

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การพัฒนาเกมส่นบน iPhone

GAME DEVELOPMENT ON IPHONE



T119211



เลขหมู่.....  
เลขทะเบียน **119211**  
วัน,เดือน,ปี..... - 6 S.ค. 2554

b. 119211  
i. ....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีโอกาสนำไปใช้

ปริญญาโทปีการศึกษา 2553

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนาเกมส์บน iPhone

GAME DEVELOPMENT ON IPHONE

ผู้จัดทำ

1. นายธีรินทร์ รัตนพิเชษฐชัย รหัสนักศึกษา 50010721
2. นายพิรุณ ศุกฤทธิธำรง รหัสนักศึกษา 50011112
3. นายวีรยุทธ พงษ์พัชรินทร์ รหัสนักศึกษา 50011504



อาจารย์ที่ปรึกษา

(ดร. วรวัฒน์ ลิ้มโกทา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การพัฒนาเกมสับสน iPhone

นายธีรินทร์ รัตนพิเชษฐชัย 50010721

นายพิรุณ สุภฤทธิธำรง 50011112

นายวีรยุทธ พงษ์พัชรินทร์ 50011504

ดร. วรวัฒน์ ลิ้มโกภา อาจารย์ที่ปรึกษา

ปีการศึกษา 2553

### บทคัดย่อ

การพัฒนาเกมบน iPhone เป็นการพัฒนาเกมทำอาหาร ไทยที่ใช้เกมเอนจินที่เรียกว่า Cocos2d for iPhone ซึ่งเกมที่สามารถเล่นได้ทั้งคนเดียวและสองคน ในโหมดเล่นคนเดียวนั้นจะมีรายการอาหารต้มยำกุ้งและผัดไท ในแต่ละขั้นตอนการทำอาหารนั้นจะใช้เทคโนโลยีของที่เด่นๆของ iPhone อันได้แก่มัลติทัชและเซ็นเซอร์ในการรับรู้การเคลื่อนไหวเพื่อเพิ่มความสนุกของเกม ในโหมดเล่นสองคนจะใช้ GameKit Framework ในการจัดการส่งข้อมูลระหว่างสองเครื่องด้วย Bluetooth

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Game Development on iPhone

Mr. Terin      Ratanaphichetchai      50010721

Mr. Pirun      Supparittitamrong      50011112

Mr. Weerayut Pongpacharin      50011504

Dr. Voravat      Limpokha      Advisor

Academic Year 2010

## ABSTRACT

The game development on iPhone project is mainly focused on developing Thai cooking game using the game engine called Cocos2d for iPhone. The game can be played either single player mode or multi-player mode. In the single player mode, the cooking menu includes Tom Yum Kung and Pad Tai. There are numbers of main features incorporated with each stages of cooking such as Multi-Touch Technology and motion sensor to detect the movement. These features make the game more enjoyable. In the Multi-Player Mode, The application uses the GameKit Framework to enable and manage the data transfer between 2 devices with Bluetooth.

# กิตติกรรมประกาศ

ปริญญาบัตรฉบับนี้คงไม่อาจเสร็จได้ด้วยดีหากไม่ได้รับความช่วยเหลือจากหลายๆฝ่ายด้วยกันบุคคลแรกที่ต้องกล่าวถึงเพราะเป็นส่วนสำคัญที่ทำให้รายงานสำเร็จลุล่วงลงได้คือ ดร. วรวัฒน์ ลิ้ม โภคาอาจารย์ที่ปรึกษาของรายงานเล่มนี้ ที่ช่วยให้คำปรึกษาช่วยตรวจสอบแก้ไขข้อบกพร่องตลอดจนให้ความรู้และข้อคิดเห็นที่เป็นประโยชน์อย่างยิ่งซึ่งต้องขอขอบพระคุณเป็นอย่างมาก

ขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้คือบิดามารดาอันเป็นที่เคารพรักรักยิ่งที่ให้การสนับสนุนทางการศึกษาอย่างเต็มที่และคอยให้กำลังใจเสมอมาในทุกๆด้าน

ขอบคุณภาควิชาวิศวกรรมคอมพิวเตอร์ที่ให้ความช่วยเหลือและความอนุเคราะห์ในด้านอุปกรณ์

ขอบคุณพี่ภราดา นาคสูงที่ช่วยในการสมัครเป็นนักพัฒนาแอปพลิเคชันกับ Apple

ขอบคุณฝ่ายศิลป์อันได้แก่คุณเจตริน กายอรุณสุทธิ คุณคิโยชิ(Kiyochii) คุณวัชรเดช ลิ้มวรพันธ์ คุณสิริน สุวรรณพุกษา คุณวนพงษ์ แสงเทียน คุณหฤษฎ์ วงษ์ดารา และคุณไพโรจน์ กงสกุล ที่สละเวลาอันมีค่ามาช่วยวาดภาพประกอบให้

ขอบคุณบุคคลที่ให้ยืม iPhone และ iPod touch ในการทำงาน ได้แก่ คุณภาคภูมิ เวชวิทยาขลัง คุณภาณุเดช ใจเดช คุณณัฐภัทร โสทธิโสภา คุณพีรวัส พลวิเศษ คุณเจษฎา เจริญกิจวรวิฒิ คุณฐานุตร์ ปีมหัทธวิฒิ คุณจิรวัฒน์ ภูวศิรีวิวัฒน์ เพื่อทดสอบในการเล่นหลายคน

ขอบคุณสถานที่คือห้องปฏิบัติการคอมพิวเตอร์ Object oriented Language & Application Lab (OLALA) อาคาร ECC เป็นสถานที่ค้นคว้าข้อมูล

ธีรินทร์ รัตนพิเชษฐชัย

พิรุณ สุภฤทธิธำรง

วีรยุทธ พงษ์พัชรินทร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย .....	I
บทคัดย่อภาษาอังกฤษ .....	II
กิตติกรรมประกาศ .....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมา.....	1
1.2 วัตถุประสงค์ของการศึกษา.....	1
1.3 ขอบเขตโครงการ.....	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ .....	2
บทที่ 2 ไอโฟน.....	3
2.1 ไอโฟน .....	3
2.2 คุณสมบัติเด่น.....	6
2.3 เหตุผลที่เลือกเขียนเกมบนระบบไอโอเอส.....	6
บทที่ 3 ภาษาอ็อบเจกทีฟ-ซี.....	7
3.1 ภาษาอ็อบเจกทีฟ-ซี .....	7
3.2 ประวัติ.....	7
3.3 Syntax .....	9
3.4 ภาษาอ็อบเจกทีฟ-ซี 2.0 .....	12
3.5 ภาษาอ็อบเจกทีฟ-ซี พลัสพลัส.....	13
3.6 เปรียบเทียบภาษาที่ใช้เขียน โปรแกรมบนไอโฟน.....	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
บทที่ 4 เครื่องมือที่ใช้.....	15
4.1 เอ๊กโค้ด (XCode).....	15
4.2 เกมเอนจิน (Game Engine).....	20
บทที่ 5 ระบบเกม .....	26
5.1 เกมบนไอโฟน .....	26
5.2 ตัวอย่างเกมที่พัฒนาโดยทีมพัฒนาชาวไทย .....	31
5.3 ระบบภายในเกม .....	34
บทที่ 6 ชิ้นงานเกมไทยอร่อย.....	37
6.1 หน้าแรก.....	37
6.2 เมนูหลัก.....	37
6.3 เลือกเมนูอาหาร .....	38
6.4 เลือกชั้นตอนที่เล่น .....	38
บรรณานุกรม.....	42
ภาคผนวก ตัวอย่างโปรแกรมที่ทดลองเขียน .....	44

# สารบัญตาราง

ตาราง	หน้า
2.1 เปรียบเทียบคุณสมบัติของไอโฟนแต่ละรุ่น.....	3
ก.1 ขั้นตอนการปลดล็อก.....	61



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

รูป	หน้า
4.1 โปรแกรมที่ใช้สำหรับการพัฒนาแอปพลิเคชันบน ไอโอเอส.....	15
4.2 เลือกประเภทรูปแบบที่จะสร้าง.....	16
4.3 ตัวอย่างโปรเจกต์ที่อยู่ระหว่างการสร้างด้วย XCode.....	17
4.4 ตัวอย่างหน้าต่าง Interface Builder ขณะใช้งาน .....	18
4.5 ทดสอบโปรแกรมที่เขียนด้วยไอโฟน ซิมูเลเตอร์ .....	19
4.6 โลโก้ของ cocos2d for iPhone.....	20
4.7 ตัวอย่างฉากในเกมฮันนี่ ฟาร์ม .....	21
4.8 ตัวอย่างโลโก้ของ iTORQUE.....	22
4.9 ตัวอย่างฉากใน Hospital Havoc .....	23
4.10 ตัวอย่างโลโก้ของ iTORQUE.....	23
4.11 ตัวอย่างฉากใน Kids VS. Zombies.....	25
4.12 ตัวอย่างฉากใน Zombie Mansion.....	25
5.1 เกมแบบพีชเชิล .....	26
5.2 เกมนึค ฟอร์ สปีด.....	27
5.3 เกมเรด อเลิร์ต.....	27
5.4 เกมฮิลแอนดร์ริเวอร์.....	28
5.5 เกมเคออส ริง .....	28
5.6 เกมคุม .....	29
5.7 เกมเบลคคอฟเฟอร์.....	29
5.8 เกมคูกกิ้ง มามา.....	30
5.9 เกมอีเมอร์เจนซี่ แคนเจอร์ ไฟร์ ไฟต์เตอร์.....	32
5.10 เกมลิตเติ้ลกัปตัน .....	32
5.11 เกมเครซี่ฟาร์ม.....	33
5.12 เกมสปอตเดอะคิฟเฟอเรนซ์.....	33
5.13 การเล่นหลายคนแบบแข่งกัน .....	35
5.14 เกม Tic-Tac-Touch.....	35
5.15 เกม iPingPong 3D.....	36
5.16 เกม Scrabble .....	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูป	หน้า
6.1 เมนูเลือกระหว่างเล่นคนเดียวหรือเล่นหลายคน .....	37
6.2 เมนูหลัก .....	37
6.3 เลือกเมนูที่จะเล่น .....	38
6.4 เลือกขั้นตอนที่จะเล่น .....	38
6.5 หน้าจอการเล่นขั้นตอนปอกกุ้ง .....	39
6.6 หน้าจอการเล่นขั้นตอนหั่นตะไคร้ .....	39
6.7 หน้าจอการเล่นขั้นตอนหั่นข่า .....	40
6.8 หน้าจอการเล่นขั้นตอนหั่นเห็ด .....	40
6.9 หน้าจอการเล่นขั้นตอนบีบมะนาว .....	41
6.10 หน้าจอการเล่นขั้นตอนต้ม .....	41
ก.1 ตัวอย่างเมนู .....	44
ก.2 ทดสอบการสัมผัสและการตอบรับ .....	49
ก.3 การทดสอบมัลติทัช .....	54
ก.4 โปรแกรมเล่นไฟล์เสียงและร้องพร้อมแสดงภาพแกะ .....	59

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมา

ในปัจจุบันเทคโนโลยีมือถือก้าวหน้าไปเร็วมาก ดังเช่นตัวอย่างสมาร์ทโฟนที่เห็นให้ในปัจจุบัน ด้วยความสามารถที่เพิ่มมากขึ้นของโทรศัพท์มือถือในปัจจุบัน ทำให้เกิดคำว่า “สมาร์ทโฟน” ขึ้น ซึ่ง สมาร์ทโฟน หมายความว่า โทรศัพท์มือถือที่เพิ่มความสามารถของ PDA(Personal Digital Assistant)เข้าไปทำให้โทรศัพท์มือถือสามารถใช้งาน ได้อย่างมีประสิทธิภาพมากขึ้น เช่น การรับส่ง อีเมล ปฏิทินทำตารางนัดหมาย รายชื่อผู้ติดต่อจนเหมือนกับว่าเรามีเครื่องคอมพิวเตอร์ขนาดย่อมา ติดตัวไปด้วยตลอดเวลาเลยทีเดียว ซึ่งคุณสมบัติเด่นของสมาร์ทโฟน นั่นก็คือมีระบบปฏิบัติการ (Operation System : OS) เป็นระบบที่ช่วยให้การทำงานของโทรศัพท์มีประสิทธิภาพมากขึ้น สามารถเขียนโปรแกรมเพื่อไปทำการติดตั้งเข้ากับสมาร์ทโฟน(Smart Phone)นั้นๆ ซึ่งในปัจจุบันมี หลากหลายระบบปฏิบัติการด้วยกัน ซึ่งใน โครงการงานชิ้นนี้จะการศึกษาไอโอเอส(iOS)เนื่องจาก ไอโฟน(iPhone)มีเทคโนโลยีที่นาสนใจอย่างเทคโนโลยีสัมผัส(Multi-Touch Interface)และระบบ ตรวจจับการเคลื่อนไหว(Accelerometer)ซึ่งเป็นลูกเล่นที่ระบบปฏิบัติการ โทรศัพท์อื่นๆ ไม่ค่อยมี และเป็นเทคโนโลยีที่ใหม่ ที่สำคัญแอปเปิ้ล(Apple)ได้อนุญาตให้นักพัฒนาโปรแกรมทั่วไป สามารถเข้ามาร่วมเขียนแอปพลิเคชันลงบนมือถือไอโฟนได้ รวมไปถึงเกม

ดังนั้นการพัฒนาเกมบน ไอโฟนถือเป็นการพัฒนาเกมที่ใช้เทคโนโลยีใหม่ๆ และน่าสนใจ ทำให้เกมที่พัฒนาเสร็จแล้วมีความน่าสนใจ สนุกสนาน ทำทนายผู้เล่น ด้วยเทคโนโลยีตามที่กล่าวมา

### 1.2 วัตถุประสงค์ของการศึกษา

- 1) ศึกษาารูปแบบเกม ไอ โฟนต่างๆที่มีอยู่ในท้องตลาด
- 2) ศึกษาการพัฒนาเกมและแอปพลิเคชันบน ไอ โฟน
- 3) ออกเกมและพัฒนาเกมบน ไอ โฟน

### 1.3 ขอบเขตโครงการ

ในโครงการมีการขอบเขตในการทำงานคือการศึกษาและการพัฒนาเกมบน ไอ โฟนเท่านั้น จะ ไม่ทำการศึกษาหรือพัฒนาเกมบนคอนโซล(Console Game) สมาร์ทโฟนระบบอื่นที่ไม่เกี่ยวข้อง

ในโครงการนี้การศึกษาจะสืบค้นสื่อจากตำราภาษาต่างประเทศและภาษาไทย เว็บไซต์ ต่างประเทศและไทย โดยจะเน้นไปที่ตำราภาษาต่างประเทศและเว็บไซต์ต่างประเทศ

สำหรับการศึกษาดูอย่างเกมหรือแอปพลิเคชันนั้นการศึกษาจากตัวอย่างที่มีอยู่แล้วใน เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัจจุบัน โดยสมาชิกในกลุ่มจะเข้าร่วม สมัครเป็นผู้พัฒนาโปรแกรมของแอปเปิ้ลติดตั้ง โปรแกรม และทดลองใช้แอปพลิเคชันที่เกี่ยวข้องกับการพัฒนาเกมบนไอโฟนเป็นหลัก

สำหรับการออกแบบเกมนั้นจะใช้เกมลูกกึ่ง มาม่า(Cooking mama)มาเป็นพื้นฐานในการ ออกแบบเกมในโครงการนี้

