cn.execute(sql)
if not rs.eof then
    'do nothing
else
    sql = "insert into idtable (cus_id,cus_password) values (" & inid & "," &
inpass & "" )"
    Cn.execute(sql)
    sql = "select * from idtable where cus_id = "& inid &""
    set rs = Cn.execute(sql)
    if not(rs.eof) then
        realid = rs("real_id")
        response.write(realid)
    end if
end if
rs.close
Cn.close
%>

```

9.5.2 NEWNAMETYPE.ASP

ในส่วนของหน้านี้นั้นเมื่อผู้เล่นทำการสมัครเสร็จแล้วจะต้องเลือกตัวละครว่าจะเป็นตัวใด หลังจากนั้นก็ทำการใส่ชื่อตัวละครเข้าไปพอดตกลงก็จะทำการส่งข้อมูลไปยังหน้านี้หน้านี้ก็จะทำการนำข้อมูลต่างๆที่เกี่ยวกับตัวละคร โดยส่งหมายเลขตัวละครมาด้วยว่าจะให้นำข้อมูลไปใส่ยังตัวละครใดที่ฐานข้อมูล

ตัวอย่าง CODE ของหน้า NEWNAMETYPE.ASP

```

<%
    inid = request("id")
    name = request("name")
    typeid = request("char")
    level = request("level")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

human_hp = request("human_hp")
inexp = request("exp")
red = request("red")
white = request("white")
m = request("m")
l = request("l")
k = request("k")
money = request("money")
job = request("job")
inname = replace(name,"","")
DataBase="File name=" & request.servervariables("APPL_PHYSICAL_PATH") &
"connect.udl"
set Cn = Server.CreateObject("ADODB.connection")
Cn.Open(DataBase)
sql = "UPDATE idtable SET cus_name=" & inname & ",cha=" & typeid &
",human_hp=" & human_hp & ",lv=" & level & ",expna=" & inexp & ",red=" & red
& ",white=" & white & ",m=" & m & ",k=" & k & ",l=" & l & ",moneyna=" & money
& ",job=" & job & " Where real_id=" & inid
Cn.execute(sql)
Cn.close
%>

```

9.5.3 CHKCON.ASP

เมื่อผู้เล่นต้องการจะเล่นตัวละครที่สมัครไว้แล้วต่อจากของเดิมเมื่อเปิดเกมมาให้เลือกที่ CONTINUE แล้วตัวเกมจะจะให้ป้อน ID กับ PASSWORD ของตัวละครนั้นๆถ้าเกิดใส่ ID หรือ PASSWORD ผิดก็จะเป็นการดึงข้อมูลต่างๆของ ID และ PASSWORD ของตัวละครนั้นๆ

ตัวอย่าง CODE ของหน้า CHK.ASP

```

<%
dim id
dim pass
id = request("id")
pass = request("pass")
inid = replace(id,"","")
inpass = replace(pass,"","")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DataBase="File name=" & request.servervariables("APPL_PHYSICAL_PATH") &
"connect.udl"

set Cn = Server.CreateObject("ADODB.connection")

Cn.Open(DataBase)

sql = "select * from idtable where cus_id = "& inid &" and cus_password = " &
inpass & ""

set rs = Cn.execute(sql)

if not rs.eof then

response.write(rs("cus_name")&" "&rs("lv")&" "&rs("m")&" "&rs("l")&"
"&rs("k")&" "&rs("red")&" "&rs("white")&" "&rs("expna")&" "&rs("human_hp")&"
"&rs("moneyna")&" "&rs("cha")&" "&rs("job")&" "&rs("real_id"))

else

'do nothing

end if

rs.close

Cn.close

%>

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 10

การออกแบบไคล์แอนต์

10.1 การออกแบบลักษณะเกมส์

เริ่มออกแบบโดยการคิดแนวเกม มุมมองที่ผู้เล่นจะเห็นการดำเนินเนื้อเรื่องในเกม ข้อจำกัดต่างๆของเกมก่อนแล้วค่อยเลือกเนื้อหาของการดำเนินเนื้อเรื่องในเกม ซึ่งในที่นี้ออกแบบให้เกมส์มีลักษณะเป็นเกมส์แนว RPG โดยที่ให้มุมมองของผู้เล่นนั้นมองเห็นแบบ 2 มิติ โดยจากข้อจำกัดต่างๆเหล่านี้สามารถเลือกเนื้อหาที่เข้ามาเกี่ยวข้องกับ เรื่องพระอภัยมณีที่เลือกเรื่องนี้นั้นเนื่องจากเป็นวรรณคดีไทยซึ่งการนำมาทำเกมส์นั้นก็จะเป็นการเป็นการเผยแพร่วัฒนธรรมไทยไปด้วยเนื่องจากผู้เล่นจำเป็นจะต้องรู้เนื้อเรื่องโดยย่อของวรรณคดีไทยในการที่จะสามารถพัฒนาความสามารถตัวละครของผู้เล่นตามเนื้อเรื่องที่วางไว้ อีกทั้งการดำเนินเนื้อเรื่องยังมีชื่อตัวละครโผล่มาให้เห็นในการต่อสู้ต่างๆจึงเป็นการทำให้คนที่ยังไม่รู้ว่าเนื้อเรื่องเป็นอย่างไรเกิดความสงสัยที่จะศึกษาวรรณคดีไทยขึ้นมาบ้างจากมุมมองต่างๆ

10.1.1 รูปแบบของเกมที่ออกแบบ

1. มีการสมัครเล่นเกมส์โดยการใส่ ID,PASSWORD เพื่อสมัครซึ่ง ID จะต้องไม่ซ้ำกับคนอื่นที่เคยมาสมัครไว้แล้วก่อนหน้านี้
2. สามารถเลือกตัวละครได้หลายตัวเพื่อที่ความต้องการจะสวมบทบาทในการที่จะดำเนินเนื้อเรื่องของเกมส์ในลักษณะที่แตกต่างกันไปในแต่ละตัวละครในบางส่วนของเท่านั้นเพราะการที่ดำเนินเรื่องหลักทุกตัวจะต้องดำเนินเหมือนกัน ตัวละครดังกล่าวที่ว่านี้เช่น พระอภัยมณี ศรีสุวรรณ แก้วเกสา นางเงือก เป็นต้น
3. เมื่อเลือกบทบาทที่ตัวละครต้องการเสร็จแล้วหลังจากนั้นก็ตั้งชื่อให้กับตัวละครได้
4. ตัวละครแต่ละตัวมีความสามารถที่ไม่เท่ากันในระดับเลเวลที่เท่ากันคือ
 - พระอภัยมณีสามารถใช้ทำพิเศษได้แรงกว่าตัวละครอื่นในเลเวลเดียวกัน
 - ศรีสุวรรณสามารถใช้ทำธรรมดาได้แรงกว่าตัวละครอื่นในเลเวลเดียวกัน
 - แก้วเกสาสามารถซื้อของได้ถูกกว่าตัวละครอื่น
 - นางเงือกเมื่อใช้ทำพิเศษแล้วจะทำให้เพิ่มพลังชีวิตอีกด้วย
5. การดำเนินเนื้อเรื่องไปในระยะหนึ่งเมื่อเลเวลถึงข้อกำหนดแล้วในที่นี้คือเลเวล 20 จะสามารถไปลุยตามเนื้อเรื่องแล้วทำให้มีความสามารถพิเศษเพิ่มขึ้น
6. ของมีอยู่ 2 ชนิดคือน้ำยาเพิ่มพลังและน้ำยาที่ใช้ทำพิเศษสิ่งของ 2 อย่างนี้สามารถซื้อได้ด้วยเงิน โดยที่เงินนั้นหาได้มาจากการฆ่าศัตรู
7. การดำเนินเนื้อเรื่องจะมีความเก่งของศัตรูในแต่ละแผนที่ไม่เท่ากันซึ่งแผนที่ที่มีตัวละครที่เก่งก็จะได้เงินและค่าประสบการณ์มากด้วยเช่นกัน
8. การเกิดของศัตรูนั้นเกิดจากการสุ่มว่าจะเกิดที่พิกัดใดของแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. การเดินบนแผนที่นั้นเราไม่สามารถจะรู้ได้ว่าศัตรูนั้นอยู่ที่พิกัดใดของแผนที่ซึ่งถ้าเราเดินไปยังบริเวณที่มีศัตรูอยู่นั้นเราก็จะถูกย้ายฉากไปเป็นฉากต่อสู้เมื่อสู้เสร็จแล้วเราจะได้อุปกรณ์และเงิน
10. การซื้อของจะต้องไปซื้อของบ้านที่มี NPC ที่ขายของยืนอยู่แล้วก็ทำการคุยกับ NPC เพื่อที่จะซื้อของที่ต้องการ
11. การออกจากเกมนั้นทางไคล์แอนต์จะทำการเซฟให้โดยอัตโนมัติ

10.2 การทำงานของคลาส ต่างๆ

10.2.1 Prapaimanee

ในส่วนนี้จะมีการทำงานในส่วนของคลาสหลักนั่นก็คือคลาส Prapaimanee ซึ่งทำงานในส่วนของ MIDlet ทั้งหมด ที่จะทำงานว่าต้องการจะให้ปัจจุบันบนหน้าจอโทรศัพท์มือถือนั้นจะโชว์หน้าจอในรูปแบบ FORM หรือ CANVAS หน้าใด

```
public class Prapaimanee extends MIDlet implements CommandListener,ItemStateListener
{
    .....
}
```

10.2.2 Mycanvas

การทำงานของคลาสนี้จะทำงานด้านเกี่ยวกับการวาดภาพทั้งหมดในตัวเกมส์หลังจากการเข้าฉากเล่นเกมเรียบร้อยแล้ว คือหลังจากการใส่ ID, PASSWORD ที่ถูกต้องแล้ว รวมการทำงานของ การติดต่อ บนปุ่มกดต่างๆของผู้เล่นที่กดจาก โทรศัพท์มือถือ ซึ่งการทำงานของคลาสนี้จะใช้ LOW LEVEL คือการจัดการด้านกราฟฟิกและการติดต่อจากผู้ใช้ด้วยการเขียน โปรแกรมจัดการสิ่งต่างๆเองทั้งหมด โดยผู้พัฒนาโปรแกรม

```
class MyCanvas extends Canvas
{
    .....
}
```

10.2.3 TestTimer

จะทำงานในส่วนของการจัดการด้านเวลาที่จะทำการแสดงผลในการวาดภาพในฉากการต่อสู้ หรือจัดการกับค่าตัวเลขต่างๆที่ใช้ในเกม ยังรวมไปถึงการจัดการด้านช่วงเวลาในการที่จะหยุดของของบางตัวละครเช่นการหยุดของหัวหน้าก่อนเข้าฉากต่อสู้สุดท้าย

```
private class TestTimerTask extends TimerTask
{
    .....
}
```