#### 1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) มีความรู้ในการออกแบบและพัฒนาโปรแกรมบนไอโฟน
- 2) ตัวเกมที่สามารถเล่นบนไอโฟนได้จริง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# ไอโฟน

### 2.1 ไอโฟน

เป็นโทรศัพท์มือถือที่มีความสามารถใช้งานอินเทอร์เน็ตและมัลติมีเดีย ผลิตและจำหน่ายโดยบริษัทแอปเปิล โดยการทำงานของไอโฟนสามารถใช้งานส่งอีเมล ใช้เป็นโทรศัพท์เคลื่อนที่ ส่งเอสเอ็มเอส ท่องอินเทอร์เน็ตผ่านทางซอฟต์แวร์ซาฟารี ค้นหาแผนที่ ฟังเพลง และความสามารถอื่นๆ

การทำงานของโทรศัพท์ไอโฟนนี้จะแตกต่างจากโทรศัพท์มือถืออื่น โดยไอโฟนจะไม่มีปุ่มสำหรับกดหมายเลขโทรศัพท์ โดยการทำงานทั้งหมดจะทำงานผ่านหน้าจอโดยการสัมผัสสัมผัสทัชผ่านคำสั่งต่างๆ โดยมีระบบปฏิบัติการหลักไอโอเอส และมีระบบเซ็นเซอร์ในการรับรู้สภาพของเครื่องเพื่อกำหนดการแสดงผลของจอภาพ เช่นหากวางเครื่องในแนวตั้งระบบก็จะปรับให้แสดงผลในแนวตั้ง หากวางในแนวนอนระบบก็จะแสดงผลในแนวนอน

**ตาราง 2.1 เปรียบเทียบคุณสมบัติของไอโฟนแต่ละรุ่น**

รุ่น	ไอโฟน	ไอโฟน 3G	ไอโฟน 3GS	ไอโฟน 4
ระบบปฏิบัติการเบื้องต้น	iPhone OS 1.0	iPhone OS 2.0	iPhone OS 3.0	iOS 4.0
จอภาพ	3.5 in (89 mm), 3:2 aspect ratio, scratch-resistant glossy glass covered screen, 262,144-color LCD, 480 × 320 px (HVGA) at 163 ppi		In addition to previous, features a fingerprint-resistant oleophobic coating	3.5 in (89 mm), 3:2 aspect ratio, aluminosilicate glass covered IPS LCD screen, 960 × 640 px at 326 ppi, 800:1 contrast ratio
ความจุ	4, 8 และ 16 GB	8 และ 16 GB	8, 16 และ 32 GB	16 และ 32 GB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.1 เปรียบเทียบคุณสมบัติของไอโฟนแต่ละรุ่น (ต่อ)

หน่วยประมวลผลกลาง	620 MHz (underclocked to 412 MHz) Samsung 32-bit RISC ARM 1176JZ(F)-S v1.0	833 MHz (underclocked to 600 MHz) ARM Cortex-A8 Samsung S5PC100	~800 MHz ARM Cortex-A8 Apple A4
กราฟฟิก	PowerVR MBX Lite 3D GPU	PowerVR SGX535 GPU	
ความจำ	128 MB DRAM	256 MB DRAM	512 MB DRAM
เทคโนโลยีขั้นสูง	Wi-Fi (802.11b/g), USB 2.0/Dock connector, Quad band GSM/GPRS/EDGE (850, 900, 1800, 1900 MHz) Bluetooth 2.0 + EDR Cambridge Bluecore4	In addition to previous: Assisted GPS, Tri-band UMTS/HSDPA (850, 1900, 2100 MHz), Includes earphones with mic	In addition to previous: Penta-band UMTS/HSDPA (800, 850, 900, 1900, 2100 MHz), 5.76 Mbit/s HSUPA, Wi-Fi (802.11n: 2.4 GHz only), 3-axis gyroscope, Dual-mic noise suppression, microSIM
กล้อง	2.0 MP with geotagging	3 ล้านพิกเซล Autofocus focus ที่ปลายนิ้วแตะ	5 ล้านพิกเซล LED แฟลช เซนเซอร์วัดการส่องสว่าง ด้านหลัง Autofocus focus ที่ปลายนิ้วแตะ กล้อง ด้านหน้า ความละเอียด ขนาด VGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.1 เปรียบเทียบคุณสมบัติของไอโฟนแต่ละรุ่น (ต่อ)

Audio codec	Wolfson Microelectronics WM8758BG	Wolfson Microelectronics WM6180C	Cirrus Logic CS42L61	
วัสดุ	อลูมิเนียม แก้ว และพลาสติก	แก้วและพลาสติก; คำ ขาว (สีขาวไม่มีใน รุ่น 8 GB)	Aluminosilicate แก้ว และสแตนเลส; คำ ขาว	
พลังงาน	Built-in non removable rechargeable ลิเทียมไอออน โพลีเมอร์แบตเตอรี่			
	3.7 V 1400 mA·h	3.7 V 1150 mA·h	3.7 V 1219 mA·h	3.7 V 1420 mA·h
แบตเตอรี่ (ชม.)	ฟังเพลง: 24 ดูวิดีโอ: 7 ใช้คุยโทรศัพท์ 2G: 8 ใช้อินเตอร์เน็ต: 6 เปิดเครื่องรอ รับสาย: 250	ฟังเพลง: 24 ดูวิดีโอ: 7 ใช้คุยโทรศัพท์ 3G: 5 ใช้อินเตอร์เน็ต 3G: 5 ใช้อินเตอร์เน็ต Wi- Fi: 9 เปิดเครื่องรอรับสาย: 300	ฟังเพลง: 30 ดูวิดีโอ: 10 ใช้คุยโทรศัพท์ 3G: 5 ใช้อินเตอร์เน็ต 3G: 5 ใช้อินเตอร์เน็ต Wi- Fi: 9 เปิดเครื่องรอรับสาย: 300	ฟังเพลง: 40 ดูวิดีโอ: 10 ใช้คุยโทรศัพท์ 3G: 7 ใช้อินเตอร์เน็ต 3G: 6 ใช้อินเตอร์เน็ต Wi-Fi: 10 เปิดเครื่องรอรับสาย: 300
Dimensions	115 × 61 × 11.6 มม. (4.5 × 2.4 × 0.46 in)	115.5 × 62.1 × 12.3 มม. (4.5 × 2.4 × 0.48 in)	115.2 × 58.6 × 9.3 มม. (4.5 × 2.31 × 0.37 in)	
น้ำหนัก	135 g (4.8 oz)	133 g (4.7 oz)	135 g (4.8 oz)	137 g (4.8 oz)
เริ่มต้นการผลิต (สหรัฐฯ)	4 และ 8 GB: 29 มิถุนายน 2550 16 GB: 5 กุมภาพันธ์ 2551	11 กรกฎาคม 2551	16 and 32 GB: 19 มิถุนายน 2552 Black 8 GB: 24 มิถุนายน 2553	24 มิถุนายน 2553
สิ้นสุดการผลิต (สหรัฐฯ)	4 GB: 5 กันยายน 2550 8 และ 16 GB: 11 กรกฎาคม 2551	16 GB: 8 มิถุนายน 2552 Black 8 GB: June 4, 2553	16 และ 32 GB: 24 มิถุนายน 2553 Black 8 GB: อยู่ ระหว่างการผลิต	อยู่ระหว่างการผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2 คุณสมบัติเด่น

### 2.2.1 มัลติทัช

เป็นการต่อยอดมาจากหน้าจอสัมผัสทั่วไป ซึ่งโดยปรกติถ้าเป็นทัชสกรีนธรรมดาจะเป็นการรับคำสั่งได้ที่ละจุดทีละคำสั่ง คล้ายๆกับเวลาเราเล่นเกมสลับทิศทาง หรือการใช้โทรศัพท์มือถือหรือPDAnั้นเอง แต่ว่ามัลติทัชจะต่างออกไปเพราะสามารถรองรับการสัมผัสได้ที่ละหลายๆจุดทำให้ เกิดรูปแบบการสั่งงานที่คล่องตัวมากขึ้นและก็มีการควบคุมที่สะดวกกว่า ให้ความรู้สึกที่แตกต่างออกไปจากการควบคุมคอมพิวเตอร์แบบเดิมๆ

### 2.2.2 เซ็นเซอร์ในการรับรู้การเคลื่อนไหว

เป็นเซ็นเซอร์ที่คอยตรวจจับว่าเครื่องถูกถือในแนวใด แล้วหน้าจอก็จะแสดงผลไปตามแนวที่ผู้ใช้ถือ เพื่อสร้างความสะดวกในการทำงานให้กับผู้ใช้ ซึ่งในภายหลังได้ถูกนำมาประยุกต์ใช้กับระบบควบคุมเกม และได้กลายมาเป็นจุดเด่นของเกมบนไอโอเอสในที่สุด

## 2.3 เหตุผลที่เลือกเขียนเกมบนระบบไอโอเอส

### 1) มีฐานผู้ใช้งานจำนวนมาก

โดยในงานเปิดตัวไอโอเอสตามนั้นแอปเปิ้ลได้แจ้งว่ามียอดผู้ใช้งานอยู่ 30 ล้านเครื่องทั่วโลก (นับรวมทั้งไอพอดทัชและไอโฟน) ซึ่งในปัจจุบันก็ยังมีผู้ใช้เพิ่มขึ้นอย่างต่อเนื่อง อีกทั้งแอปเปิ้ลยังเปิดตัวผลิตภัณฑ์ใหม่ที่ใช้ไอโอเอสอย่างไอแพด ซึ่งแน่นอนว่ามีกระแสตอบรับจากผู้ที่มีความสนใจเป็นอย่างดี

### 2) มีตลาดรองรับ

โดยทาง Apple ได้เปิดระบบ app store ให้ผู้พัฒนาโปรแกรมภายนอกสามารถส่งโปรแกรมเข้าไปขายได้ ทำให้การขายโปรแกรมทำได้ง่ายและเป็นระเบียบ โดยที่ผู้พัฒนาไม่จำเป็นต้องลงทุนเรื่องของบรรจุภัณฑ์หรือระบบการขายเองเลย

### 3) เป็นการขายแบบเน้นปริมาณ

เนื่องจากโดยทั่วไปแล้ว โปรแกรมบน App Store จะขายที่ราคาถูกมาก(ประมาณ 30บาท) จึงทำให้ผู้บริโภคตัดสินใจซื้อได้ง่าย และกลุ่มของผู้ซื้อก็กระจายตัวอยู่ทั่วโลก ไม่ได้กระจุกตัวอยู่แค่ในประเทศเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

# ภาษาอ็อบเจกทีฟ-ซี

### 3.1 ภาษาอ็อบเจกทีฟ-ซี

ภาษาอ็อบเจกทีฟ-ซี (Objective-C หรือ ObjC) เป็นภาษาโปรแกรมเชิงวัตถุและมีสมบัติการสะท้อน โดยแรกเริ่ม ภาษาอ็อบเจกทีฟ-ซี พัฒนารูปร่างจากภาษาซีโดยยังคงคุณลักษณะของภาษาซีไว้ครบทุกประการเพียงแต่เพิ่มระบบส่งข้อความ (messaging) แบบเดียวกับภาษาสมอลล์ทอล์กเข้าไปเท่านั้น (Objective-C runtime) ปัจจุบันภาษาอ็อบเจกทีฟ-ซีมีคุณสมบัติอื่นๆเพิ่มเติมจากการพัฒนาภาษาอ็อบเจกทีฟ-ซี 2.0 โดยบริษัทแอปเปิล

ปัจจุบันภาษาอ็อบเจกทีฟ-ซีถูกใช้มากใน Cocoa (API) ใน Mac OS X, GNUstep (API) และ Cocotron (API) เป็นต้น ซึ่งระบบเหล่านี้ได้รับการพัฒนาขึ้น โดยมีพื้นฐานจากมาตรฐาน OpenStep (API) ใน Nextstep (Operating system) โดยมีภาษาอ็อบเจกทีฟ-ซีเป็นภาษาหลัก ปัจจุบัน Mac OS X ใช้ Cocoa เป็นเฟรมเวิร์กสำหรับสร้าง โปรแกรมประยุกต์ โดย ไลบรารีและ/หรือ API เหล่านี้เป็นเพียงส่วนเพิ่มขยาย (Software extension) เท่านั้น โปรแกรมที่ใช้ภาษาอ็อบเจกทีฟ-ซีทั่วไปที่ไม่ได้ใช้ส่วนเพิ่มขยายเหล่านี้ก็ยังสามารถคอมไพล์ได้ เช่นอาจใช้แค่ gcc ซึ่งรองรับภาษาอ็อบเจกทีฟ-ซี

### 3.2 ประวัติ

ในช่วงต้นของปี 1980s วิศวกรรมซอฟต์แวร์นิยมออกแบบ โปรแกรมแบบ โครงสร้าง เรานิยามการออกแบบ โปรแกรมแบบ โครงสร้างขึ้นเพื่อแยกย่อย โปรแกรมขนาดใหญ่ลงเป็นส่วนเล็กๆ เพื่อให้ง่ายต่อการจัดการเมื่อโปรแกรมมีขนาดใหญ่ขึ้น อย่างไรก็ตาม โปรแกรมแบบ โครงสร้างก็มีประโยชน์น้อยลงเมื่อนำขนาดของปัญหาใหญ่ขึ้น เพราะต้องเขียน procedure จำนวนมากเพื่อรองรับปัญหาที่ใหญ่ขึ้น และทำให้รหัสคำสั่งมีความซับซ้อนและยุ่งเหยิง

ทางเลือกหนึ่งสำหรับแก้ปัญหาดังกล่าวคือการเขียน โปรแกรมเชิงวัตถุ ที่จริงแล้วภาษาสมอลล์ทอล์กก็ได้แก้ไขข้อบกพร่องในกระบวนการวิศวกรรมซอฟต์แวร์เหล่านี้ แต่ในอดีตสมอลท์อล์กก็มีปัญหาในแง่ของความเร็วและการใช้หน่วยความจำมากเนื่องจากคอมไพเลอร์ในสมัยก่อนมีข้อจำกัดมากกว่าในปัจจุบัน

ในต้นยุค 80 Brad Cox และ Tom Love ได้ให้พัฒนาภาษาอ็อบเจกทีฟ-ซีขึ้นที่บริษัท Stepstone ของพวกเขา โดยพวกเขาได้เรียนรู้ภาษาสมอลล์ทอล์กจาก Programming Technology Center ของบริษัท ITT Corporation ในปี.ศ.1981 Cox ให้ความสนใจกับปัญหาเรื่องการนำรหัสมาใช้ซ้ำ

(reusability) ในกระบวนการวิศวกรรมซอฟต์แวร์ เขาคิดว่าภาษาสมอลล์ทอล์กไม่เหมาะจะใช้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พัฒนา development environment สำหรับนักพัฒนาซอฟต์แวร์ระบบที่ ITT Cox เริ่มแก้ไข คอมไพเลอร์ภาษาซี โดยเพิ่มความสามารถด้านการจัดการเชิงวัตถุของภาษาสมอลล์ทอล์คเข้าไป โดยเขาเรียกมันว่า "OOPC" หมายถึง Object-Oriented Programming in C ในขณะนั้น Love ซึ่งทำงานให้กับ Schlumberger Research ในปี 1982 ก็ได้มีโอกาสใช้งาน Smalltalk-80 ซอฟต์แวร์ซึ่งส่งอิทธิพลต่อการพัฒนาภาษาอ็อบเจกทีฟ-ซีในเวลาต่อมา

Smalltalk จัดได้ว่าเป็นภาษาที่มี Productivity สูงมาก เขียนสั้นกระชับรัด และ Object Oriented Programming เต็มรูปแบบ แต่ในยุคนั้น ภาษา SmallTalk ไม่ค่อยดังเพราะว่ามันต้องการกำลัง Hardware ที่สูงมาก เพราะทุกอย่างเป็น Graphics แต่ปัจจุบันมีคนใช้อยู่ไม่น้อย บรรดาปรมาจารย์ด้าน Object Oriented Programming ในยุคปัจจุบัน มีไม่น้อยที่เป็น Programmer ภาษา SmallTalk มาก่อน ภาษา Smalltalk ยังสอนให้โลกรู้จัก Garbage Collection ด้วย

และเพื่อแสดงประสิทธิภาพ Cox ได้แสดงให้เห็นว่าการปรับปรุงเครื่องมือที่มีอยู่แล้วเพียงเล็กน้อยก็สามารถทำ software component ให้ปรับเปลี่ยนได้ง่ายๆ โดยการทำให้ยืดหยุ่น และสนับสนุนด้วยชุดไลบรารีซึ่งประกอบด้วยรหัสและรีซอร์ส ที่รวมกันอยู่ในรูปแบบที่สามารถนำไปใช้ข้าม platform ได้

Cox และ Love ได้ก่อตั้ง Productivity Products International (PPI) เพื่อขายคอมไพเลอร์ภาษาอ็อบเจกทีฟ-ซีและคลาสไลบรารีในเวลาต่อมา

ในปีค.ศ. 1986 Cox ได้ตีพิมพ์หนังสือที่อธิบายถึงภาษาอ็อบเจกทีฟ-ซีชื่อ Object-Oriented Programming, An Evolutionary Approach ซึ่งแม้ว่า Cox จะได้พยายามชี้ให้เห็นว่าปัญหาหลักคือการนำรหัสมาใช้ซ้ำ (reusability) มากกว่าจะเป็นการแก้ปัญหาด้วยภาษา แต่กระนั้น Objective-C ก็ยังถูกนำไปเปรียบเทียบแบบที่เจอร์กับภาษาอื่นๆอยู่ดี

ในปีค.ศ. 1988 บริษัท NeXT ซึ่งตั้งขึ้นโดย Steve Jobs ได้ลิขสิทธิ์จาก StepStone (เจ้าของเครื่องหมายการค้าภาษาอ็อบเจกทีฟ-ซีในขณะนั้น) โดยได้พัฒนาคอมไพเลอร์ภาษาอ็อบเจกทีฟ-ซีและชุดไลบรารี ของตัวเองโดยนำมาใช้พัฒนาระบบติดต่อผู้ใช้และระบบพัฒนาซอฟต์แวร์ชื่อ NEXTSTEP ถึงแม้ว่าเครื่อง NeXT จะขายไม่ดี แต่เครื่องมือพัฒนาของมันกลับได้รับความนิยมพอสมควร และในที่สุด NeXT ก็เลิกขายฮาร์ดแวร์ และหันมาขายซอฟต์แวร์แทน ภายใต้ชื่อ NeXTstep และ OpenStep

โครงการกนู ได้พัฒนาระบบพัฒนาซอฟต์แวร์หนึ่งตามมาตรฐาน OpenStep ในรูปแบบซอฟต์แวร์เสรี โดย Dennis Glatting ได้พัฒนา gnu-objc runtime ตัวหนึ่งขึ้นในปี 1992 โดย Richard Stallman ได้เขียนอีกตัวขึ้นแทนในเวลาไล่เลี่ยกัน ส่วน GNU Objective-C runtime ตัวที่ถูกใช้งานตั้งแต่ 1993 พัฒนาโดย Kresten Krab Thorup เมื่อเขาเป็นนักเรียนมหาวิทยาลัยในเดนมาร์ก หลังจากที่ซื้อกิจการของ NeXT ในปี 1996 Apple ได้ใช้ OpenStep ในระบบปฏิบัติการของตนเอง หรือ Mac OS X โดยได้รวมเอาภาษาอ็อบเจกทีฟ-ซีและระบบพัฒนาซอฟต์แวร์จาก NeXT ชื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Builder ซึ่งถูกแทนที่โดย Xcode ในเวลาต่อมา รวมถึงระบบออกแบบระบบการติดต่อผู้ใช้แบบ object ชื่อ Interface Builder และ Apple เรียกไลบรารีซึ่งพัฒนาต่อจาก OpenStep ว่า Cocoa

### 3.3 Syntax

ภาษาอ็อบเจกทีฟ-ซี เป็นชั้นบางๆ บน C และเป็น สตรีคต์ superset ของ C ดังนั้นคอมไพเลอร์ภาษาอ็อบเจกทีฟ-ซีจึงสามารถคอมไพล์โปรแกรมภาษาซีใดๆ ก็ได้ ภาษาอ็อบเจกทีฟ-ซี ได้รับรูปแบบการเขียนมาจากภาษาซีและภาษาสมอลล์ทอล์ก โดยรูปแบบส่วนใหญ่ (preprocessing, expressions, การประกาศฟังก์ชัน และการเรียกฟังก์ชัน) มาจากภาษาซี ขณะที่ส่วนที่เป็นการจัดการเชิงวัตถุมาจากสมอลทอล์ก

#### 3.3.1 Messages

ภาษาอ็อบเจกทีฟ-ซี ได้เพิ่มเติมรูปแบบการเขียนโปรแกรม เพื่อรองรับการออกแบบโปรแกรมเชิงวัตถุ โดยจะใช้การส่ง message ไปยัง object ต่างๆเช่นเดียวกับสมอลทอล์ก ซึ่งแตกต่างจากภาษาในตระกูล Simula (เช่น C++) ข้อแตกต่างนี้มีความสำคัญ เพราะภาษาอ็อบเจกทีฟ-ซีจะไม่เรียก method แต่จะส่ง message

ในภาษาอ็อบเจกทีฟ-ซี ถ้ามี object หนึ่งชื่อ obj โดย class มี method ชื่อ doSomething หมายความว่า obj respond หรือตอบสนองต่อ message doSomething และถ้าเราต้องการจะส่ง message doSomething ไปยัง obj เราจะเขียนคำสั่งดังนี้

```
[obj doSomething];
```

ขณะที่ถ้าเป็น C++ เราจะเขียนว่า

```
obj.doSomething ();
```

##### 3.3.1.1 Forwarding

ในภาษาอ็อบเจกทีฟ-ซีจะยอมให้มีการส่ง message ไปยัง object ใดๆ แม้ว่าจะไม่มีการเตรียม method เอาไว้รองรับ (คือไม่ respond) ต่างจากภาษาอื่นๆ เช่น C++ หรือ Java ที่การเรียกใช้ method ต้องมีการระบุไว้ล่วงหน้า หาก object ได้รับ message ที่ไม่รู้จัก object จะผ่านต่อ message ที่ได้รับไปยัง method เหล่านี้

#### โปรแกรม 3.1 ภาษาอ็อบเจกทีฟ-ซีส่ง message

```
- (retval_t) forward:(SEL) sel :(arglist_t) args; //
with GCC
- (id) forward:(SEL) sel :(marg_list) args; // with
NeXT/Apple systems
```

Method เหล่านี้อาจแตกต่างกันไปตามชนิดของ runtime และมักนิยมผ่านเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อไปยัง method อื่นๆ ในระดับของ framework เช่น forwardInvocation: ใน OpenStep

### 3.3.2 อินเทอร์เฟซ และ อิมพลีเมนเทชัน

ในภาษาอ็อบเจกทีฟ-ซี ส่วนอินเทอร์เฟซ (@interface) และอิมพลีเมนเทชัน (@implementation) จะถูกแยกออกจากกัน ในทางปฏิบัติ เรามักเก็บส่วนอินเทอร์เฟซไว้ในแฟ้ม .h และส่วนอิมพลีเมนเทชันใน .m

#### 3.3.2.1 @interface

เรามักนิยามส่วนอินเทอร์เฟซของคลาสในแฟ้ม .h โดยทั่วไปเรามักตั้งชื่อแฟ้มนี้ให้ตรงหรือสอดคล้องกับชื่อของคลาส เช่นถ้าคลาสเราชื่อ Thing เราก็มักจะประกาศอินเทอร์เฟซของคลาส Thing ในแฟ้มThing.h

#### โปรแกรม 3.2 รูปแบบของการประกาศอินเทอร์เฟซ

```
@interface classname : superclass name
{
    instance variables
}
+ classMethod1;
+(return_type) classMethod2;
+(return_type) classMethod3:(param1_type)
parameter_varName;

-(return_type) instanceMethod1: (param1_type)
param1_varName : (param2_type) param2_varName;
-(return_type) instanceMethod2WithParameter:
(param1_type) param1_varName andOtherParameter:
(param2_type) param2_varName;
@end
```

method แบบ instance จะถูกนำหน้าด้วยเครื่องหมายลบ "-" ส่วน method แบบ class จะถูกนำหน้าโดยเครื่องหมายบวก "+" ตรงนี้จะแตกต่างจาก UML diagrams ซึ่งใช้ในแสดงว่า method เป็นแบบ private หรือ public

ค่าที่ถูก return จาก method มีลักษณะเช่นเดียวกับในภาษาซี เช่น void, int, ฯลฯ โดยจะมีการประกาศ type ชื่อ id ไว้แทน instance อ็อบเจกอะไรก็ได้

เราประกาศพารามิเตอร์ของ method ด้วยเครื่องหมายวงเล็บหรือโคลอน ":" ตามด้วย type ของพารามิเตอร์ในวงเล็บ แล้วตามด้วยชื่อของพารามิเตอร์ เรามักใส่ชื่อที่มีความสอดคล้องกับพารามิเตอร์หน้าเครื่องหมายวงเล็บ เพื่อบอกว่าพารามิเตอร์แต่ละตัวมีบทบาทอย่างไร

### โปรแกรม 3.3 ประกาศmethod

```
-(void) setRange: (int) start : (int) end;
-(void) importDocumentWithName: (NSString *) name
withSpecifiedPreferences: (Preferences *) prefs
beforePage: (int) insertPage;
```

#### 3.3.2.1 @implementation

ส่วนอินเตอร์เฟซจะประกาศแต่ต้นแบบของ method เท่านั้น โดยไม่รวมถึงการกำหนดว่า method นั้นจะต้องทำอะไรบ้าง การกำหนดว่า method นั้นจะต้องทำอะไรจะทำในส่วนอิมพลีเม้นเทชัน ตามปกติ ส่วนอิมพลีเม้นเทชันจะอยู่ในไฟล์นามสกุล .m ตัวอย่างเช่น Thing.m ที่มีการกำหนดส่วนอิมพลีเม้นเทชันไว้ดังนี้

### โปรแกรม 3.4 รูปแบบของการประกาศอิมพลีเม้นเทชัน

```
@implementation classname
+ classMethod
{
    // implementation
}
- instanceMethod
{
    // implementation
}
@end
```

method มีหน้าตาแตกต่างจากฟังก์ชันในภาษา C เช่นถ้าในภาษาซีเป็นอย่างนี้

### โปรแกรม 3.5 รูปแบบของการประกาศฟังก์ชันในภาษาซี

```
int do_something (int i)
{
    return square_root (i) ;
}
```

โดยมี int do\_something (int) เป็นต้นแบบ

เมื่อมาประกาศเป็น method ในภาษาอ็อบเจกทีฟ-ซี จะมีหน้าตาอย่างนี้

### โปรแกรม 3.6 รูปแบบของการประกาศmethod ในภาษาอ็อบเจกทีฟ-ซี

```
-(int) do_something: (int) i
{ return [self square_root: i];}
```

ถ้าให้ดูกรรมเนียบเราจะกำหนดชื่อ method ข้างต้นใหม่ โดยเราจะตั้งชื่อ method ให้สอดคล้องกับ argument ตัวแรกดังนี้

### โปรแกรม 3.7 รูปแบบของการประกาศmethod ในภาษาอ็อบเจกทีฟ-ซี2

```
- (int) doSomethingWithInt: (int) i
{
return [self squareRootOfInt:i];
}
```

แม้ว่าอาจจะดูยุ่งยากแต่ก็ช่วยให้เราจำ argument ได้ง่ายขึ้น ตัวอย่างเช่น

### โปรแกรม 3.8 ใช้ method ในภาษาอ็อบเจกทีฟ-ซี

```
- (int) changeColorWithRed: (int) r green: (int) g
blue: (int) b
โดยเราจะเรียกmethod แบบนี้:
[myColor changeColorWithRed:5 green:2 blue:6];
```

คอมไพเลอร์ภาษาอ็อบเจกทีฟ-ซีแต่ละตัวก็จะกำหนดชื่อ method เป็นการภายในแตกต่างกันไป เช่นถ้า method -changeColorWithRed:green:blue: เป็น method แบบ instance ของ class Colorคอมไพเลอร์ก็จะอาจจำ method นี้ในชื่อ `_i_Color_changeColorWithRed_green_blue` เป็นต้น โดยตัว `i` จะบอกว่าคุณนี่คืออิมพลีเม้นเทชัน method แบบ instance และตามด้วยชื่อคลาสและชื่อ method และมีการเปลี่ยนเครื่องหมาย : เป็น `_` ดังนี้แล้ว เมื่อพารามิเตอร์เป็นส่วนหนึ่งของชื่อ method ลำดับของพารามิเตอร์ในภาษาอ็อบเจกทีฟ-ซีจึงสลับที่กันไม่ได้

อย่างไรก็ตาม เราแทบไม่ได้ใช้ชื่อภายในของฟังก์ชันโดยตรง โดยทั่วไป การส่ง message จะถูกเปลี่ยนให้อยู่ในรูปแบบของการเรียกฟังก์ชันที่กำหนดไว้ในไลบรารีส่วน run-time และปรกติเวลาเราส่ง message เราก็จะไม่คำนึงถึง class ที่แท้จริงของส่วน receiver (myColor) อยู่แล้ว นั่นคือการจัดการหา method จะเป็นหน้าที่ของ run-time

## 3.4 ภาษาอ็อบเจกทีฟ-ซี 2.0

เป็นส่วนเพิ่มเติม syntax จากภาษา Objective-C โดยบริษัท Apple เป็นผู้พัฒนาเพิ่ม เช่น

- 1) Garbage collection (computer science) (ทั้งนี้ ใน Objective-C runtime ของ GNU สามารถใช้งาน Boehm-Demers-Weiser conservative garbage collector ได้ก่อนหน้า Objective-C 2.0 แล้ว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) Properties (@property) ช่วยจัดการการประกาศ instance variable
- 3) Fast enumeration การเพิ่มประสิทธิภาพในส่วน runtime

### โปรแกรม 3.9 ภาษาอ็อบเจกทีฟ-ซี 2.0

```
for (int i=0; i<[thePeople count]; i++) {
    Person *p = [thePeople objectAtIndex:i];
    NSLog(@"%@ is %i years old.", [p getName], [p
getAge]) ;
}
for (Person *p in thePeople)
    NSLog(@"%@ is %i years old.", [p getName], [p
getAge]);
```