10.2.4 Apaimanee, Fish, Seesuwun, Kaewkaesa

การทำงานของคลาสเหล่านี้เพื่อใช้กำหนดความแตกต่างของรูปภาพแต่ละแอกชั่นของแต่ละตัวละครสร้างเพื่อให้เกิดความเป็นระเบียบและง่ายต่อการเรียกใช้ ซึ่งอาจมีประโชน์ที่สามารถใช้ได้รายละเอียดของตัวละครเมื่อเกิดการพัฒนาเกมสัจงมีขนาดใหญ่และมีความซับซ้อนมากขึ้น

```
public class Apaimanee
```

```
{ ..... }
```

```
public class Seesuwun
```

```
{ ..... }
```

```
public class Kaewkaesa
```

```
{ ..... }
```

```
public class Fish
```

```
{ ..... }
```

10.2.5 Attack

ทำการเก็บรูปภาพและรายละเอียดของท่าทีแต่ละตัวละครใช้โจมตีศัตรูรูปภาพที่เก็บจะเป็นภาพที่เป็นภาพของการเคลื่อนไหวที่ละนิดจนเมื่อนำมาแสดงด้วยคีย์แล้วสามารถเกิดเป็นแอนิเมชันได้

```
public class Attack
```

```
{ ..... }
```

10.2.6 Background

จะเห็นว่าคลาสนี้จะทำการเก็บรูปภาพต่างที่เป็นภาพพื้นหลังในฉากต่อสู้โดยการแยกคลาสนี้ออกมาเก็บภาพเพื่อเป็นการแยกส่วนของการโหลดภาพเพื่อที่ให้ลดหน่วยความจำให้ลดน้อยลงในการเก็บภาพของเกมสัจงไว้เพื่อที่จะแสดงผล

```
Public class Background
```

```
{ ..... }
```

10.2.7 Monster

คลาสนี้จะทำการเก็บรายละเอียดของศัตรูทั้งหมดที่มีรวมทั้งหัวหน้าด้วยเช่นกัน รายละเอียดของศัตรูที่มีคือพลังชีวิต ความเร็ว เงินที่ได้และค่าประสบการณ์ที่ได้เมื่อนำศัตรูตัวนั้นตาย ค่าการโจมตีคลาสนี้ยังทำงานในการคำนวณพลังชีวิตของศัตรูตัวนั้นที่เกิดจากการโจมตีของตัวละครต่างๆ

```
public class Monsters
```

```
{ ..... }
```

10.2.8 MapGame

เป็นคลาสที่ทำหน้าที่ในการเก็บรายละเอียดของแต่ละแผนที่ว่ามีรายละเอียดของแผนที่เป็นอย่างไรรายละเอียดที่ว่ามีหมายถึงลักษณะของแต่ละพิกัดว่าจะให้แสดงภาพใดออกไปทางหน้าจอโทรศัพท์นั่นเอง

```
class MapGame
{
    .....
}
```

10.2.9 Random

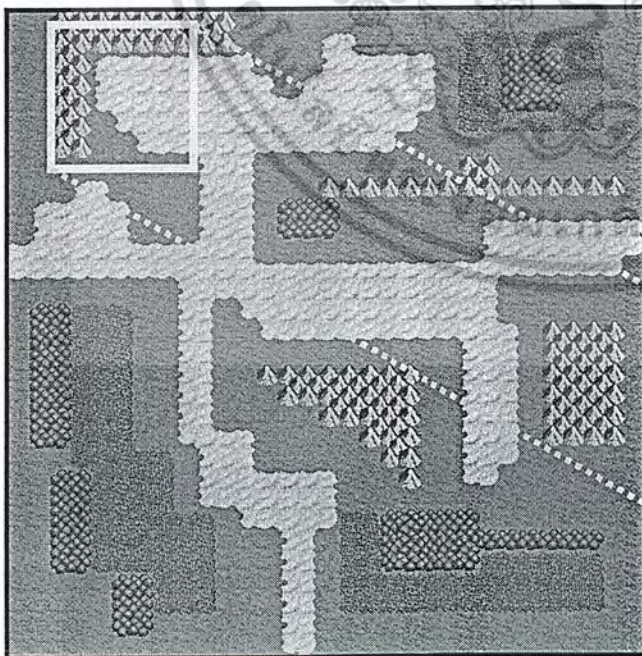
เป็นคลาสที่ทำหน้าที่ในการที่จะสุ่มตำแหน่งของศัตรูต่างๆขึ้นมาบนแต่ละแผนที่โดยที่แต่ละแผนที่นั้นก็จะมีศัตรูที่แตกต่างกันออกไป และแต่ละแผนที่ก็ยังมีจำนวนความหนาแน่นของศัตรูแต่ละชนิดไม่เท่ากันอีกโดยถ้าต้องการกำหนดความหนาแน่นของศัตรูของแต่ละแผนที่ก็สามารถมากำหนดที่คลาสนี้เท่านั้นก็เป็นอันเสร็จ

```
Public class random
{
    .....
}
```

10.3 เทคนิคต่างๆที่นำมาใช้ในเกม

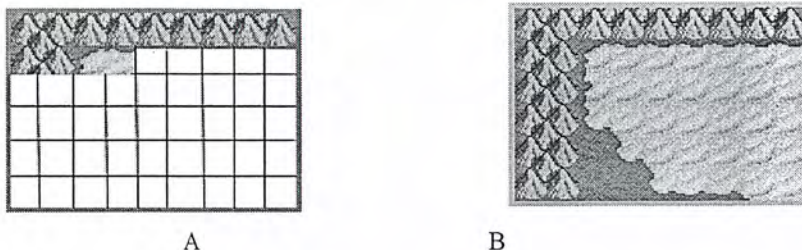
10.3.1 เทคนิคการวาดแผนที่

เนื่องการเป็นเกม 2 มิติซึ่งการดำเนินเนื้อเรื่องของของเกมนั้นต้องมีการเดินในแผนที่ซึ่งแผนที่นั้นมีความกว้างและความยาวมากเราจึงต้องทำการแบ่งการสร้างภาพออกเป็นส่วนย่อยในการที่สร้างแผนที่ในบนหน้าจอส่วนที่หน้าจอมองเห็นได้



รูปที่ 10.1 ภาพแผนที่ใหญ่เปรียบเทียบกับส่วนที่หน้าจอมองเห็นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 10.2 A:วิธีการวาดในหนึ่งหน้าจอ B:ภาพหลังจากการวาด

จากรูปที่ 10.2 ซึ่งแสดงเทคนิคการวาดรูปแผนที่จะเห็นได้ว่าการวาดภาพในหนึ่งหน้าจอั้นจะทำการวาดทีละ block โดยที่แต่ละ block มีขนาดเท่าซึ่งมีลักษณะคล้ายตารางโดยการวาดจะกระทำทีละรูป โดยวาดไล่จากซ้ายไปขวาเมื่อวาดจนถึงสุดขอบจอด้านซ้ายก็จะทำการวาดถัดลงมาอีกแถว ประโยชน์จากการวาดแผนที่ด้วยวิธีนี้ก็คือการประหยัดรูปที่จะต้องใช้นื่องมาจากในบางแผนที่นั้นจะมีพื้นที่ซ้ำกันเป็นจำนวนมากแล้วถ้าเรานำส่วนนั้นมาแสดงทีละรูปก็จะมีขนาดใหญ่มาก ตัวอย่างเช่น จากรูป 10.1 นั้นมีส่วนที่เป็นพื้นดินนั้นเยอะมากดังนั้นภาพพื้นดินที่ซ้ำกันนั้นเรานำรูปภาพมาเก็บเพียงแค่ส่วนเดียวแล้วมาแสดงผลจากลักษณะการวาดดังที่อธิบายไป

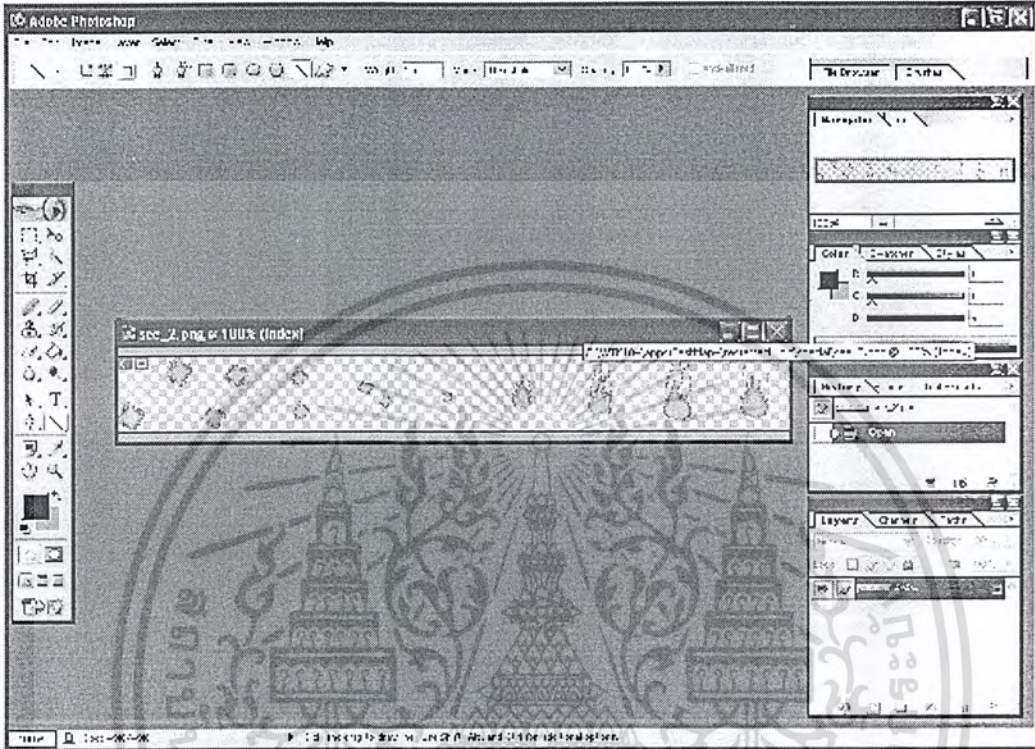
ตัวอย่าง CODE ของการวาดแผนที่

```
for(j=0;j<=6;j++)
{
for(i=0;i<=8;i++)
{
g.drawImage(arrayP0[map.M0[j+1][i+k]],i*20,j*17+25,0);
}
}
```