### 3.5 ภาษาอ็อบเจกทีฟ-ซี พลัสพลัส

เป็นส่วนเพิ่มภาษาที่ช่วยให้รวม code Objective-C และ C++ เข้าด้วยกันง่ายขึ้น

### 3.6 เปรียบเทียบภาษาที่ใช้เขียนโปรแกรมบนไอโฟน

#### 3.6.1 ภาษา C

ข้อดีของภาษา C คือ การหาหนังสือเกี่ยวกับการเขียน โปรแกรมภาษา C ง่าย และ ไม่ใช่แค่หนังสือแต่รวมไปถึง Source Code, Library, Tutorial และยังมี Community ที่ใหญ่มาก และมันเป็นภาษาที่นิยมแพร่หลายและเก่าแก่ และในปัจจุบันก็ยังนิยมใช้อยู่

ข้อเสียที่ของภาษา C คือ การเขียน โปรแกรมด้วยภาษา C ให้ได้ด้นนั้นค่อนข้างยาก เป็นเพราะว่าตัวภาษาค่อนข้างจะมีรายละเอียดปลีกย่อยเยอะ แต่ด้วยความสามารถของตัวภาษานั้น สามารถลงลึกถึงขั้นจัดการหน่วยความจำโดยตรงได้ และนี่ก็อาจจะเป็นอีกข้อเสียหนึ่ง เพราะถ้าเขียนโปรแกรมให้จัดการหน่วยความจำไม่ดี โปรแกรมเราก็ทำงานได้ประสิทธิภาพที่ไม่ดีเอามากๆ และอาจจะไปทำให้ระบบรวนได้ (สำหรับภาษา Java หรือภาษาอื่น ๆ นั้น แทบจะไม่ต้องไปทำเขียนอะไรเกี่ยวกับหน่วยความจำเลย เพราะมี Garbage Collector เอาไว้จัดการหน่วยความจำให้เรียบร้อย)

การเขียน โปรแกรมด้วย ภาษา C นั้นจำเป็นต้องพึ่งภาษา Objective-C อยู่ดีเพราะต้องอาศัย Interface Builder ในการทำ User Interface แต่อย่างที่บอกไว้ข้างต้นว่า Library ของภาษา C นั้นมีเยอะมาก สามารถนำ Library ของภาษาที่มีอยู่แล้วนำมาใช้ร่วมกันกับ ภาษา Objective-C ได้ อย่างไม่มีปัญหา

#### 3.6.2 ภาษา Objective-C

ข้อดีของภาษา Objective-C คือ ค่อนข้างเรียนรู้ง่าย ไม่ซับซ้อน เพราะว่าเป็นเหมือนเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษา C ผสมกับ Smalltalk (เป็นภาษาโปรแกรมเชิงวัตถุมีกระบวนการจำแนกชนิดในภาษาโปรแกรมกระบวนการจำแนกชนิดแบบยืดหยุ่น(Dynamic))และมี Framework, Library ให้ใช้เยอะมาก ภาษามีความยืดหยุ่นสูง และเนื่องจากว่าเป็น Small Super Set of C ซึ่งหมายความว่า เราสามารถใช้ Library หรือ Source Code ของภาษา C ได้ และตัวภาษายังอนุญาตให้เราเข้าไปจัดการหน่วยความจำได้เหมือนกัน แต่ก็ไม่ยืดหยุ่นมากเท่ากับ C และมันยังมี Garbage Collector ในกรณีที่ไม่ต้องจัดการหน่วยความจำเอง (แต่ในกรณีของ iPhone Garbage Collector จะไม่สามารถใช้ได้) และข้อดีที่สุดคือ การเขียน Application บน iPhone นั้น ถ้าเขียนด้วยภาษา Objective-C ก็มันมี Interface Builder ทำให้เราออกแบบ User Interface ได้ง่ายซึ่งทาง Apple ก็ออกแบบให้ ภาษา Objective-C สนับสนุนต่อการเขียนโปรแกรมบน iPhone อีกด้วย

ข้อเสียของภาษา Objective-C คือ หนังสือที่ใช้ในการศึกษานั้นมีน้อยมาก ในทุกทางรวมไปทั้งจำนวนแหล่งความรู้เพิ่มเติมของ Objective-C ด้วย จากสิ่งนี้ทำให้ การหา Source Code, Tutorial นั้นยากตามไปด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

# เครื่องมือที่ใช้

### 4.1 แอ็กโค้ด (XCode)

แอ็กโค้ดเป็นสิ่งแวดล้อมสำหรับการพัฒนาแบบเบ็ดเสร็จ(Integrated Development Environment application) ที่ใช้เขียนโปรแกรมบนระบบปฏิบัติการแม็คโอเอส(Mac OS)และไอโอเอสถ้าจะเปรียบเทียบก็เหมือนกับการใช้วิชวล สตูดิโอ(Visual Studio)เขียนโปรแกรมบนระบบปฏิบัติการวินโดวส์ ในชุดจะประกอบด้วยโปรเจกต์ ออแกไนเซอร์(Project organizers), อีดิเตอร์ (Editors), ยูทิลิตี้ (utilities), ดีบั๊กเกอร์ (debuggers), ชุดพัฒนาโปรแกรม (software development kits), และเอกสารช่วยเหลือ

สำหรับการพัฒนาแอปพลิเคชันบนไอโฟนจะมีส่วน โปรแกรมหลักที่ใช้ในการพัฒนาเบื้องต้น 3 ตัวด้วยกันนั่นก็คือ แอ็กโค้ดใช้สำหรับเขียนโปรแกรม, อินเตอร์เฟซ บิวเดอร์(Interface Builder)ใช้สำหรับการออกแบบ หน้าตาโปรแกรมและไอโฟน ซิมูเลเตอร์(iPhone Simulator)ใช้สำหรับทดสอบโปรแกรมที่เราได้ทำการเขียนขึ้นมา

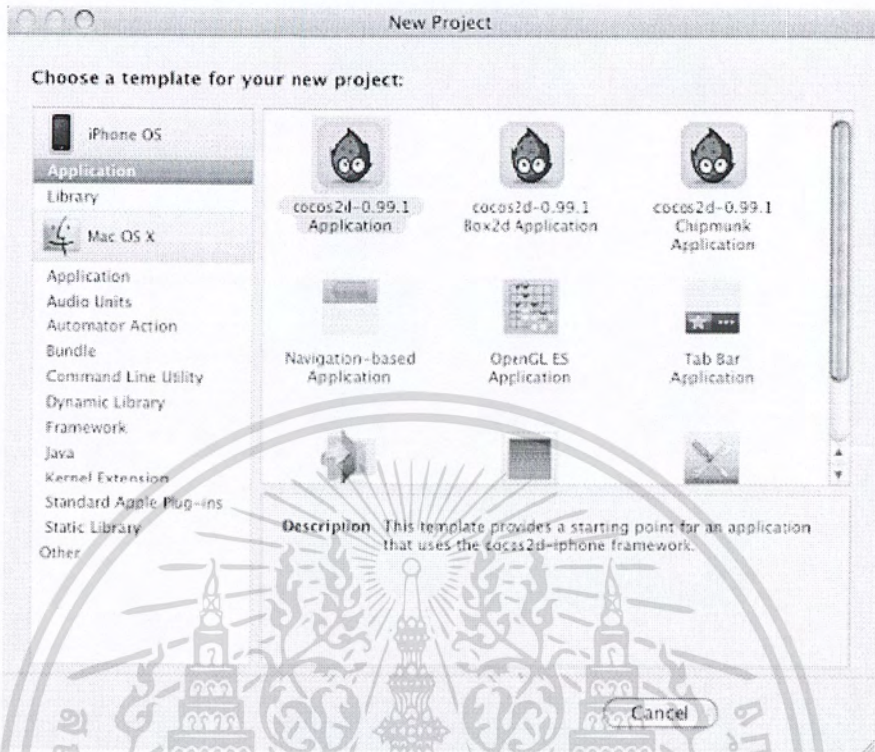


รูป 4.1 โปรแกรมที่ใช้สำหรับการพัฒนาแอปพลิเคชันบนไอโอเอส

โดยจะสามารถหาแอ็กโค้ดได้จากชุดไอโฟน เอสดีเค(iPhone SDK) ซึ่งสามารถดาวน์โหลดได้ที่ <http://developer.apple.com> เครื่องคอมพิวเตอร์ที่สามารถติดตั้งไอโฟนเอสดีเคได้นั้นจะต้องมีคุณสมบัติดังต่อไปนี้คือ เครื่องคอมพิวเตอร์จะต้องมีโปรเซสเซอร์ในสถาปัตยกรรม x86และระบบปฏิบัติการจะต้องเป็นโอเอสทีเอ็น เลฟเพิร์ด10.5.3 (OS X Leopard10.5.3) เป็นขั้นต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

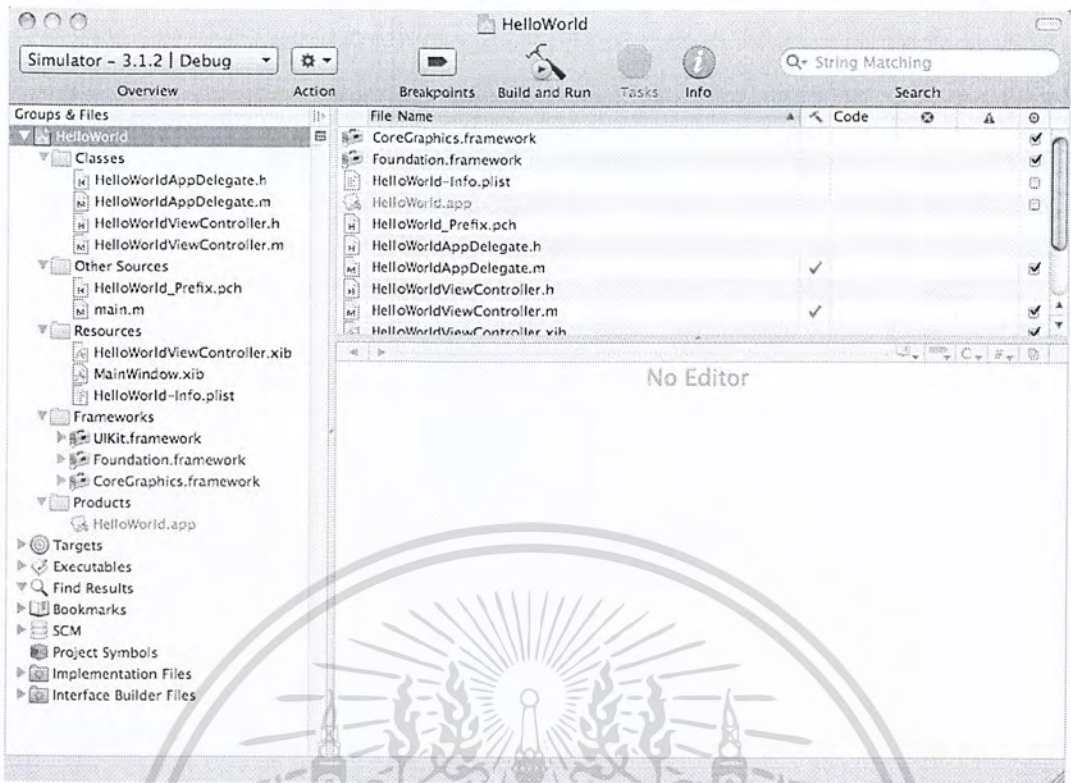
#### 4.1.1 โครงสร้างของโปรเจกต์โปรแกรมไอโฟนบนเอ็กซ์โค้ด



รูป 4.2 เลือกประเภทรูปแบบที่จะสร้าง

เมื่อผู้พัฒนาสร้างโปรเจกต์โดยผ่านโปรแกรมเอ็กซ์โค้ดผู้พัฒนาโปรแกรมสามารถพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแม็ก โอเอสหรือ ไอ โอเอส โดยการเลือกรูปแบบและเครื่องที่จะพัฒนา เมื่อเลือกได้แล้ว โปรเจกต์ก็จะถูกสร้างขึ้นดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.3 ตัวอย่างโปรเจกต์ที่อยู่ระหว่างการสร้างด้วย XCode

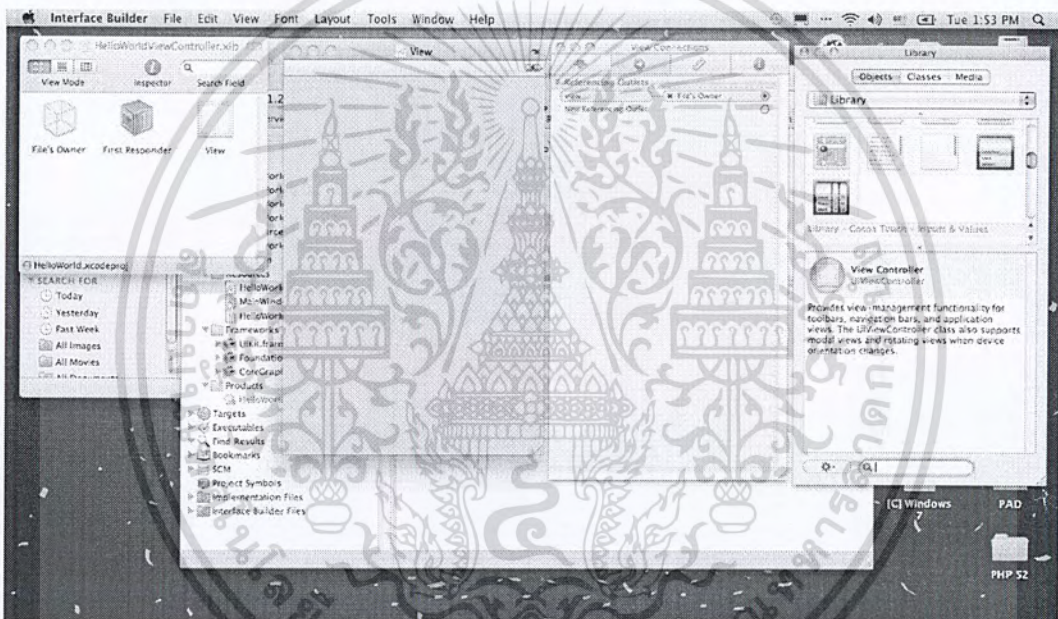
ด้านซ้ายจะเป็น Group & File Panel แสดงส่วนประกอบทั้งหมด โฟลเดอร์ที่อยู่ในนี้อาจมีพาร์ชไม่ตรงกับโฟลเดอร์ หรือไฟล์ที่เก็บของมูลจริงในเครื่อง เนื่องจากอีกโค้ดมีการจัดเอาไฟล์มาไว้ในโปรเจกต์ด้วยกันเพื่อให้ทำงานได้ง่าย รายละเอียดมีดังต่อไปนี้

- 1) Classes เป็นที่เก็บโค้ดที่เขียนด้วยภาษาอ็อบเจกทีฟ-ซีทั้งหมด โดยที่เราสามารถสร้างโฟลเดอร์ย่อยเพื่อเก็บโค้ดได้
- 2) Other Sources เป็นที่เก็บโค้ดที่ไม่ได้เขียนด้วยภาษาอ็อบเจกทีฟ-ซี โดยโปรแกรมที่เขียนสำหรับไอโฟนจะประกอบไปด้วย HelloWorld\_Prefix.pch (pch = precompiled header) ใช้สำหรับเก็บไฟล์เฮดเดอร์(header file)ที่ถูกนำมาใช้ในโปรเจกต์นี้ โดยอีกโค้ดจะทำการคอมไพล์ไฟล์เหล่านี้ไว้ก่อนเพื่อลดเวลาในการคอมไพล์เมื่อเราใช้คำสั่ง build หรือ build and go และ main.m เป็นที่เก็บ main() ซึ่งโดยปกติเราไม่จำเป็นต้องเปลี่ยนแปลง
- 3) Resource เป็นที่เก็บไฟล์ เช่นรูปภาพ และวิดีโอ ที่เราต้องการใช้ในโปรเจกต์ทั้งหมด นอกจากนั้นยังมีโฟลเดอร์ย่อยไปอีกคือ  
Hello\_ViewController.xib เก็บข้อมูลสำหรับ อินเตอร์เฟส บิวเดอร์,  
Info.plist เก็บข้อมูลรายละเอียดของโปรแกรมที่เราสร้างขึ้น, MainWindow.xip เป็นส่วนที่เก็บ main ของ Interface Builder (nib)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงแก้ไข หรือต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 4) Frameworks เป็นที่เก็บไลบรารี (code, image, sound) ที่เราสามารถเรียกใช้จากโปรแกรมเราได้
  - 5) Product เก็บโปรแกรมที่ได้หลังจากเราทำการคอมไพล์โปรเจกต์ ถ้าเรายังไม่ได้สังคอมไพล์ ไฟล์ HelloWorld.app เป็นสีแดงในอีกโค้ดคือ เป็นไฟล์ที่ไม่พบใน disk
- ด้านขวาบนจะเป็น Detail View แสดงรายละเอียดของส่วนประกอบที่ถูกเลือก  
ด้านขวาล่างจะเป็น Editor แสดงข้อมูลของส่วนประกอบที่ถูกเลือก เพื่อแก้ไข

#### 4.1.2 โครงสร้างของโปรเจกต์โปรแกรม iPhone บน Builder



รูป 4.4 ตัวอย่างหน้าต่าง Interface Builder ขณะใช้งาน

Interface Viewer เป็นโปรแกรมใช้จัดการหน้าต่างของแอปพลิเคชันที่จะใช้พัฒนา ผู้พัฒนาสามารถเปิดผ่านโปรแกรมอีกโค้ดได้ โดยเปิดไฟล์ HelloWorldViewContrller.xib หน้าต่างด้านซ้ายสุดเป็น nib ไฟล์ของ main window ซึ่งก็คือ

HelloWorldViewContrller.xib โดยในแต่ละ nib file ประกอบไปด้วย 2 icon คือ

- 1) File's Owner เป็นอ็อบเจกต์ของ nib file ที่ถูกสร้างขึ้นเมื่อเรียกใช้
- 2) First Responder เป็นอ็อบเจกต์ที่ user กำลังใช้งานอยู่ เช่นถ้า user กำลังกรอกข้อมูล text field ก็คือ first responder และเมื่อ user เลือกไปที่ส่วนอื่นของหน้าจอ first responder ก็จะเปลี่ยนตามไปด้วย

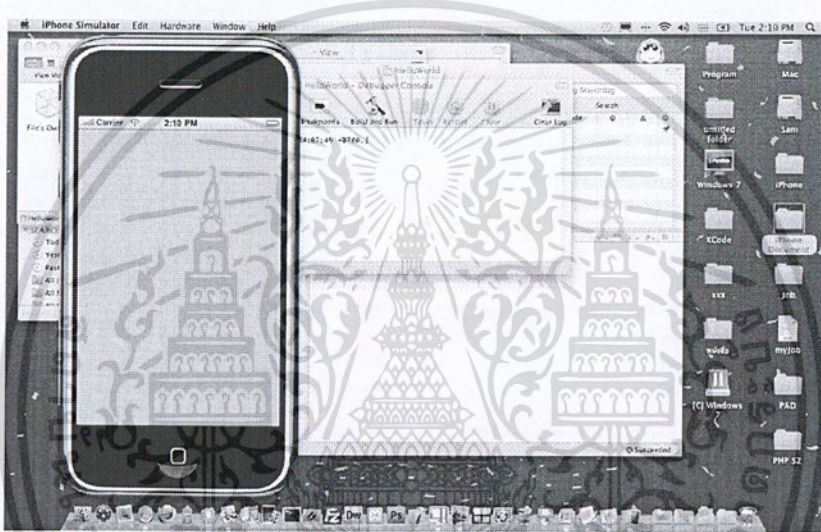
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ไฟล์อื่นๆ จะเป็น object instant ที่ถูกสร้างเมื่อ nib file ถูกโหลดขึ้นมา หน้าต่างที่สองจากซ้าย คือส่วนที่ไว้ออกแบบหน้าจอ เราสามารถเปิดหน้าต่างนี้ขึ้นมาได้โดยการกด icon view จากหน้าต่างซ้าย

หน้าต่างที่สามจากซ้าย คือ หน้าต่างสำหรับจัดการไลบรารีคุณสมบัติต่างๆ ของไลบรารีนั้นๆ

หน้าต่างขวาสุดคือ library เป็นที่เก็บ Cocoa Touch object ทั้งหมด ที่เราสามารถลากเข้ามาเป็นส่วนหนึ่งในหน้าจอ view ได้

#### 4.1.3 ทดสอบโปรแกรมบนไอโฟนด้วยไอโฟน ซิมูเลเตอร์



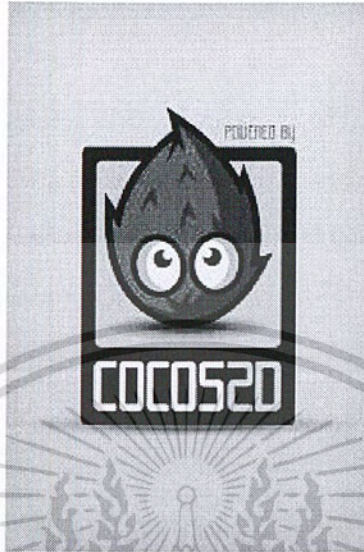
รูป 4.5 ทดสอบโปรแกรมที่เขียนด้วยไอโฟน ซิมูเลเตอร์

โดยเราสามารถที่จะเรียกใช้งานได้จาก โปรแกรมเอ็กซ์โค้ด โดยทำการคลิกที่ Icon -> Build and Run และได้ผลดังรูป 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 เกม เอนจิน (Game Engine)

### 4.2.1 Cocos2d for iPhone



รูป 4.6 โลโก้ของ cocos2d for iPhone

เป็นโอเพ่นซอร์สใช้สำหรับเขียนเกมสองมิติบน ไอโฟน โกโก้ (Cocos) เป็นที่สนใจในหมู่ผู้ที่เริ่มต้นเป็นนักพัฒนาเกมบน ไอโฟน ใช้จัดการซีน (Scene), เลเยอร์ (layer) และการโหลดรูปมาใช้ โดยจะใช้อ็อบเจกทีฟ-ซีในการเขียนโปรแกรม ในปัจจุบันเกมส์สองมิติบน ไอโฟนจำนวนมากก็ใช้โกโก้เป็นตัวพัฒนา โดยมีลักษณะเด่นของดังนี้

- 1) จัดการ Scene (แบบ workflow)
- 2) การเปลี่ยนระหว่าง Scenes
- 3) Sprites และ Sprite Sheets
- 4) ใส่ Effects เช่น เสมือนมองผ่านเลนส์, ทำฉากกระเพื่อม, ทำให้ฉากเป็นลูกคลื่น, เสมือนมองผ่านของเหลว, ทำให้ภาพหมุน และอื่นๆ
- 5) สามารถกำหนดการเปลี่ยนรูป ลำดับการจัดเรียง และ Misc actions
- 6) เมนูและ button ง่ายๆ
- 7) รวม Box2d and Chipmunk physics engine
- 8) Particle system
- 9) สนับสนุน render Text
- 10) สนับสนุน Texture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 11) สนับสนุนการสร้างแผนที่
- 12) สนับสนุน Parallax scrolling
- 13) สนับสนุน Streak Motion
- 14) High Score server
- 15) สนับสนุนระบบTouch/Accelerometer
- 16) สนับสนุนระบบเสียง
- 17) โหมดรูปและระบบภูมิประเทศ
- 18) รวมระบบ Pause/Resume
- 19) สนับสนุน PowerVR Texture Compression (PVRTC) format
- 20) ภาษาที่ใช้เขียนคือ objective-c

หากผู้พัฒนาต้องการพัฒนาด้วย Cocos2d for iphone ต้องพัฒนาบนเครื่องแม็ค ไอเอสที่มีเอ็ค โค้ดและไอโฟน เอสดีเค



รูป 4.7 ตัวอย่างฉากในเกมฮันนี่ ฟาร์ม

ตัวอย่างเกมสที่ใช้ Cocos2d for iphone เช่นฮันนี่ ฟาร์ม(Honey Farm) ซึ่งเป็นเกมของดิจิตโพลิสต์สตูดิโอ(digitopolisstudio) ซึ่งเป็นบริษัทเกมของคนไทย เกมก็คือควบคุมการบินของผึ้งสีต่างๆ ให้บินเข้ารังตามสีของผึ้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.2 iTORQUE 2D for iphone

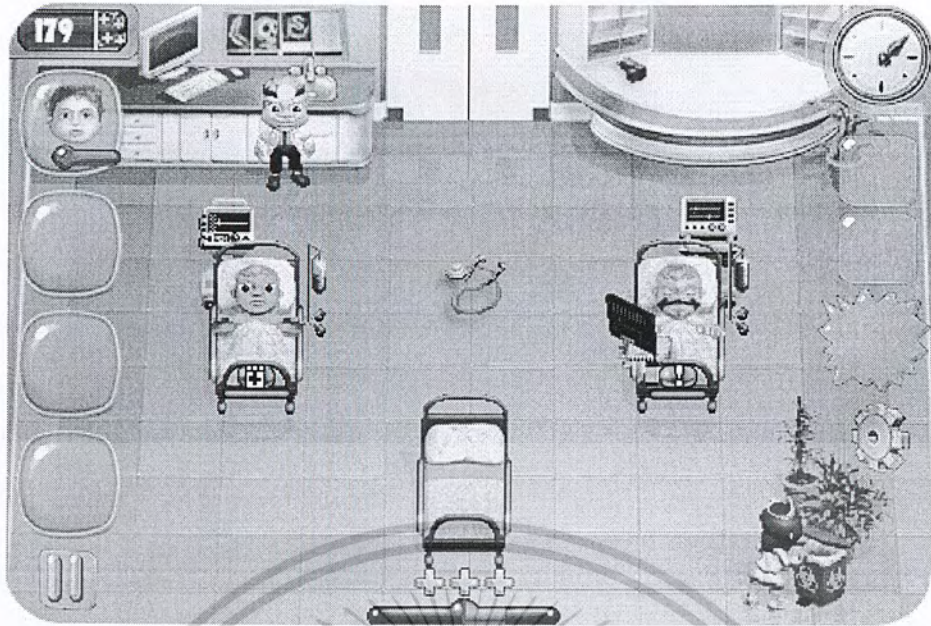


รูป 4.8 ตัวอย่างโลโก้ของ iTORQUE

เป็นเกม เอนจินที่ใช้ในทางการค้า ใช้สำหรับเขียนเกมสองมิติบน ไอโฟน เป็นตัวอิดิตเตอร์ที่ใช้ในการจำลองฉากที่ตีความ มีจำนวนลักษณะเด่นที่มาก โดยตัว iTORQUE มาจาก Torque engine เพื่อแปลงลง ไอโฟน โดยเฉพาะ จากเอนจินตัวเดียวกันจากผู้สร้างเดียวกัน ทำให้ใช้ตัว Torque Game Builder ได้ในการสร้างเกม ราคาปัจจุบัน (28 มีนาคม 2553) 750 ดอลลาร์ หากจะพูดถึง ลักษณะเด่นของ iTORQUE for iphone มีดังนี้

- 1) วอทยูซีอีสวอทยูเก็ต พูดี อิดิตเตอร์ (WYSIWYG 2D editor) จะแสดงผลแบบเดียวกันบนหน้าจอ ไอโฟน
- 2) ใช้ภาษาสคริปความสามารถสูงที่สามารถประมวลผลซ้ำๆ ได้อย่างเหมาะสม
- 3) สนับสนุนการป้อนแบบมัลติทัชและการจับการเคลื่อนไหว
- 4) PowerVR Texture Compression
- 5) มีsource ทั้งหมดประกอบอยู่ด้วยเพื่อใช้ร่วมกับ middleware อื่นๆ
- 6) ลักษณะและเครื่องมือจะสอดคล้องกับ Torque Game Builder

หากผู้พัฒนาต้องการพัฒนาด้วย iTORQUE 2D for iphone ต้องพัฒนาบนระบบปฏิบัติการแมค โอเอสเทน เลฟเฟิร์ด โดยมี ไอโฟน เอสดีเคด้วย



รูป 4.9 ตัวอย่างฉากในHospital Havoc

ตัวอย่างเกมที่ใช้ iTORQUE for iphone เช่น Hospital Havoc ซึ่งเป็นเกมจำลองสถานการณ์โรงพยาบาล ผู้เล่นจะต้องจัดการหมอและพยาบาลให้ดูแลคนไข้ที่มีอยู่

#### 4.2.3 Oolong ( 乌龙 หรือ อุหลง)

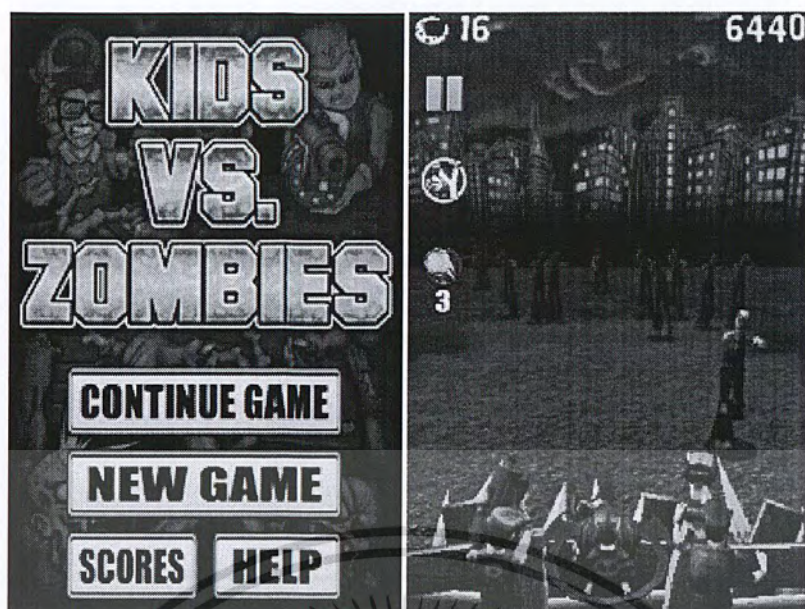
รูป 4.10 ตัวอย่างโลโก้ของiTORQUE

อุหลงเป็นโอเพ่นซอร์ส ที่เป็นแอนจินสามมิติ ถูกเขียนในภาษาซีพลัสพลัส และมีประสิทธิภาพที่ดี อุหลงประกอบด้วยคุณสมบัติเฉพาะของเอ็มไอที ไลเซนส์ (MIT license) อาจกล่าวได้ว่าแอนจินนี้ถูกสร้างสำหรับโปรแกรมเมอร์และต้องการความรู้ของโอเพ่นจีแอล อีเอส (OpenGL ES) ในการใช้ อย่างไรก็ตามอุหลงเป็นเกมแอนจินระดับสูง ผู้เริ่มต้นจึงควรหลีกเลี่ยงเพราะใช้งานยากโดยลักษณะเด่นของอุหลงมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) รองรับโอเพ่นจีแอล อีเอส 1.1
- 2) Math library รองรับการคำนวณแบบ floating-point ซึ่งมี interface คล้ายๆ กับ Direct 3D math library มาก
- 3) รองรับ texture format หลายแบบ ได้แก่ PowerVR 2-bit, 4-bit และ normal map compression formats
- 4) รองรับ PowerVR's POD (Scene and Meshes) และฟอร์แมตไฟล์ 3DS
- 5) รองรับหน้าจอสัมผัส
- 6) รองรับระบบตรวจจับความเคลื่อนไหว
- 7) รองรับ Text rendering และ basic UI
- 8) Timing: ฟังก์ชันบางตัวที่จะมาแทนที่ rdstc, QueryPerformance, etc.
- 9) Profiler: industry proven in-game profiler
- 10) มี Resources streaming system
- 11) ระบบฟิสิกส์สามมิติ Bullet SDK
- 12) ระบบเสียงรองรับ OpenAL
- 13) Networking library based on ENet
- 14) ระบบจัดการหน่วยความจำระดับอุตสาหกรรม จาก Fluid Studios