จะเห็นได้ว่าจาก code นั้นใช้อาร์เรย์ 2 มิติเพื่อที่จะทำการวาดรูปในหนึ่งหน้าจอ โดยการวาดรูปนั้นใช้ใช้คำสั่ง g.drawImage() โดยที่ฟังก์ชันนี้สามารถกำหนดได้ว่าเราจะวาดภาพชื่ออะไรและวาดในตำแหน่งไหนของหน้าจอ โดยที่ code ในส่วนนี้สามารถที่จะดูรายละเอียดเพิ่มเติมได้ที่คลาส MyCanvas

10.3.2 เทคนิคการทำภาพแอนิเมชันในการใช้ท่า

ในส่วนของการเล่นเกมนั้นฉากต่อสู้จะมีการใช้ animation ในการใช้ท่าพิเศษของตัวละครต่างๆ ซึ่งท่าพิเศษของแต่ละตัวก็จะไม่เหมือนกันในส่วนนี้จะใช้การสร้างภาพขึ้นมากให้ต่อเนื่องกันเป็นเฟรมๆ แล้วทำการวาดทีละเฟรมโดยใช้ timer เป็นตัวจับเวลา



รูปที่ 10.3 ภาพในแต่ละเฟรม



รูปที่ 10.4 แอนิเมชันที่ปรากฏในเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง CODE ของการทำแอนิเมชั่น

```

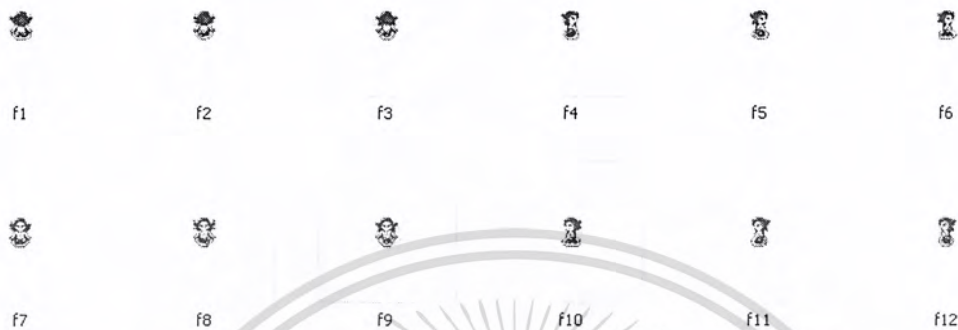
if(this.prapaimanee.Char == 3)//-----this.prapaimanee.Char = 3
{
    g.setColor(255,255,0);
    temp_string = Integer.toString(this.prapaimanee.level*5);
    g.drawString(temp_string,8,33,0);
    g.setColor(0,0,255);
    if(special_time == 2)
    {
        g.drawImage(attack.special[2],50,40,0);
        g.drawString(human_damage,80,100,0);//-----damage
    }
    else
    if(special_time == 3)
    {
        g.drawImage(attack.special[2],-25,40,0);
    }
    else
    if(special_time == 4)
    {
        g.drawImage(attack.special[2],-100,40,0);
    }
}

```

จากตัวอย่างของ code จะมีการกำหนดว่าถ้าช่วงเวลาเท่าไรจะให้ทำการแสดงภาพไหนซึ่งตัวที่จะกำหนดช่วงเวลาทั้งหมดนั้นก็คือในส่วนของ Timer นั่นเองแต่การกำหนดช่วงเวลานั้นไม่ได้มีแค่ Timer เท่านั้นที่จะใช้ได้ยังสามารถใช้ความสามารถของ Thread เพื่อที่จะกำหนดคิเลขของเวลา และ 2 วิธีที่กล่าวมานี้ยังสามารถใช้ประโยชน์ในส่วนของการที่ต้องการจะให้มีการทำงานของหลายๆส่วนพร้อมกันเช่นการที่จะทำให้ลูกบอลนั้นโค้งไปโค้งมาในหน้าจอหลายๆลูกในเวลาเดียวกัน เป็นต้น

10.3.3 เทคนิคการแสดงผลการเคลื่อนไหวของตัวละคร

เนื่องจากตัวละครในเกมสแน็กจะมีบทบาทในตอนที่ได้เดินอยู่ในแผนที่ซึ่งมีการขยับไปมา หันซ้าย หันขวาในการเดินก็จำเป็นต้องมีการก้าวขา ดังนั้นจะต้องมีการนำรูปต่างๆเพื่อมาทำให้เกิดท่าทางเหล่านี้เพื่อเพิ่มความสวยงามและความน่าเล่นของเกม



รูปที่ 10.5 ท่าทางต่างๆของตัวละครชื่อนางเงือก

ตัวอย่าง CODE การวาดท่าทางการเคลื่อนไหวของตัวละคร

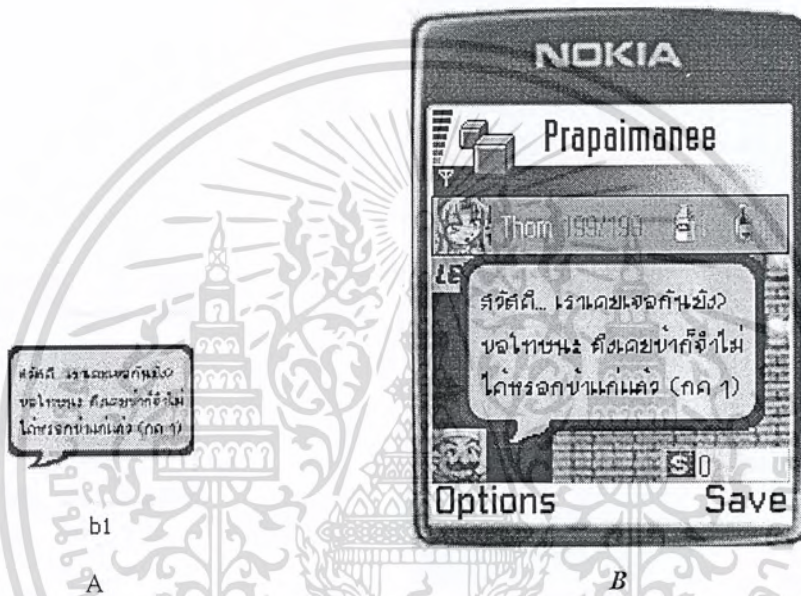
```
public void draw(Graphics g,int hun,int cb,int cf,int cl,int cr)
{
    switch(hun)
    {
        case 0 : { if(cb==0){ g.drawImage(Fi[0],79,67,0);}
                 else {g.drawImage(Fi[2],79,67,0);}
                 break;}
        case 1 : {if(cf==0){g.drawImage(Fi[6],79,67,0);}
                 else {g.drawImage(Fi[8],79,67,0);}
                 break;}
        case 2 : {if(cl==0){g.drawImage(Fi[9],79,67,0);}
                 else {g.drawImage(Fi[11],79,67,0);}
                 break;}
        case 3 : {if(cr==0){g.drawImage(Fi[3],79,67,0);}
                 else {g.drawImage(Fi[5],79,67,0);}
                 break;}
    }
}
```

จากภาพในลักษณะนี้จำเป็นต้องใช้การติดต่อจากผู้เล่นภายนอกด้วยนี่ก็คือใช้ฟังก์ชัน keyPressed() มาช่วยเพื่อที่จะให้กำหนดว่าตอนไหนจะให้เกิดท่าทางแบบใดเช่นเมื่อติดต่อจากการกดปุ่มให้ตัวละครไปด้านซ้ายเพราะฉะนั้นรูปของตัวละครก็ต้องเป็นรูปที่หันด้านซ้ายเป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.3.4 เทคนิคในการสนทนาซื้อของกับพ่อค้าในเกมส์

เนื่องจากในเกมส้นั้นจำเป็นที่จะต้องใช้ของพิเศษโดยมาจากการซื้อของเพื่อเพิ่มความสามารถ เพื่อให้การเล่นเกมส์สามารถดำเนินไปได้โดยสะดวก ไม่ง่ายหรือยากจนเกินไปและการมีเงื่อนไขในการที่จะได้ของมานั้นต้องไปฆ่าศัตรูเพื่อที่จะเก็บตังค์แล้วนำมาซื้อของก็เป็นสิ่งที่เพิ่มความน่าสนใจในตัวเกมส์มากขึ้นอีกด้วย ซึ่งเทคนิคที่จะใช้ในการสนทนานั้นเป็นการติดต่อพูดคุยจึงใช้ภาษาไทยในการพูดคุย แต่ตัวอิมูเลเตอร์ที่ทำการพัฒนาโปรแกรมนี้ไม่สนับสนุนการใช้ภาษาไทยจึงใช้ภาพแทนเพื่อความง่ายและสะดวกในการติดต่อพูดคุยกับผู้เล่น



รูปที่ 10.6 A: ภาพการพูดคุย B: ภาพที่เห็นในเกม

ตัวอย่าง CODE ในการแสดงเงื่อนไขของการซื้อของ

```

if((keyCode==49)&&(prapaimanee.money>=500))
{
    prapaimanee.money=prapaimanee.money-500;
    prapaimanee.temp_red=prapaimanee.temp_red+1;
    talknpc=5;
    repaint();
}

```

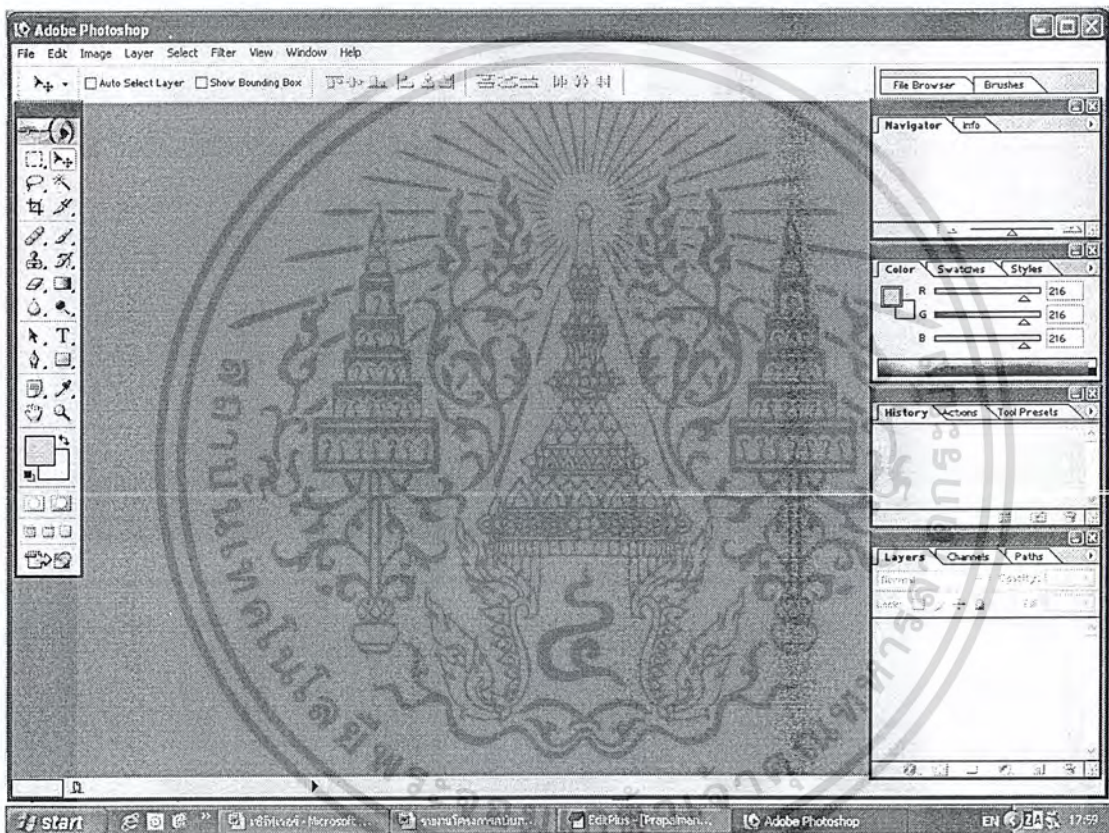
การทำงานของ CODE ในส่วนนี้นั้นจะทำการตรวจสอบว่ามีการกดคีย์ที่มีรหัส 49 ริปแล้วถ้ามีการกดแสดงว่าต้องการซื้อของสิ่งนี้เมื่อมีการกดแล้วก็จะทำการตรวจสอบด้วยว่ามีจำนวนพอร์ิปแล้วถ้ามีจำนวนเงินพอก็ทำการเพิ่มของให้และทำการลดเงินออกไปตามราคาที่กำหนดไว้หลังจากนั้นก็ให้แสดงภาพพูดคุยกับผู้เล่นให้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.4 โปรแกรมที่ใช้ในการพัฒนา

10.4.1 ADOBE PHOTOSHOP

เป็นโปรแกรมที่ใช้ในการออกแบบและตัดต่อ ตกแต่งภาพต่างๆ ในเกมส์เช่น ภาพฉากการต่อสู้ ภาพตัวละคร ภาพท่าต่างๆ ของตัวละคร ภาพศัตรูต่างๆ ภาพเข้าเกม รวมทั้งการออกแบบแอนิเมชันในเกม ซึ่งจะใช้โปรแกรมนี้ในการออกแบบภาพเหล่านี้โดยที่ลักษณะของภาพนั้นมีลักษณะเป็นมุมมอง มิตติการใช้โปรแกรมนี้ก็จะเป็นโปรแกรมที่มีความนิยมสามารถใช้ได้ง่ายรวมทั้งหาหนังสือเพื่อที่จะศึกษาเพื่อค้นคว้าในการที่ใช้งานที่ซ้ำซ้อนกว่านี้ได้

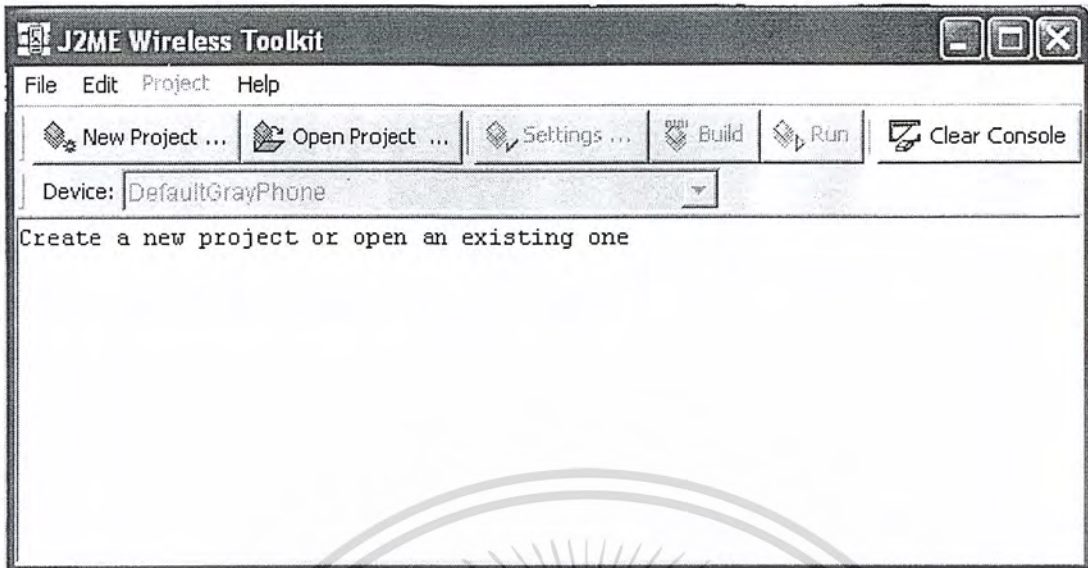


รูปที่ 10.7 โปรแกรม Adobe PhotoShop

10.4.2 J2ME Wireless Toolkit

เป็นโปรแกรมอิมูเลเตอร์ที่ใช้ในการสร้างโปรเจ็คของแอปพลิเคชันที่เขียนด้วยภาษา J2ME สามารถใช้คอมไพเลอร์ และรันแอปพลิเคชันนั้นๆ โดยสามารถเลือกอุปกรณ์ที่ต้องการจำลองการทำงานได้ซึ่งมีทั้งโทรศัพท์มือถือรุ่นต่างๆ หรืออาจเป็นพีดีเอก็ได้ โดยเราสามารถนำอิมูเลเตอร์ของอุปกรณ์ที่เราต้องการมาทำการใส่เข้าไปก็ได้โดยการใส่เข้าไปในโฟลเดอร์ อุปกรณ์ เท่านั้นก็สามารถเลือกใช้งานได้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 10.8 โปรแกรม J2ME Wireless Toolkit

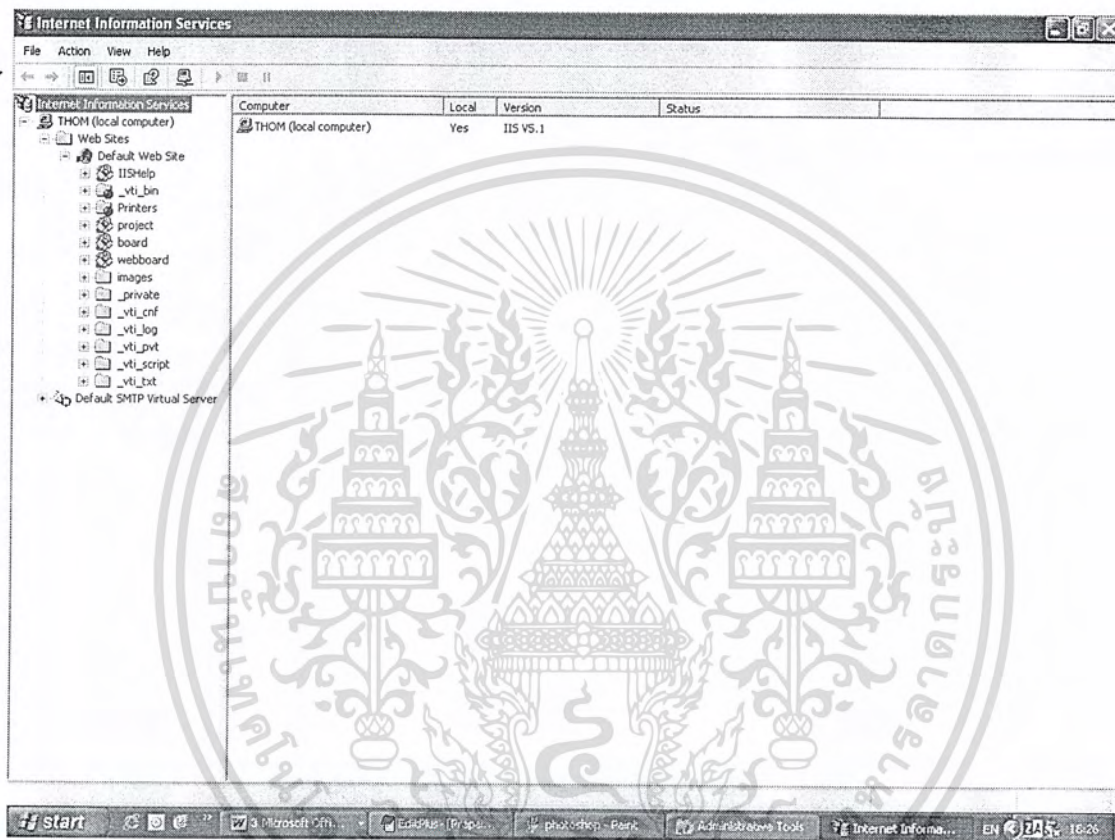


รูปที่ 10.9 อุปกรณ์จำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.4.3 IIS

เป็นโปรแกรมที่ใช้ในการจำลองการทำงานของเว็บเซิร์ฟเวอร์เพื่อใช้ในการจำลองการติดต่อของไคลเอนต์เพื่อที่จะให้เวลาเขียนโปรแกรมสามารถทดสอบได้ว่าการเขียนโปรแกรมต่าง ๆ นั้นสามารถทำงานได้จริงหรือไม่ ซึ่งการใช้งาน IIS นั้นเป็นโปรแกรมที่ใช้ในการทดสอบที่ใช้งานง่ายไม่มีอะไรยุ่งยาก โดยโปรแกรมนี้จะทำงานของมันเองไม่ต้องเซ็ทอะไรให้ยุ่งยาก เพียงแต่เราต้องกำหนดไดเรกทอรีที่เราต้องการจำลองการทำงานแบบเว็บเซิร์ฟเวอร์เท่านั้นก็สามารถทำงานได้



รูปที่ 10.10 โปรแกรม IIS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.5 การติดต่อ

ในการติดต่อนั้นเราใช้ มือถือเป็นฝั่ง client โดยใช้ภาษา J2ME ติดต่อกับเว็บเซิร์ฟเวอร์ที่ใช้ โพรโทคอล HTTP โดยการที่นำ HttpURLConnection มาใช้ในการติดต่อสื่อสารในแอปพลิเคชันซึ่งมีข้อดีหลายประการคือ

1. อุปกรณ์ MIDP บางอุปกรณ์ไม่สนับสนุนการติดต่อสื่อสารแบบซ็อกเก็ตและค่าค่าธรรมเนียมที่ทุกอุปกรณ์สนับสนุนการติดต่อสื่อสารแบบ HTTP