หากผู้พัฒนาต้องการพัฒนาด้วยอยู่หลังจะต้องใช้เครื่องที่ใช้แม็คโอเอสเห็น เลฟเฟิร์ดที่ใช้สถาปัตยกรรมของส่วนประมวลผลกลางแบบ x86 เท่านั้น และมีไอโฟนหรือไอพอดทัชที่ใช้ไอโอเอส2ขึ้นไป



รูป 4.11 ตัวอย่างฉากใน Kids VS. Zombies

ตัวอย่างเกมที่ใช้อู๋หลง เช่น Kids VS. Zombies หรือ Zombie Mansion ซึ่ง Kids VS. Zombies เป็นเกมที่ให้เด็กสามคนตั้งป้อมปืนสู้กับซอมบี้ ส่วน Zombie Mansion เป็นเกมยิงมุมมองบุคคลที่หนึ่ง เข้าไปยังสัตว์ประหลาดในสถานที่ต่างๆ



รูป 4.12 ตัวอย่างฉากใน Zombie Mansion

ซึ่งทางกลุ่มได้เลือก Cocos2d for iPhone เพราะเป็น Game engine ที่มีความนิยม ง่ายต่อการศึกษา เหมาะสำหรับการเริ่มต้น มีแหล่งชุมชนพร้อมช่วยเหลือ ไม่มีค่าใช้จ่ายในการขอใช้ Game engine อีกทั้ง iPhone นั้นสนับสนุน 2 มิติ มากกว่า 3 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

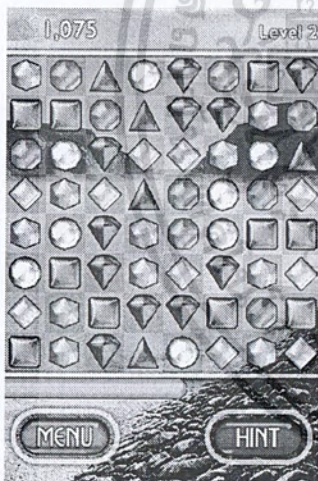
## บทที่ 5

### ระบบเกม

#### 5.1 เกมบนไอโฟน

เกมส่วนใหญ่จะใช้คุณสมบัติของมัลติทัชและระบบจับความเคลื่อนไหวเป็นหลัก ตัวเกมมีหลายประเภท แยกได้คร่าวๆดังนี้

- 1) พัชเชิล (Puzzle) เป็นรูปแบบเกมที่ใช้ชาวปัญญาในการปัญหาต่างๆในเกม เช่นเกมบีจีเวล (Bejeweled) จะใช้การลากนิ้วไปตามจอ เพื่อย้ายตำแหน่งวัตถุ คล้ายๆกับที่ใช้นิ้วแทนเมาส์บนคอมพิวเตอร์ โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเอง หรือเจ้านกขี้โมโห (Angry Birds) วิธีการเล่นนั้นคือลากเจ้านกพวกนี้ ยิ่งออกไปจากหนังสือตึก เพื่อไปทำลายกลุ่มหมูหน้าเขี้ยว โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเองและระบบฟิสิกส์ใช้ของ Box2D ในการพัฒนา



ก)



ข)

รูป 5.1 เกมแบบพัชเชิล

ก) เกมบีจีเวล

ข) เจ้านกขี้โมโห

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) เรซซิ่ง (Racing) เช่น นีด ฟอ์ สปีด (Need for Speed) หลักๆเลยคือจะใช้การเอียงเครื่องเพื่อเลี้ยวซ้ายขวา บางเกมจะทำการเร่งเครื่องให้เราเอง เราทำได้แค่เลี้ยวและเบรกเท่านั้น แต่บางเกมก็มีปุ่มเสมือนขึ้นมาบนจอเพื่อบังคับคันเร่งและเบรก และในบางเกมก็จะมีการใช้มัลติทัชในการช่วยใช้ฟังก์ชันพิเศษ เช่น การเปิดไนตรัสเครื่องเป็นต้น โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเอง



รูป 5.2 เกม นีด ฟอ์ สปีด

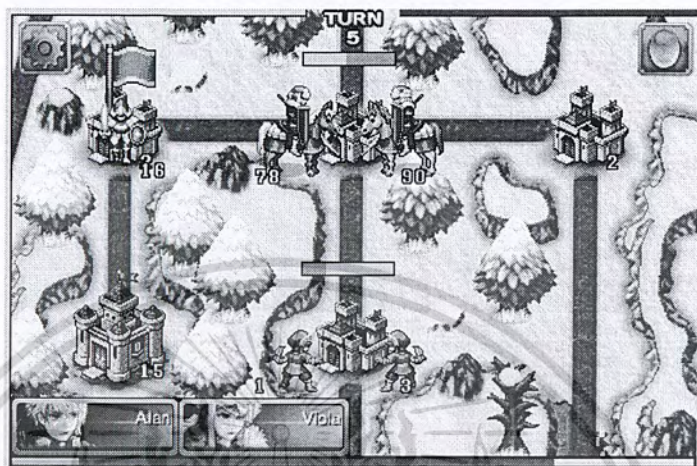
- 3) อาร์ทีเอส (RTS - Real Time Strategy) เช่น เรด อเลิร์ต (Red Alert) จะใช้ทัชสกรีนแทนการใช้เมาส์บนคอมพิวเตอร์ การควบคุมอื่นๆจะเหมือนบนคอมพิวเตอร์ โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเอง



รูป 5.3 เกม เรด อเลิร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 4) ทีบีเอส (TBS - Turn-based Strategy) เช่น ฮิลล์ แอนด์ ริเวอร์ (Hills and Rivers) จะมีการควบคุมคล้ายอาร์ทีเอสคือใช้ทัชสกรีนแทนเมาส์ โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเอง



รูป 5.4 เกมฮิลล์ แอนด์ ริเวอร์

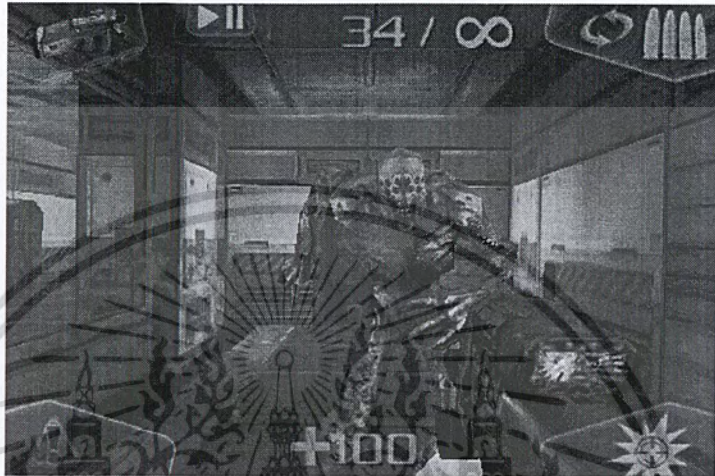
- 5) อาร์พีจี (RPG- Role Playing Game) เช่น แควออส ริง (Chaos Ring) จะใช้ทัชสกรีนแทนการใช้เมาส์ในการเลือกสิ่งของและมีปุ่มเสมือนขึ้นมาบนจอเพื่อควบคุมการเดินของตัวละคร ในบางเกมอาจมีการใช้ระบบจับความเคลื่อนไหวเข้าร่วมในมินิเกม โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเอง



รูป 5.5 เกมแควออส ริง

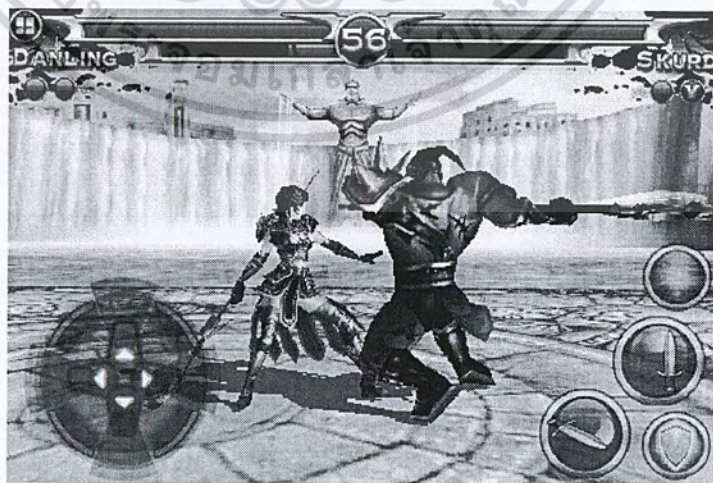
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 6) เอฟพีเอส (FPS- First Person Shooting) เช่น ดอม (Doom) ส่วนใหญ่จะไม่สามารถบังคับทิศทางเดินเองได้ บังคับได้แค่การเล็งปืนเท่านั้น ซึ่งจะบังคับโดยการเอียงเครื่องขึ้นลงซ้ายขวา (เป็นการใช้ระบบจับความเคลื่อนไหว) โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเอง



รูป 5.6 เกมดอม

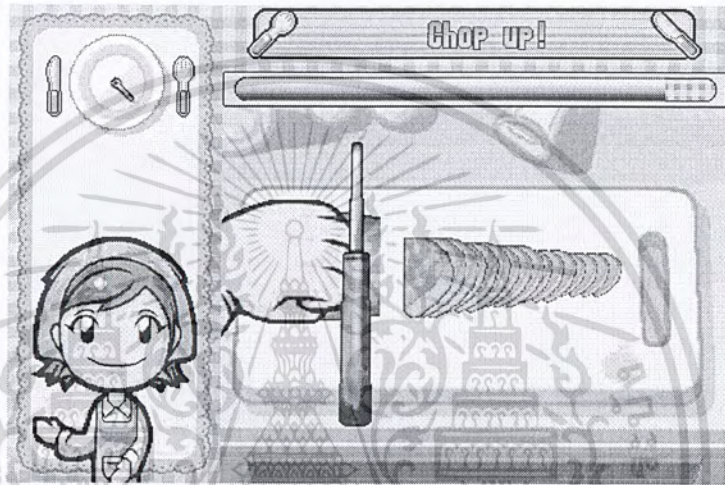
- 7) ไฟต์ติ้ง (Fighting) เช่น เกมเบลดออฟเฟอรี (Blades of Fury) จะมีปุ่มเสมือนขึ้นมาบนจอ ลักษณะเหมือนปุ่มบนเครื่องเกมทั่วไป ด้านซ้ายเป็นปุ่มควบคุมทิศทาง ด้านขวาเป็นปุ่มเพื่อกระทำ โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเอง



รูป 5.7 เกมเบลดออฟเฟอรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 8) คุกกิ้ง (Cooking) เกมแนวนี้เป็น 1 ในเกมที่ไม่พุดถึงไม่ได้ ถึงแม้ว่าจะมีตั้งอยู่เพียงแค่เกมเดียวคือคุกกิ้ง มาม่า (Cooking Mama) แต่ว่าได้สร้างกระแสให้ไอโฟนมากทีเดียว เกมนี้จะเน้นในการทำอาหาร โดยจำลองการทำอาหารจริงๆมา เช่นเวลาหั่นของ ก็ลูปนิ้วไปบนจอเหมือนกับว่าหั่นจริงๆ เหน้าก็เอียงเครื่องเหมือนว่าหน้าออกจากเครื่องจริงๆ หรือการผัด ก็กระดกเครื่องเหมือนกว่ากระดกกระทะจริงๆ โดยเกมนี้จะใช้เกมเอนจินที่ได้พัฒนาขึ้นเอง



รูป 5.8 เกมคุกกิ้ง มาม่า

หากเราจำแนกเกมของ ไอ โฟนออกตามเทคโนโลยีที่ใช้ในเกม จะได้เกมหลายประเภท ดังนี้

- 1) เกมที่ใช้เฉพาะทัชสกรีน (ไม่ได้ใช้ความสามารถของมัลติทัช) เช่นเกมประเภทพัชเชิล ประเภทอาร์พีจีและประเภทที่บีเอส ซึ่งเกมเหล่านี้สามารถเล่นได้โดยใช้การสัมผัสหน้าจอเพียงแค่ 1 จุด ณ เวลาใดๆ จึงไม่จำเป็นต้องใช้มัลติทัชเข้าช่วย
- 2) เกมส่ที่ใช้มัลติทัชจะเป็นเกมส่ที่ผู้เล่นต้องตอบสนองต่อเกมอย่างรวดเร็ว จนอาจมีการกดหลายๆจุดบนจอพร้อมกัน ณ เวลาใดๆ เช่นการกดเดินหน้าพร้อมกับกดต่อของเกมนแนว F ไรต์ดิ่งหรือการเล่นเครื่องดนตรีเสมือนบน ไอ โฟนของเกมเกี่ยวกับดนตรีต่างๆ
- 3) เกมส่ที่ใช้ระบบจับความเคลื่อนไหวเข้าช่วย ส่วนใหญ่จะใช้เพื่อควบคุมทิศทางของสิ่งต่างๆ เช่นการเล็งกระบอกปืนในเกมแนวเอฟพีเอส การควบคุมรถยนต์ในเกมประเภทเรซซิ่งหรือแม้แต่การทำอาหารบางอย่างในเกมแนวคุกกิ้งก็ใช้ฟังก์ชันตัวนี้เช่นเดียวกัน ซึ่งเกมเหล่านี้ มักจะมีการใช้ทัชสกรีนร่วมด้วยในการสั่งการ เช่นการยิงปืน หรือการเร่งเรื่ง เบรคเครื่องนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) เกมที่ใช้จีโอ แท็กกิ้ง(Geo-Tagging)เป็นเกมใหม่ล่าสุดที่ผู้เขียนยังไม่ได้ลองเล่น แต่ได้อ่านจากสื่อต่างๆมาพอสมควร จะขออธิบายคร่าวๆคือเกมจะใช้ระบบจีพีเอส(GPS – Global Positioning System) ช่วยในการระบุตำแหน่งของผู้ใช้ และเมื่อผู้ใช้ไปยังสถานที่บางสถานที่ในชีวิตจริงตามที่เกมส์กำหนด(เช่นสั่งให้เดินทางไปยังสถานีรถไฟฟ้าว่อนนุช) ก็จะได้รับไอเท็มพิเศษมาครอบครอง ซึ่งเกมแบบนี้เริ่มนิยมในประเทศไทยญี่ปุ่น ซึ่งระบบอื่นๆของตัวเกม ก็ยังคงต้องใช้ทัชสกรีนในการควบคุมอยู่ดี