2. การติดต่อสื่อสารแบบซ็อกเก็ตและค่าค่าธรรมเนียมต้องอาศัยเครือข่ายค่อนข้างมาก ขณะที่บางเครือข่ายอาจเลือกใช้รูปแบบการติดต่อสื่อสารเพียงแบบใดแบบหนึ่งและไม่สนับสนุนแบบอื่นๆทำให้โยกย้ายแอปพลิเคชัน สำหรับอุปกรณ์ไร้สายได้น้อยลง

3. การกำหนดให้อุปกรณ์ MIDP สนับสนุนโพรโทคอล HTTP ทำให้แอปพลิเคชันบนอุปกรณ์ไร้สายมีโพรโทคอลระดับสูง (HttpURLConnection) เป็นมาตรฐาน และไม่ต้องอาศัยเครือข่ายไวใช้งาน นอกจากนี้ยังสามารถโยกย้ายข้ามเครือข่ายไร้สายต่างๆ ได้คืออีกด้วย

4. สามารถเก็บข้อมูลประเภทต่างๆ ลงในการร้องขอให้ HTTP ได้ง่ายโดยเฉพาะอย่างยิ่งหากนักพัฒนานำ XML มาใช้ในแอปพลิเคชันได้ การติดต่อสื่อสารแบบ HTTP ทำให้สามารถจัดการกับปัญหาต่างๆ ได้ดียิ่งขึ้น เช่น ความปลอดภัยของระบบและไฟร์วอลล์ เนื่องจากพอร์ต 80 ของ HTTP เป็นพอร์ตที่ถูกไฟร์วอลล์ปิดกั้นค่อนข้างน้อย

โดยการร้องขอ HttpURLConnection ไปยังเว็บเซิร์ฟเวอร์ทำได้ 3 ทางด้วยกัน คือ GET HEAD และ POST โดยเมธอดเหล่านี้มีการทำงานดังนี้

- เมธอด GET นำมาใช้เพื่ออ่านเนื้อหาของเอกสารในเว็บจาก URL ที่กำหนดไว้ การตอบรับของเว็บเซิร์ฟเวอร์ประกอบด้วยข้อมูลส่วนหัวและเอกสารในเว็บ ข้อมูลแบบ MIME เกี่ยวกับเนื้อหาข้อมูล และตัวเนื้อหาข้อมูล
- เมธอด HEAD นำมาใช้เพื่ออ่านข้อมูลเกี่ยวกับเอกสารในเว็บแทนที่จะเป็นเนื้อหาของเอกสารนั้นๆ เมื่อเว็บเซิร์ฟเวอร์ได้รับการร้องขอ HEAD จะมีเพียงข้อมูลส่วนหัวของ HTTP เท่านั้นที่จะตอบกลับมา (ไม่มีเนื้อหาข้อมูล)
- เมธอด POST นำมาใช้เพื่อส่งข้อมูลเกี่ยวกับลักษณะไปยัง URL ของโปรแกรม CGI ทั้ง POST และ GET ต่างถูกนำมาใช้เพื่อส่งข้อมูลไปยังโปรแกรม CGI ข้อแตกต่างอยู่ที่ เมธอด POST ส่งข้อมูลแบบ stream ขณะที่ เมธอด GET ส่งข้อมูลผ่าน environment ของตัวแปรที่ฝังตัวในสตริงการสอบถาม

โดยทางฝั่ง เว็บเซิร์ฟเวอร์นั้น ใช้ภาษา ASP ที่เลือกภาษา ASP นั้นเพราะว่า ภาษา XML นั้นไม่สนับสนุนในการพิจารณาเงื่อนไขและติดต่อกับ DATABASE ต่างๆ โดยตัว DATABASE นั้นใช้ Microsoft Access

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยด้านการออกแบบการติดต่อที่เราจะเลือกวิธีการติดแบบเมธอด GET เพื่อที่จะทำการส่งผ่านข้อมูลไปสู่นำเว็บเพจนั้น โดยตัวอย่างของการติดต่อการทำงานกับหน้าเว็บเพจแต่ละหน้าจะแสดงให้ดูรวมทั้งอธิบายการทำงานของการเขียน โปรแกรมการติดต่อดังกล่าวด้วย

ตัวอย่าง CODE แสดงการติดต่อแบบ GET

```
public String sendGetRequest(String urlString) throws IOException
{
    HttpURLConnection hc=null;
    DataInputStream dis=null;

    String message="";
    try
    {
        hc=(HttpURLConnection) Connector.open(urlString);
        dis=new DataInputStream(hc.openInputStream());
        int ch;
        while((ch=dis.read())!=-1)
        {
            message=message+(char)ch;
        }
    }
    finally
    {
        if(hc!=null) hc.close();
        if(dis!=null) dis.close();
    }

    return message ;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.5.1 การติดต่อตอนสร้าง ID และ PASSWORD ใหม่

ตัวอย่าง CODE แสดงการติดต่อ

```
public String chkid(String id,String pass)
{
    String urlstring = addidurl+id+"&pass="+pass;
    //system.out.println(urlstring);
    String resultstring="";
    try
    {
        resultstring=sendGetRequest(urlstring);
    }
    catch(IOException e)
    {
        //do nothing;
    }
    //-----
    if(!resultstring.equals(""))
    {
        realid = resultstring;
    }
    return resultstring;
}
```

ID	PASSWORD
----	----------

จาก CODE นั้นจะทำหลังจากการตรวจสอบว่าได้มีการใส่ ID, PASSWORD รั้ง แล้วได้ทำการใส่ PASSWORD กับ REPASSWORD นั้นตรงกันเรียบร้อยโดยเมื่อเงื่อนไขเหล่านี้เป็นจริงแล้วก็จะทำการส่ง ID กับ PASSWORD นั้นไปที่ URL ดังกล่าวที่มีเซิร์ฟเวอร์ที่ต้องการอยู่แล้วหลังจากนั้นจะทำการตรวจสอบว่าได้มีการใช้ ID นี้รั้งถ้ามีแล้วทางเซิร์ฟเวอร์จะไม่ทำการส่งอะไรกลับมาแต่ถ้ายังไม่มีจะสร้างหมายแล้วตัวละครใหม่ขึ้นมาแล้วทำการส่งค่านี้มาเพื่อบ่งบอกว่าเป็นตัวละครไหนสำหรับ ID ,PASSWORD นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.5.2 การติดต่อตอนส่งชื่อตัวละครและเลือกตัวละคร ใหม่

ตัวอย่าง CODE แสดงการติดต่อ

```
public void addnametype()
{
    //system.out.println(addnametypeurl);
    String resultstring="";
    try
    {
        resultstring=sendGetRequest(addnametypeurl);
    }
    catch(IOException e)
    {
        //resultstring="no";
    }
}
```

RealID	Char	Name
--------	------	------

จาก Code จะเห็นได้ว่าเป็นการส่งข้อมูลของชื่อและตัวละครไปจะไม่มีกรอ่านค่าอะไรกับมา โดยค่าของ addnametypeurl คือ addnametypeurl + realid + "&char=" + charecter_choice.getSelectedIndex() + "&name=" นั่นก็คือการส่งค่าต่างๆเหล่านี้แบบ GET นั่นเอง

10.5.3 การติดต่อตอนที่ทำการเซฟเกม

การที่จะทำการติดต่อในเวลาที่เราต้องการจะเซฟเกมนั้นก็จะทำการส่งข้อมูลต่างๆทั้งหลายของตัวละครที่เล่นอยู่ไปยังเว็บเซิร์ฟเวอร์เพื่อให้สามารถเก็บข้อมูลเหล่านั้นไปยังตัวละครที่ต้องการเซฟให้ถูกต้องล่าสุดเพื่อที่จะสามารถทำการโหลดข้อมูลต่างๆเหล่านั้นมาทำการเล่นต่อโดยหลักการทำงานแล้วในส่วนนี้ก็จะเหมือนตัวอย่าง CODE อื่นๆที่ทำการยกตัวอย่างมาแต่ส่วนที่แตกต่างออกไปนั่นก็คือในส่วน of ข้อมูลที่ทำการส่งไปเท่านั้นและสิ่งที่ส่งไปนั่นก็คือ

ReadID	Char	Name	Level	Human_hp	Exp	Red
White	M	L	K	Money	Job	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
addnametypeurl = "http://127.0.0.1/project/newnametype.asp?id="+realid+"&char="+Char+
"&name="+ name+"&level="+level+"&human_hp="+temp_human_hp+"&exp="+exp+"&red="
+temp_red+"&white="+temp_white+"&m="+canvas.m+"&l="+canvas.l+"&k="+canvas.k+"&money="
+money+"&job="+job;
```

10.5.4 การติดต่อตอนที่ทำการโหลดเกม

ตัวอย่าง CODE แสดงการติดต่อ

```
public String chkcon(String id,String pass)
{
    String urlstring = chkconurl+id+"&pass="+pass;
    //system.out.println(urlstring);
    String resultstring="";
    try
    {
        resultstring=sendGetchkcon(urlstring);
    }
    catch(IOException e)
    {
        //do nothing;
    }
    //-----
    if(!resultstring.equals(""))
    {
        realid = resultstring;
    }
    return resultstring;
}
```

ID	PASSWORD
----	----------

จาก CODE จะเห็นได้ว่าการส่ง ID กับ PASSWORD ไปเพื่อที่จะทำการติดต่อไปว่า ID กับ PASSWORD นี้ถูกต้องหรือไม่ ถ้าถูกต้องจะทำการส่งค่าต่างๆที่เกี่ยวกับตัวละครนั้นทั้งหมดมาเพื่อที่จะให้ผู้เล่นได้ทำการเริ่มเล่นต่อจากข้อมูลเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 11

การทดสอบการทำงาน

11.1 วิธีการทดสอบการทำงาน

ทำการเปิดเครื่องที่เป็นต้องการจะเป็นเซิร์ฟเวอร์ แล้วติดตั้งเว็บเซิร์ฟเวอร์ลงไป โดยในที่นี้ใช้ภาษา ASP ทางฝั่งเซิร์ฟเวอร์โดยใช้ IIS เป็นตัวจำลองเซิร์ฟเวอร์ แล้วจึงทำการเปิดแอปพลิเคชันทางฝั่งไคลเอนต์

11.2 ผลการทดสอบการทำงาน

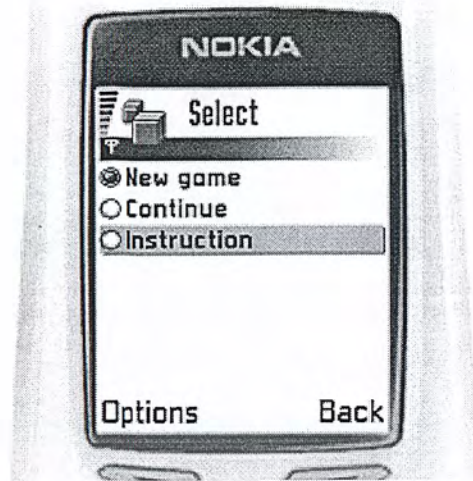
เมื่อเริ่มต้นเปิดแอปพลิเคชันทางฝั่งไคลเอนต์นั้น จะปรากฏหน้าจอ ดังรูปที่ 11.1



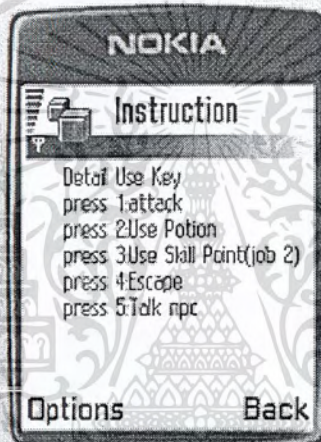
รูปที่ 11.1 หน้าจอหลักเมื่อเริ่มต้นเข้าสู่เกมส์

เมื่อเข้าสู่เกมส์จะปรากฏหน้าจอ ดังรูป 11.2 ซึ่งจะมี LIST ให้เลือก 3 ตัวเลือกนั้นก็คือ

- New Game นั้นคือต้องการสมัคร ID เพื่อสร้างตัวละครใหม่
- Continoue คือ ต้องการเล่นต่อในตัวละครเก่าที่เคยสร้างไว้แล้ว
- Instruction คือ แนะนำวิธีการเล่นเกมส์

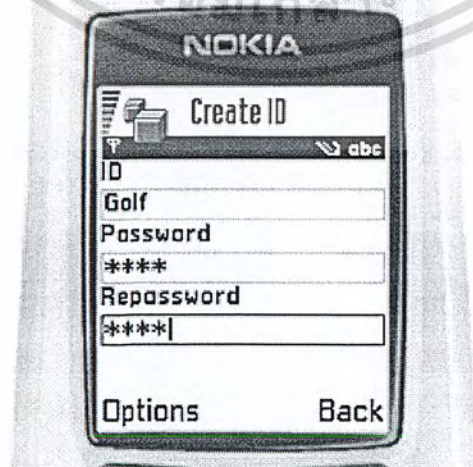


รูปที่ 11.2 หน้าจอเมื่อเริ่มต้นเข้าสู่เกมส์



รูปที่ 11.3 หน้าจอเมื่อเลือก Instruction เพื่อดูวิธีการเล่น

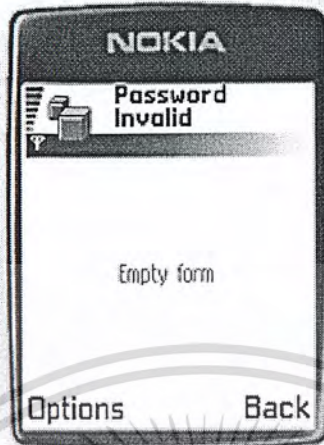
มีการสมัครเล่นเกมส์โดยการใส่ ID,PASSWORD เพื่อสมัครซึ่ง ID จะต้องไม่ซ้ำกับคนอื่นๆที่เคยมาสมัครไว้แล้วก่อนหน้านี้ดังรูปที่ 11.4



รูปที่ 11.5 หน้าจอเมื่อต้องการสมัครตัวละครใหม่

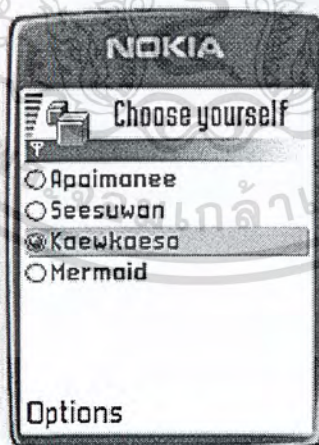
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการกรอกข้อความเพื่อสมัครนั้น นอกจาก ID จะต้องไม่ซ้ำกับคนอื่น ๆ แล้ว ยังต้องใช้ Password และ Repassword ให้ตรงกันอีกด้วย ถ้าใส่อะไรที่ผิดจากเงื่อนไขนี้จะปรากฏหน้าจอดังรูป 11.6



รูปที่ 11.6 หน้าจอเมื่อมีการป้อนข้อมูลที่ไม่ถูกต้องตามเงื่อนไข

เมื่อป้อนข้อมูลเสร็จเรียบร้อยแล้ว ถ้าข้อมูลที่ป้อนไปนั้นถูกต้องตามเงื่อนไขแล้ว จะปรากฏหน้าจอเพื่อเลือกตัวละคร ดังรูป 11.7 ซึ่งสามารถเลือกตัวละครได้หลายตัวเพื่อที่ที่ต้องการจะสวมบทบาทในการที่จะดำเนินเนื้อเรื่องของเกมส์ในลักษณะที่แตกต่างกันไปในแต่ละตัวละครในบางส่วนของเกมส์ เท่านั้น เพราะการที่ดำเนินเรื่องหลักทุกตัวจะต้องดำเนินเหมือนกัน ตัวละครดังกล่าวที่ว่ามีเช่น พระอภัยมณี ศรีสุวรรณ แก้วเกสา นางเงือก เป็นต้น



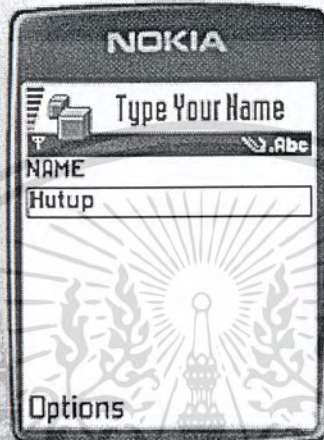
รูปที่ 11.7 หน้าจอเพื่อเลือกตัวละคร

โดยในที่นี้เราได้ทำการเลือกแก้วเกสา หลังจากนั้นจะปรากฏหน้าจอเพื่อให้เราใส่ชื่อของตัวละครของเรา ดังรูปที่ 11.8 ในที่นี้เราใส่ชื่อเป็น "hutup"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

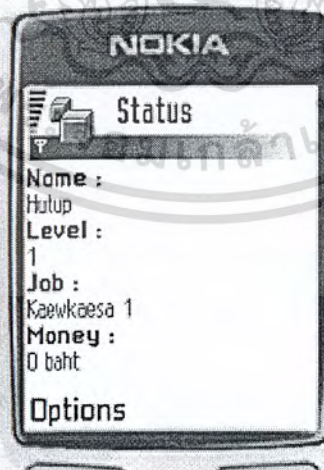
ตัวละครแต่ละตัวมีความสามารถที่ไม่เท่ากันในระดับเลเวลที่เท่ากันคือ

- พระอภัยมณีสามารถใช้ทำพิเศษได้แรงกว่าตัวละครอื่นในเลเวลเดียวกัน
- ศรีสุวรรณสามารถใช้ทำธรรมดาได้แรงกว่าตัวละครอื่นในเลเวลเดียวกัน
- แก้วเกสาสามารถซื้อของได้ถูกกว่าตัวละครอื่น
- นางเงือกเมื่อใช้ทำพิเศษแล้วจะทำให้เพิ่มพลังชีวิตอีกด้วย



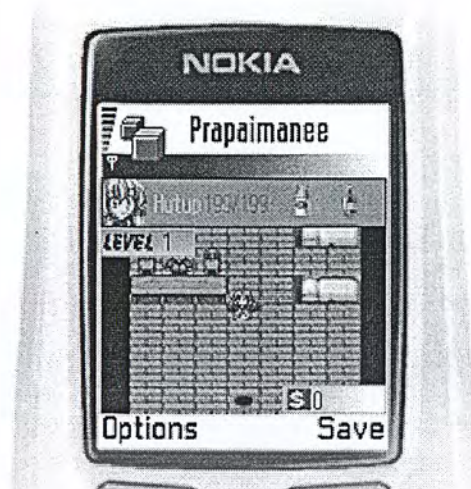
รูปที่ 11.8 หน้าจอเพื่อป้อนชื่อของตัวละคร

เมื่อทำการป้อนชื่อตัวละครเรียบร้อยแล้ว จะปรากฏหน้าแสดงสถานะของตัวละครเพื่อเป็นการยืนยันการสร้างตัวละครใหม่ของเราดังรูป 11.9



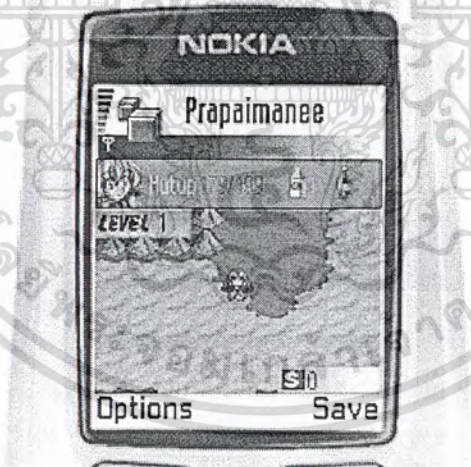
รูปที่ 11.9 หน้าจอสถานะของตัวละคร

หลังจากนั้นก็เข้าสู่หน้าจอเพื่อเล่นเกมจริงๆ ซึ่งจะแสดงให้เห็นถึงชื่อของตัวละคร พลังชีวิต เติมน้ำเงินที่มี และไอเทมต่างๆที่ตัวละครนั้นมีดังรูป 11.10 เอกสารนี้เป็นเอกสารที่สแกนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 11.10 หน้าจอปกติในเกมส์

การดำเนินเนื้อเรื่องจะมีความเก่งของศัตรูในแต่ละแผนที่ไม่เท่ากันซึ่งแผนที่ที่มีตัวละครที่เก่งก็จะ
ได้เงินและค่าประสบการณ์มากด้วยเช่นกัน การเกิดของศัตรูนั้นเกิดจากการสุ่มว่าจะเกิดที่พิกัดใดของแผนที่
ที่ การเดินบนแผนที่นั้นเราไม่สามารถจะรู้ได้ว่าศัตรูนั้นอยู่ที่พิกัดใดของแผนที่ซึ่งถ้าเราเดินไปยังบริเวณที่
มีศัตรูอยู่เราก็จะถูกย้ายฉากไปเป็นฉากต่อสู้เมื่อสู้เสร็จแล้วเราจะได้ค่าประสบการณ์และเงินดังรูปที่
11.11