สรุปคือเกมส่วนใหญ่จะใช้ความสามารถของมัลติทัชเป็นหลัก ทำให้สามารถกดหลายๆจุดบนจอพร้อมกันได้ รองลงมาคือระบบจับความเคลื่อนไหวแต่ยังไม่พบเกมใดใช้เสียง หรือกล้องในการเล่นเกมร่วมด้วย แต่ในเรื่องของกล้องนั้น จากการติดตามข่าวใหม่ๆเกี่ยวกับไอโฟนรุ่นล่าสุดพบว่าไอโฟนรุ่นใหม่ที่จะเปิดตัวในเดือนกรกฎาคมที่ผ่านมา มีกล้องหน้ามาด้วย ซึ่งน่าจะเป็นไปได้สูงที่จะสามารถนำมาประยุกต์ใช้กับเกมได้มากกว่ากล้องที่มีอยู่เพียงด้านหลังแบบในรุ่นก่อนๆ

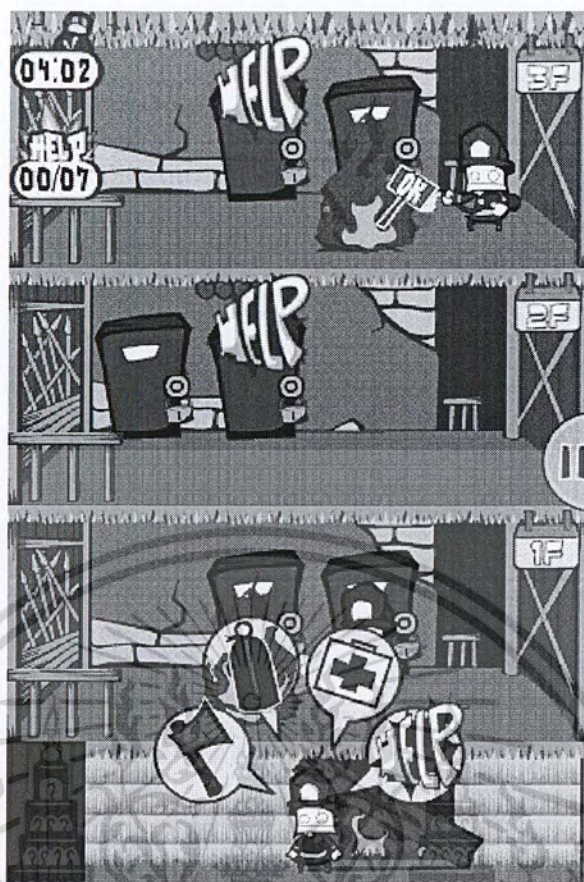
## 5.2 ตัวอย่างเกมที่พัฒนาโดยทีมพัฒนาชาวไทย

ปัจจุบันก็มีคนไทยจำนวนไม่น้อยให้ความสนใจในการพัฒนาโปรแกรมหรือเกมบนไอโฟน ทางทีมจัดทำจึงอยากนำเสนอให้เห็นตัวอย่าง เพื่อเป็นแนวทางต่อไป

- 1) อีเมอร์เจนซี แดนเจอร์ ไฟร์ ไฟต์เตอร์ (Emergency Danger Fire Fighter)

ผู้พัฒนา: <http://www.tripleplay.co.th/>

การเล่น: เราจะเป็นพนักงานดับเพลิงที่ตามดับไฟด้วยถังดับเพลิงและขวานเพื่อฟังกลอนประตู

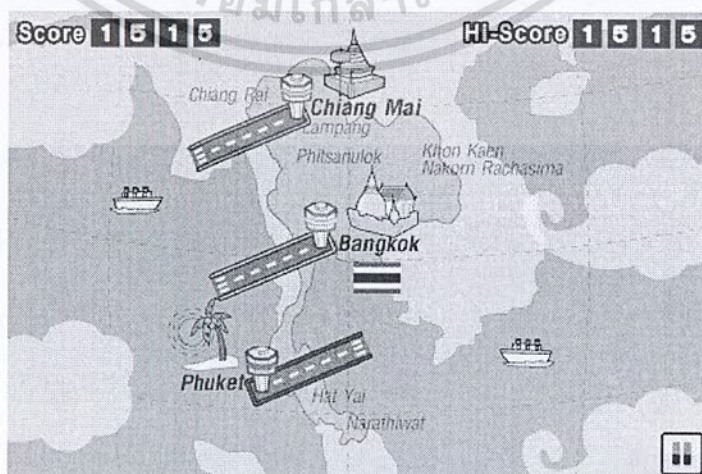


รูป 5.9 เกมอีมอร์เจนซ์ แคนเจอร์ ไฟร์ ไฟต์เตอร์

2) ลิตเติ้ลกัปตัน (Little Captain)

ผู้พัฒนา: <http://digitopolisstudio.com/>

การเล่น: ผู้เล่นจะต้องลากเครื่องบินไปจอดยังสนามบินที่ถูกต้อง โดยดูจากสีของเครื่องบิน



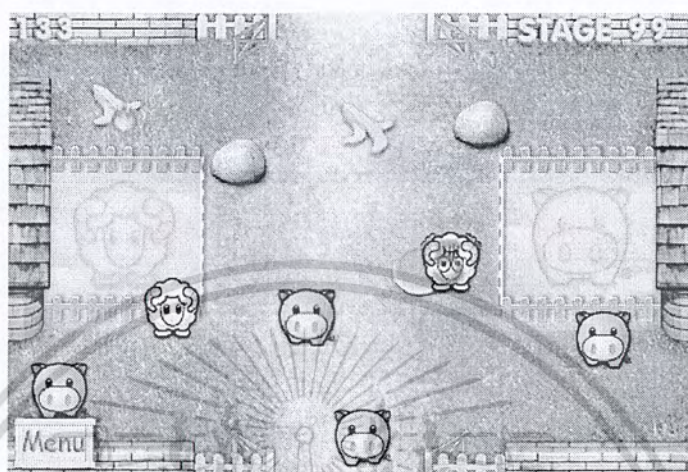
รูป 5.10 เกมลิตเติ้ลกัปตัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3) เกอร์ซี่ฟาร์ม (Crazy Farm)

ทีมผู้พัฒนา: <http://www.pigcanfly.com/?p=82>

การเล่น: เป็นเกมที่ผู้เล่นจะต้องคอยจับสัตว์เลี้ยงเข้าคอกให้ถูก

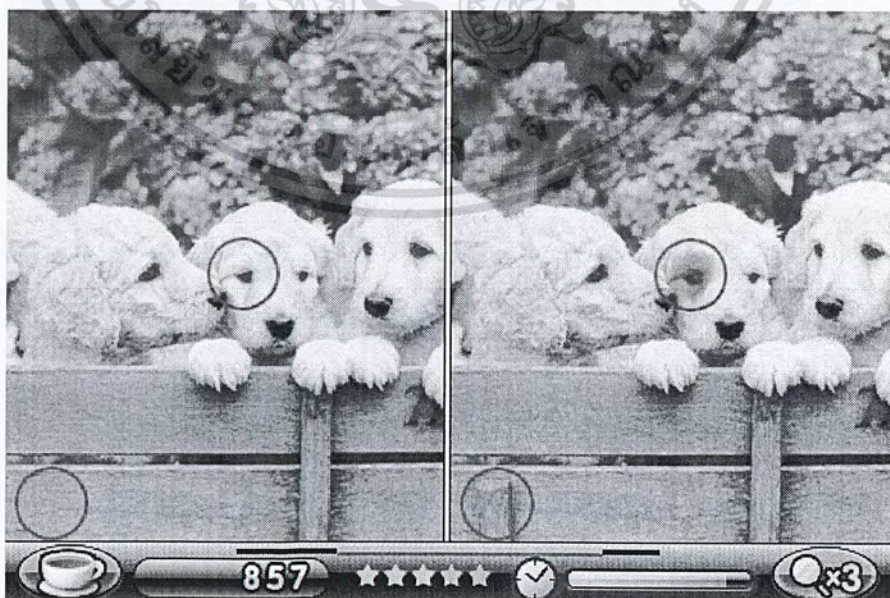


รูป 5.11 เกมเกอร์ซี่ฟาร์ม

## 4) สปอตเดอะดิฟเฟอเรนซ์ (Spot the Differences)

ผู้พัฒนา: [http://www.sanukgames.com/?page\\_id=121](http://www.sanukgames.com/?page_id=121)

การเล่น: เป็นเกมจับผิดภาพ ผู้เล่นจะต้องหาจุดที่แตกต่างกันของทั้งสองภาพให้เจอ



รูป 5.12 เกมสปอตเดอะดิฟเฟอเรนซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.3 ระบบภายในเกม

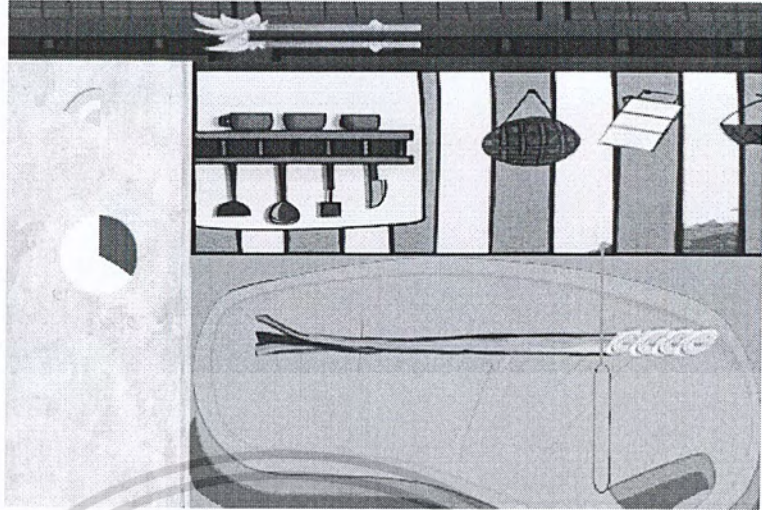
ทางเราได้เลือกที่จะทำเกมคูกิ่งซึ่งไม่ค่อยมีเกมแนวนี้มากนัก อีกทั้งยังสามารถประยุกต์เข้ากับคุณสมบัติต่างๆของไอโอเอสได้เป็นอย่างดี นอกจากนี้ยังสามารถนำเอาวัฒนธรรมไทยมาใส่ได้ อย่างไม่ดูแปลกแยกอีกด้วย โดยระบบภายในเกมจะมีคูกิ่งมา่าเป็นต้นแบบ โดยเราจะนำข้อดีมาใช้และปรับปรุงข้อเสีย พร้อมทั้งเพิ่มการเล่นรูปแบบใหม่ๆเข้าไป

### 5.3.1 ระบบการเล่นหลายคน

โดยเราจะนำเมนูคูกิ่งมาใช้ในการเล่น 2 คน โดยผู้เล่นจะทำการเล่นโดยผ่านระบบบลูทูธ (Bluetooth) ในการเล่นสองคนเราจะวัดกันว่าผู้เล่นฝ่ายใดทำขั้นตอนทั้งหมดเสร็จก่อนกัน ในระหว่างการแข่งขันจะมีวงกลมสีแดงที่เป็นตัวบอกว่าผู้เล่นฝ่ายตรงข้ามทำถึงขั้นตอนไหนแล้ว ถ้าผู้เล่นฝ่ายตรงข้ามทำถึงขั้นตอนสุดท้ายแล้ว หน้าจอของเราจะกระพริบสีแดงเพื่อทำการเตือน โดยจะมีขั้นตอนนี้

- 1) ผู้เล่นทั้งสองคนทำการเลือกเล่นในโหมด “เล่นสองคน”
- 2) ระบบเกมจะทำการตรวจดูว่าได้มีการเปิดใช้งาน บลูทูธ หรือยัง ถ้ายังไม่ได้เปิดใช้งานจะแสดงหน้าต่างขึ้นมาให้ผู้เล่น ทำการเปิดใช้งานบลูทูธ
- 3) ระบบจะทำการค้นหาเครื่องของผู้เล่นฝ่ายตรงข้าม
- 4) เมื่อทำการค้นหาเครื่องของผู้เล่นฝ่ายตรงข้ามพบแล้วจะแสดงกล่องข้อความขึ้นมา โดยภายในกล่องข้อความจะมีชื่อเครื่องของผู้เล่นฝ่ายตรงข้าม แล้วให้ผู้เล่นเลือกไปยังชื่อของผู้ที่จะทำการเล่นด้วย
- 5) เมื่อผู้เล่นทั้งสองทำการเลือกแล้ว เกมก็จะเริ่มขึ้น โดยให้ผู้เล่นทั้งสองทำขั้นตอนต่างๆให้เร็วที่สุด โดยจะมีวงกลมสีแดงที่จะคอยบอกว่า ผู้เล่นฝั่งตรงข้ามถึงขั้นไหนแล้ว
- 6) เมื่อผู้เล่นฝ่ายตรงข้ามทำถึงขั้นตอนสุดท้ายแล้ว หน้าจอของเราจะกระพริบสีแดงเพื่อทำการเตือนว่า ผู้เล่นฝั่งตรงข้ามทำขั้นตอนสุดท้ายแล้ว ให้ผู้เล่นรีบทำขั้นตอนให้เร็วขึ้น
- 7) เมื่อผู้เล่นฝ่ายใดทำขั้นตอนเสร็จแล้ว จะมีข้อความว่า “ชนะ” ขึ้นมา แล้วผู้เล่นอีกฝ่ายเมื่อทำเสร็จก็มีคำว่า “แพ้” ขึ้นมาที่หน้าจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 5.13 การเล่นหลายคนแบบแข่งกัน

#### 5.3.1.1 ข้อมูลเชิงเทคนิคของการเชื่อมต่อไอโฟนด้วยบลูทูธ

- 1) ติดต่อกับเครื่องที่อยู่ในรัศมี 10 เมตรแบบไร้สายได้
- 2) เชื่อมต่อได้สูงสุดพร้อมกัน 6 เครื่อง แต่ชุมชนผู้พัฒนาแนะนำว่าอย่าเกิน 2 เครื่อง
- 3) ใช้พลังงานน้อยกว่าการเชื่อมต่อแบบแลนไร้สาย(Wireless-Lan)

#### 5.3.1.2 ตัวอย่างเกมที่เล่นหลายคน

- 1) Tic-Tac-Touch เป็นเกมกระดานที่ให้ผู้เล่นทำการ ผลัดกันวางตัวโอหรือเอ็กซ์ลงบนกระดาน โดยสามารถที่จะเล่นกับผู้เล่นอื่นผ่านการเล่นออนไลน์



รูป 5.14 เกม Tic-Tac-Touch

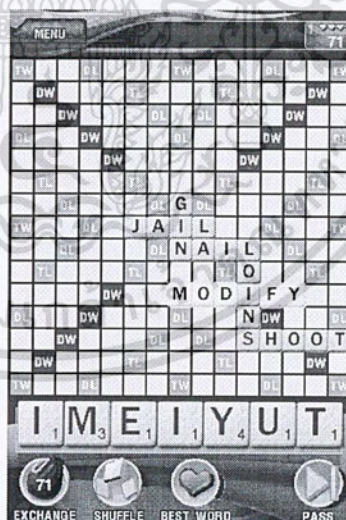
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) iPingPong 3D เป็นเกมตีปิงปอง โดยใช้การใช้นิ้วสัมผัสลงบนไม้เพื่อทำการตี โดยสามารถที่จะเล่นได้ 2 คน ผ่านระบบ wi-fi