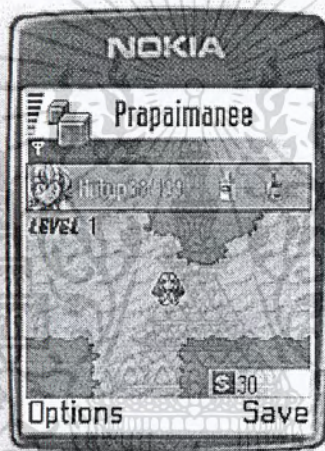
รูปที่ 11.11 หน้าจอบนแผนที่ปกติ

เมื่อเดินไปตกยังตำแหน่งที่มีศัตรูอยู่ก็จะเข้าสู่ฉากต่อสู้ ดังรูป 11.12 และเมื่อสามารถเอาชนะศัตรู
ได้เราก็จะได้ประสบการณ์และเงินมาดังรูป 11.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



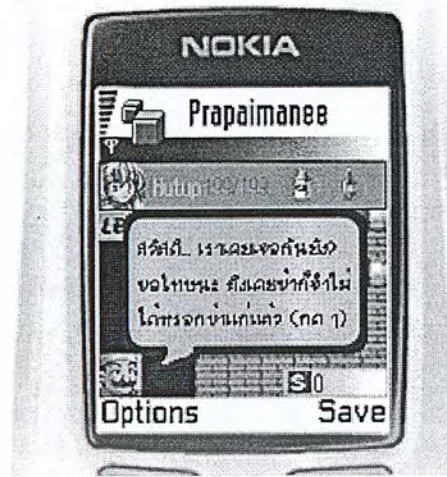
รูปที่ 11.12 หน้าจอเมื่อเข้าฉากต่อสู้



รูปที่ 11.13 หน้าจอเมื่อสามารถเอาชนะศัตรูได้

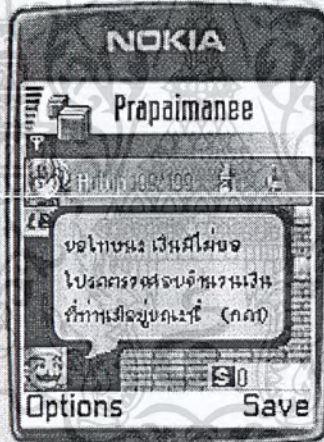
ของมีอยู่ 2 ชนิด คือ น้ำยาเพิ่มพลัง (น้ำผลไม้) และน้ำยาที่ใช้ทำพิเศษสิ่งของ 2 อย่างนี้สามารถซื้อได้ด้วยเงิน โดยที่เงินนั้น หาได้มาจากการฆ่าศัตรู การซื้อของจะต้องไปซื้อของบ้านที่มี NPC ที่ขายของขึ้นอยู่แล้วก็ทำการคุยกับ NPC เพื่อที่จะซื้อของที่ต้องการ ดังรูปที่ 11.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 11.14 หน้าจอการซื้อของจาก NPC

โดยในการซื้อของนั้นจะมีการตรวจเช็คด้วยว่า เงินที่เรามีอยู่นั้นมีเพียงพอหรือไม่ ถ้าไม่เพียงพอ จะปรากฏหน้าจอดังรูป 11.15



รูปที่ 11.15 หน้าจอเมื่อจำนวนเงินไม่พอ

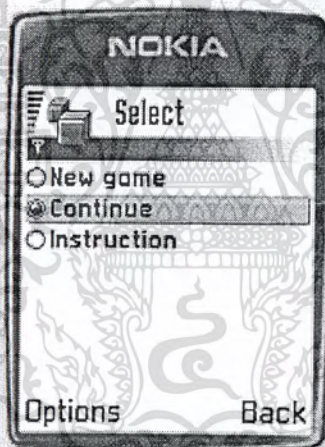
ในการต่อสู้นั้น ถ้าเราแพ้ต่อศัตรูจะขึ้นหน้าจอดังรูป 11.16 แล้วตัวเราจะทำการเซฟไปเกิดใหม่ที่ บ้านที่เราเกิดครั้งแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 11.16 หน้าจอเมื่อเราแพ้ต่อศัตรู

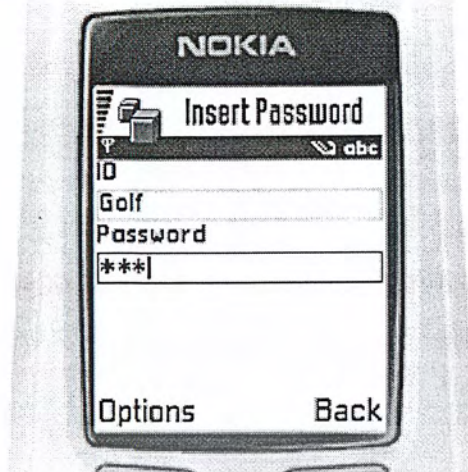
สำหรับการ Continue นั้น ในฉากปกติ ผู้เล่นสามารถเซฟสถานะตัวเองตอนไหนก็ได้ และทำการเซฟเองเมื่อ gameover เราสามารถกลับมาเล่นใหม่ได้โดย เลือกที่ Continue ดังรูป 11.17



รูปที่ 11.17 หน้าจอการเลือก Continue

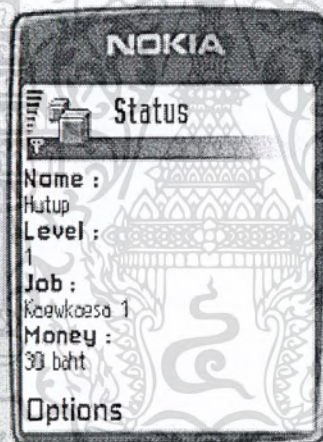
โดยในการที่เราจะเข้าไปเล่นในตัวละครนั้นๆ เราต้องป้อน ID และ Password ให้ถูกต้องเสียก่อน ซึ่งถ้าป้อนข้อมูลไม่ถูกต้องนั้น จะไม่สามารถไปยังหน้าถัดได้ ดังรูป 11.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 11.17 หน้าจอการป้อนข้อมูลเมื่อ Continue

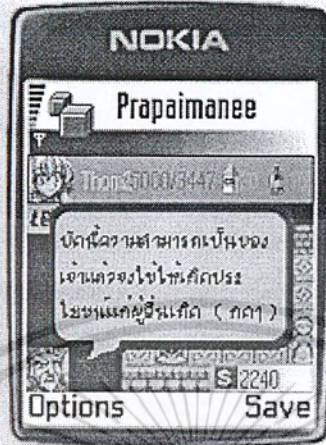
ถ้า ID และ Password ที่ใส่เข้าไปนั้นอยู่ใน DATABASE แล้วก็จะสามารถไปยังหน้าจอเพื่อดูสถานะของตัวละครของเรา ดังรูปที่ 11.18



รูปที่ 11.18 หน้าจอสถานะของตัวละคร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การดำเนินเนื้อเรื่องไปในระยะหนึ่งเมื่อเลเวลถึงข้อกำหนดแล้วในที่นี้คือเลเวล 20 จะสามารถไปคุยตามเนื้อเรื่องแล้วทำให้มีความสามารถพิเศษเพิ่มขึ้น โดยในรูปที่ 11.19 นั้นเป็นการดำเนินตามเนื้อเรื่องของแก้วเกศาเพื่อทำให้มีความสามารถพิเศษเพิ่มขึ้น



รูปที่ 11.19 หน้าจอการดำเนินตามเนื้อเรื่องของแก้วเกศา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 12

บทวิจารณ์และสรุป

12.1 สรุปผลการทดลอง

เกมสามารถโหลดและเซฟสถานะได้ สามารถเดินในแผนที่และพบกับศัตรูและต่อสู้กัน ได้สามารถดำเนินตามเนื้อเรื่องในแต่ละตัวละครได้เพื่อที่จะให้ได้ทำพิเศษของแต่ละตัวละครได้ สามารถคำนวณพลังการโจมตีต่างๆและสถานะต่างๆโดยขึ้นที่ทางหน้าจอได้ สามารถดำเนินเนื้อเรื่องจนจบได้ นั่นก็คือสามารถทำตามรูปแบบของเกมที่ออกไว้ได้ทั้งหมด

แต่ถึงการทำงานจะเสร็จสมบูรณ์ตามที่คาดไว้นั้นก็ยังมีปัญหาที่ควรคำนึงถึงนั่นก็คือเนื่องจากการเขียนโปรแกรมใน อุปกรณ์ ขนาดเล็กนั้นมีข้อจำกัดทางด้านความเร็วและหน่วยความจำ ดังนั้นการเขียนภาษา J2ME จึงมีข้อจำกัดต่างๆมากกว่าภาษาจาวาธรรมดา ทำให้การเขียนเป็นไปได้ยากขึ้นกว่าธรรมดาที่ภาษาเขียนบนเครื่องคอมพิวเตอร์

ปัญหาสำคัญที่เกิดขึ้นอีกประการก็คือ ขนาดของหน้าจอการแสดงผลนั้น ไม่ว่าจะเป็นปาล์มรุ่นต่างๆหรือมือถือแต่ละรุ่นก็จะมีหน้าจอการแสดงผลไม่เท่ากันดังนั้นการพัฒนา แอปพลิเคชัน นั้นต้องเลือกพัฒนาจำกัดว่าควรใช้กับ อุปกรณ์ ยี่ห้ออะไรรุ่นอะไรไปเลยไปและการพัฒนาจะต้องทำการพัฒนาในอีมีเลเตอร์ ของ อุปกรณ์ ยี่ห้อนั้นรุ่นนั้น แต่ก็อาจเกิดปัญหาที่ว่าสามารถใช้ใน อีมีเลเตอร์ ได้แต่พอนำมารันใน อุปกรณ์ จริงกลับรันไม่ได้ ทำให้ต้องเปลี่ยนรุ่นในการพัฒนาดังนั้น โปรแกรมที่เขียนเสร็จก็ต้องนำมาปรับปรุงอีกเพื่อให้ขนาดการแสดงผลต่างๆเหมาะสมกับรุ่นที่เราจะเปลี่ยนมาใช้

ปัญหาด้านการติดต่อเครือข่ายนั้นเนื่องมาจากความเร็วในการรับ-ส่งข้อมูลในปัจจุบันยังช้ามาก ทำให้การพัฒนา แอปพลิเคชัน ที่ติดต่อกับเครือข่ายบางครั้งเมื่อพัฒนาเสร็จก็อาจใช้งานจริงบน อุปกรณ์ ไม่ได้

12.2 แนวทางในการพัฒนาและประยุกต์ใช้งานร่วมกับงานอื่น

ในส่วนของโปรแกรมที่สร้างสามารถแบ่งได้ออกเป็น 3 ส่วน

1. ส่วนของการติดต่อกับ network โดยส่วนของการติดต่อกับเครื่องเซิร์ฟเวอร์เพื่อที่จะเซฟสถานะของตัวละครเดิมของผู้เล่นนั้นสามารถนำส่วนนี้ไปใช้ประโยชน์เพื่อที่จะนำข้อมูลที่สำคัญต่างๆไปเก็บได้เพื่อที่จะนำข้อมูลที่สำคัญนั้นไปใช้จากที่อื่นได้โดยไม่มีข้อจำกัดว่าต้องเป็นเครื่องที่เราทำงานอยู่เท่านั้นจึงจะสามารถทำงานต่อไปได้เราสามารถดึงข้อมูลเหล่านี้มาใช้งานกับ อุปกรณ์ อื่นที่มี แอปพลิเคชัน เดียวกันก็ได้

2. ส่วนของ High-Level API อินเทอร์เฟซระดับสูงซึ่งในส่วนนี้เป็นการติดต่อกับผู้เล่นเกมในตอนเริ่มแรกเพื่อที่จะป้อน ID และ Password ของตัวผู้เล่นเพื่อที่จะโหลดสถานะเดิมมาเพื่อเล่นต่อในส่วนนี้สามารถนำไปประยุกต์ใช้งานกับการป้อนข้อมูลต่างๆของผู้ใช้งาน แอปพลิเคชัน นั้นๆเพื่อนำข้อมูลเหล่านั้นไปใช้ประโยชน์ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ส่วนของ Low-Level API อินเทอร์เฟซระดับต่ำเป็นซึ่งเป็นส่วนหลักของเกมในการทำ animation ต่างๆของเกมซึ่งในส่วนนี้จะสามารถนำไปประยุกต์ต่อไปในการที่จะทำด้านกราฟิกซึ่งในส่วนนี้ผู้เขียน แอปพลิเคชัน สามารถออกแบบทุกอย่างเองได้ตามใจชอบ

ในส่วนของเกมส่นนั้นถือว่าสมบูรณ์ตามเป้าหมายที่ได้คาดหวังไว้ แต่ถ้าต้องการที่จะพัฒนาต่อโดยส่วนตัวของเกมเพื่อความสนุกหรือความน่าสนใจให้มากขึ้นนั้น เพื่อให้เป็นเกมออนไลน์ที่สมบูรณ์แล้ว ควรทำให้มีผู้เล่นได้หลายคน สามารถมองเห็นกันได้เมื่อผู้เล่นแต่ละคนนั้นอยู่ในแผนที่เดียวกันและอยู่ขอบเขตของหน้าจอของผู้เล่นด้วยโดยเมื่อมีผู้เล่นหลายคนแล้วก็ควรจะมีการช่วยเหลือกันเล่นในการเล่นโดยอาศัยความสามารถของแต่ละตัวที่ไม่เหมือนกันจึงมีข้อดีและข้อเสียต่างกันซึ่งอาจทำในลักษณะการเล่นแบบปาร์ตี้ ควรเพิ่มทางด้านรายละเอียดของเกมส่น เช่น ควรมีแผนที่เพิ่มขึ้น , เพิ่มของพิเศษต่างๆ เพื่อช่วยในการเล่นให้สนุกและหลากหลาย, เพิ่มเงื่อนไขในการเล่นเพื่อให้แต่ละตัวละครนั้นแตกต่างกันมากขึ้น หรืออาจเพิ่มตัวละครให้มากขึ้น ดังนั้นการพัฒนาต่อก็สุดแล้วแต่เทคนิคของแต่ละผู้พัฒนาซอฟต์แวร์ ว่าต้องการจะให้มันเป็นแนวใดต่อไปเพื่อที่จะให้เกิดความพัฒนาในการเขียนโปรแกรมเพื่อนำมาใช้กับชีวิตจริงให้มากที่สุด และสามารถเป็นแนวทางในการพัฒนาการเขียนโปรแกรมต่างๆได้อีกด้วยเช่นกัน



บรรณานุกรม

ภาษาไทย

[1] หนังสือ J2ME wireless java programming with Java 2 Micro Edition

แต่งโดย Yu Feng, Jun Zhu

[2] อินไซต์ ASP และ ASP.NET ฉบับสมบูรณ์ บริษัท โปรวิชั่น จำกัด

ผู้แต่ง/แปล ทวีชัย หงษ์สุมาลย์

ภาษาอังกฤษ

[3] John w. Muchow , “Core J2ME Technology & MIDP” , Prentice Hall PTR , Upper Saddle

River , NJ 07458

เว็บไซต์

[4] MrPalm website “www.mrpalm.com” (ค้นคว้าเกี่ยวกับการอุปกรณ์ไร้สาย)

[5] Midlet “www.MidletZone.com” (ตัวอย่าง MIDLET)

[6] Wireless Toolkit “www.java.sun.com” (ดาวน์โหลดอิมูเลเตอร์)

[7] <http://java.sun.com/aboutJava/communityprocess/>

[8] <http://java.sun.com/j2me/>

[9] <http://java.sun.com/product/cldc/>

[10] <http://java.sun.com/products/javaphone/>

[11] <http://www.sun.com/forte/ffj/>

[12] <http://sun.java.com/products/j2mewtoolkit/>

[13] <http://java.sun.com/product/midp/>

[14] <http://www.wirelessdevnet.com>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
แสดงคลาสไลบรารีของ CLDC

Java.lang

```

class java.lang.Object
class java.lang.Boolean
class java.lang.Byte
class java.lang.Character
class java.lang..Class
class java.lang.Integer
class java.lang.Long
class java.lang.Math
class java.lang.Runtime
class java.lang.Short
class java.lang.String
class java.lang.StringBuffer
class java.lang.System
class java.lang.Thread (implements java.lang.Runnable)
class java.lang.Throwable
class java.lang.Error
class java.lang.VirtualMachineError
class java.lang.OutOfMemoryError
class java.lang.Exception
class java.lang.ClassNotFoundException
class java.lang.IllegalAccessException
class java.lang.InstantiationException
class java.lang.InterruptedExecution
class java.lang.RuntimeException
class java.lang.ArithmeticException
class java.lang.ArrayStoreException
class java.lang.ClassCastException
class java.lang.IllegalArgumentException
class java.lang.IllegalThreadStateException
class java.lang.NumberFormatException
class java.lang.IllegalMonitorStateException

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

class java.lang.IllegalStateException
class java.lang.IndexOutOfBoundsException
class java.lang.ArrayIndexOutOfBoundsException
class java.lang.StringIndexOutOfBoundsException
class java.lang.NegativeArraySizeException
class java.lang.NullPointerException
class java.lang.SecurityException

```

java.io

```

class java.lang.Object
class java.io.InputStream
class java.io.ByteArrayInputStream (implements java.io.DataInput)
class java.io.OutputStream
class java.io.ByteArrayOutputStream
class java.io.DataOutputStream (implements java.io.DataOutput)
class java.io.PrintStream
class java.io.Reader
class java.io.InputStreamReader
class java.io.Throwable
class java.io.Exception
class java.io.IOException
class java.io.EOFException
class java.io.InterruptedIOException
class java.io.UnsupportedEncodingException
class java.io.UTFDataFormatException
class java.io.Write
class java.io.OutputStreamException
Interface java.io.DataInput
Interface java.io.DataOutput

```

Java.util

```

class java.lang.Object
class java.util.Calendar
class java.util.Date
class java.util.Hashtable
class java.util.Random

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

class java.lang.Throwable
class java.lang.Exception
class java.lang.RuntimeException
class java.util.EmptyStackException
class java.util.NoSuchElementException
class java.util.Timer
class java.util.TimerTask(implements java.lang.Runnable)
class java.util.TimeZone
class java.util.Vector
class java.util.Stack

```

javax.microedition.io

```

class java.lang.Object
class javax.microedition.io.Connector
class java.lang.Throwable
class java.io.IOException
class javax.microedition.io.ConnectionNotFoundException
interface javax.microedition.io.Connection
interface javax.microedition.io.DatagramConnection
interface javax.microedition.io.InputConnection
interface javax.microedition.io.StreamConnection
(also extends javax.microedition.io.OutputConnection)
interface javax.microedition.io.ContentConnection
interface javax.microedition.io.OutputConnection
interface javax.microedition.io.StreamConnection
(also extends javax.microedition.io.InputConnection)
interface javax.microedition.io.ContentConnection
interface javax.microedition.io.StreamConnectionNotifier
interface javax.io.DataInput
interface javax.microedition.io.Datagram(also extends java.io.DataOutput)
interface javax.io.DataOutput
interface javax.microedition.io.Datagram(also extends java.io.DataInput)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
 คลาสไลบรารีของ MIDP

Javax.microedition.lcdui

```

class java.lang.Object
class javax.microedition.lcdui.AlertType
class javax.microedition.lcdui.Command
class javax.microedition.lcdui.Display
class javax.microedition.lcdui.Displayable
class javax.microedition.lcdui.Canvas
class javax.microedition.lcdui.Screen
class javax.microedition.lcdui.Alert
class javax.microedition.lcdui.Form
class javax.microedition.lcdui.List(implements class javax.microedition.lcdui.Choice)
class javax.microedition.lcdui.TextBox
class javax.microedition.lcdui.Font
class javax.microedition.lcdui.Graphics
class javax.microedition.lcdui.Image
class javax.microedition.lcdui.Item
class javax.microedition.lcdui.ChoiceGroup(implements class javax.microedition.lcdui.Choice)
class javax.microedition.lcdui.DateField
class javax.microedition.lcdui.Gauge
class javax.microedition.lcdui.ImageItem
class javax.microedition.lcdui.StringItem
class javax.microedition.lcdui.TextField
class javax.microedition.lcdui.Ticker
Interface javax.microedition.lcdui.Choice
Interface javax. microedition.lcdui.CommandListener
Interface javax. microedition.lcdui.ItemStateListener
  
```

Javax.microedition.rms

```

class java.lang.Object
class javax.microedition.rms.RecordStore
class java.lang.Throwable
class java.lang.Exception
class java.microedition.rms.RecordStoreException
  
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
class java.microedition.rms.InvalidRecordIDException  
class java.microedition.rms.RecordStoreFullException  
class java.microedition.rms.RecordStoreNotFoundException  
class java.microedition.rms.RecordStoreNotOpenException  
Interface javax.microedition.rms.RecordComparator  
Interface javax.microedition.rms.RecordEnumeration  
Interface javax.microedition.rms.RecordFilter  
Interface javax.microedition.rms.RecordListener
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้