รูป 5.15 เกม iPingPong 3D

- 3) Scrabble เป็นเกมกระดานแนวพัชเชิล ให้ผู้เล่นทำการเรียงตัวอักษรที่มีอยู่ให้เป็นคำ โดยพยายามเรียงให้ได้คะแนนมากที่สุด สามารถที่จะแข่งกับเพื่อนผ่านทางระบบ wi-fi



รูป 5.16 เกม Scrabble

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 6

# ชิ้นงานเกมไทยอ้อย

### 6.1 หน้าแรก

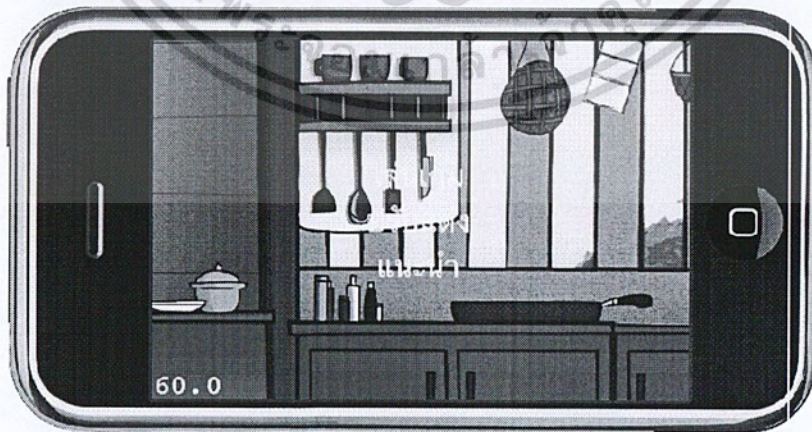
เมื่อเปิดโปรแกรมขึ้นมา จะพบกับภาพโลโก้ของ Cocos2D จากนั้นจะเป็นเมนูให้เลือกว่าจะทำการเล่นคนเดียวหรือเล่นหลายคน ดังรูป



รูป 6.1 เมนูเลือกระหว่างเล่นคนเดียวหรือเล่นหลายคน

### 6.2 เมนูหลัก

หน้าเมนูหลักจะมีให้ผู้เล่นเลือกว่าจะเริ่มเล่นเกม ตั้งค่าก่อนเล่นหรืออ่านคำแนะนำ



รูป 6.2 เมนูหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6.3 เลือกเมนูอาหาร

เมื่อผู้เล่นเลือกที่จะเล่นเกม โปรแกรมจะให้ผู้ผู้เล่นเลือกเมนูอาหาร



รูป 6.3 เลือกเมนูที่จะเล่น

### 6.4 เลือกขั้นตอนที่จะเล่น

หลังจากเลือกเมนูที่จะเล่นแล้ว ผู้เล่นก็ต้องเลือกขั้นตอนที่ต้องการ

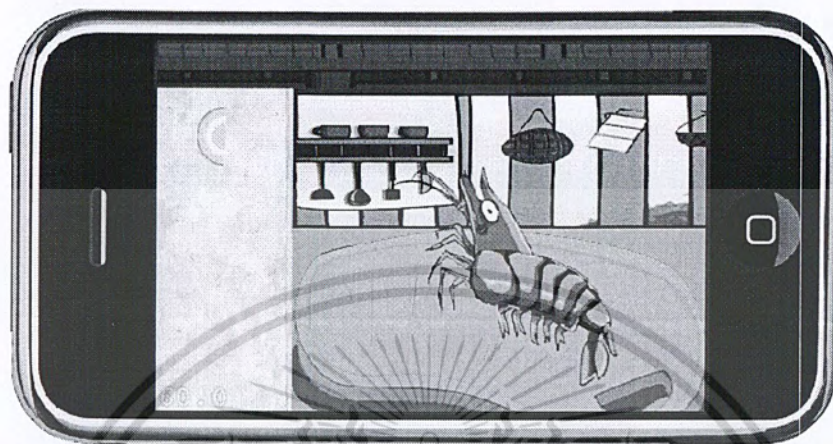


รูป 6.4 เลือกขั้นตอนที่จะเล่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6.4.1 ปอกกุ้ง

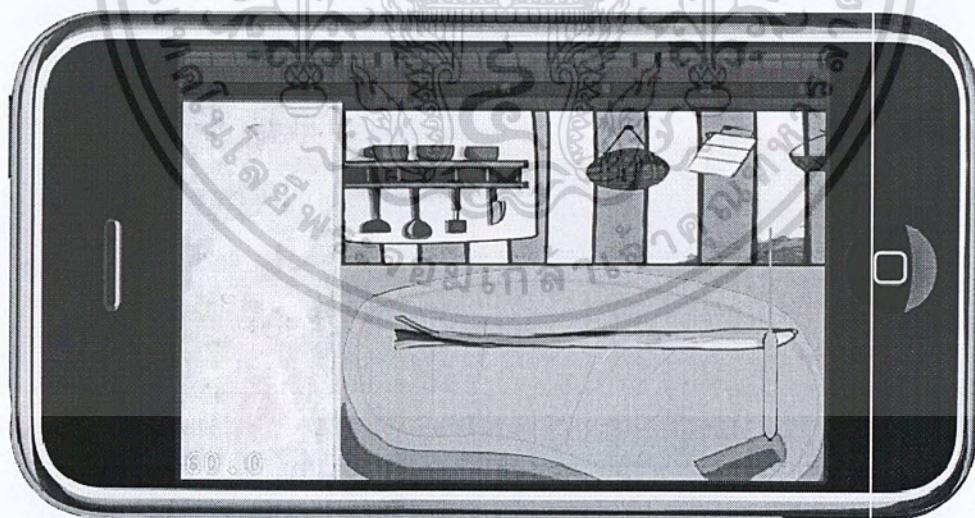
ผู้เล่นจะต้องปอกเปลือกกุ้งออกให้เสร็จทันตามเวลาที่กำหนด โดยจะเริ่มจากการปอกหัวกุ้ง เปลือกกุ้ง และขากุ้งตามลำดับ



รูป 6.5 หน้าจอการเล่นขั้นตอนปอกกุ้ง

### 6.4.2 หั่นตะไคร้

ผู้เล่นจะต้องเลื่อนมีดขึ้น-ลง ไปเรื่อยๆ เพื่อหั่นตะไคร้ให้เสร็จทันเวลาที่กำหนด

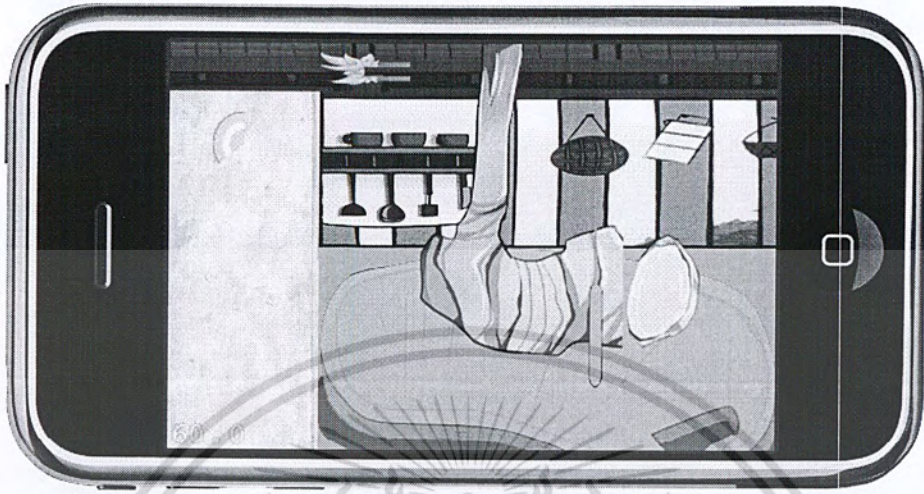


รูป 6.6 หน้าจอการเล่นขั้นตอนหั่นตะไคร้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6.4.3 หั่นข่า

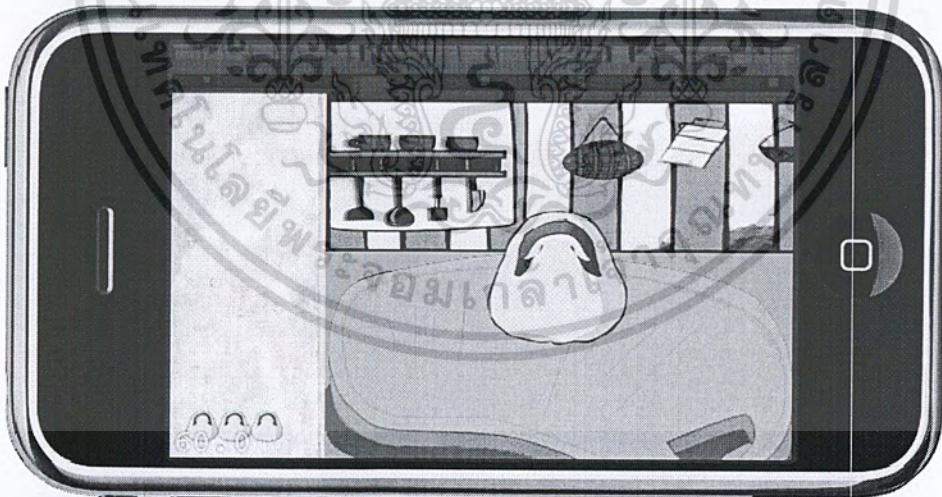
ผู้เล่นจะต้องกดมิกไปบนข่าเพื่อหั่นข่าให้ทันเวลาที่กำหนด



รูป 6.7 หน้าจอการเล่นขั้นตอนหั่นข่า

### 6.4.4 หั่นเห็ด

ผู้เล่นจะต้องลากนิ้วลงจากยอดเห็ดจนถึงโคนเห็ดเพื่อหั่นเห็ด

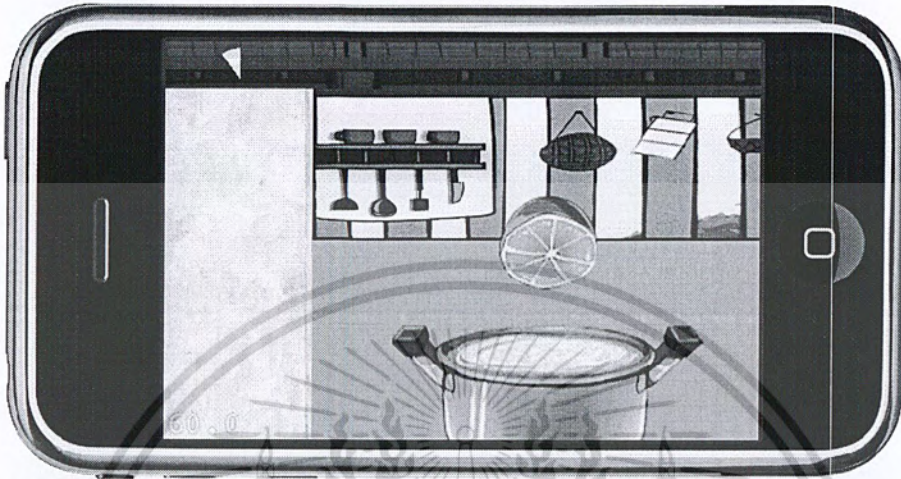


รูป 6.8 หน้าจอการเล่นขั้นตอนหั่นเห็ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 6.4.5 บีบมะนาว

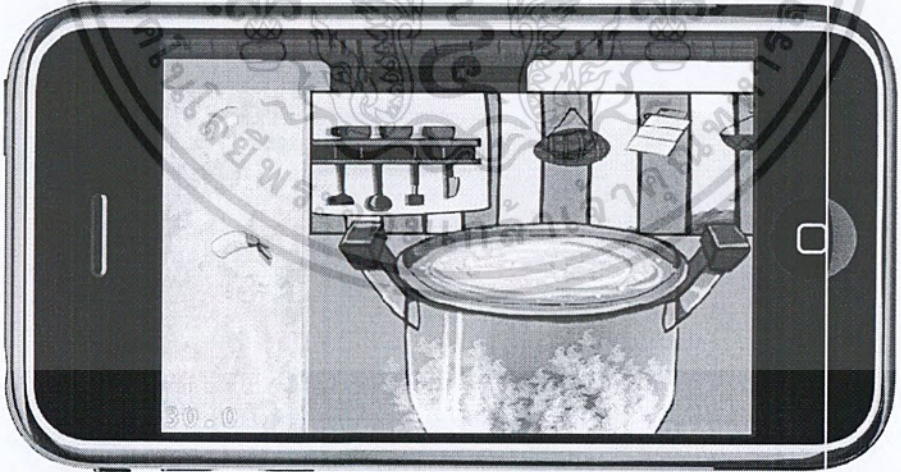
ผู้เล่นจะต้องใช้สองนิ้วบีบไปที่ด้านข้างของมะนาว เพื่อให้ น้ำมะนาวหยดลงหม้อต้ม  
ขั้นตอนนี้ได้ใช้เทคนิคการตรวจจับหน้าจอแบบมัลติทัชเข้าช่วย



รูป 6.9 หน้าจอการเล่นขั้นตอนบีบมะนาว

#### 6.4.6 ต้ม

ผู้เล่นจะต้องใส่วัตถุดิบที่จัดเตรียมมาลงในหม้อให้ถูกลำดับและทันเวลาที่กำหนด



รูป 6.10 หน้าจอการเล่นขั้นตอนต้ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

James Bucanek. 2010. **Profession Xcode 3**. Indiana : Willey Publishing, Inc.

Jiva Devoe. **Cocoa Touch For iphone OS3**. Indiana : Willey Publishing, Inc.

Neal Goldstein. **Objective-C For DUMMIES**. 1<sup>st</sup> Edition.Indiana : Wiley Publishing , Inc.

Paul Zirkle & Joe Hogue. 2009. **iPhone Game Development**. 1<sup>st</sup> Edition. O'Reilly Media, Inc.

SADUN, ERICA. 2009. **IPHONE DEVELOPER‘ S COOKBOOK**. 1<sup>st</sup> Edition. PearSon Education, Inc.

วิกิพีเดีย. 2553. ภาษาอ็อบเจกทีฟ-ซี.[Online].Available :<http://th.wikipedia.org/wiki/ภาษาอ็อบเจกทีฟ-ซี>.

วิกิพีเดีย. 2552. มัลติทัช. [Online].Available :<http://th.wikipedia.org/wiki/มัลติทัช>.

วิกิพีเดีย. 2553. ไอโฟน. [Online].Available :<http://th.wikipedia.org/wiki/ไอโฟน>.

Deborah M. Fike. 2009. **Playing Doctor on the iphone with Hospital Havoc**. [Online].Available :<http://www.torquepowered.com/community/blogs/view16625>.

JUAN SEBASTIAN MUNOC. 2009. **Cocos2d Iphone Dynamically Touch Detection**. [Online].Available :<http://juanmunozar.blogspot.com/2009/02/cocos2d-iphone-dynamically-touch.html>.

Mitchallen. 2010. **Touching Sprites in Cocos2D**. [Online].Available :<http://mitchallen.com/iphone/archives/424>.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SDKTutor. 2010. **SDKTutor**. [Online].Available :<http://youtube.com/user/SDKTutor>.

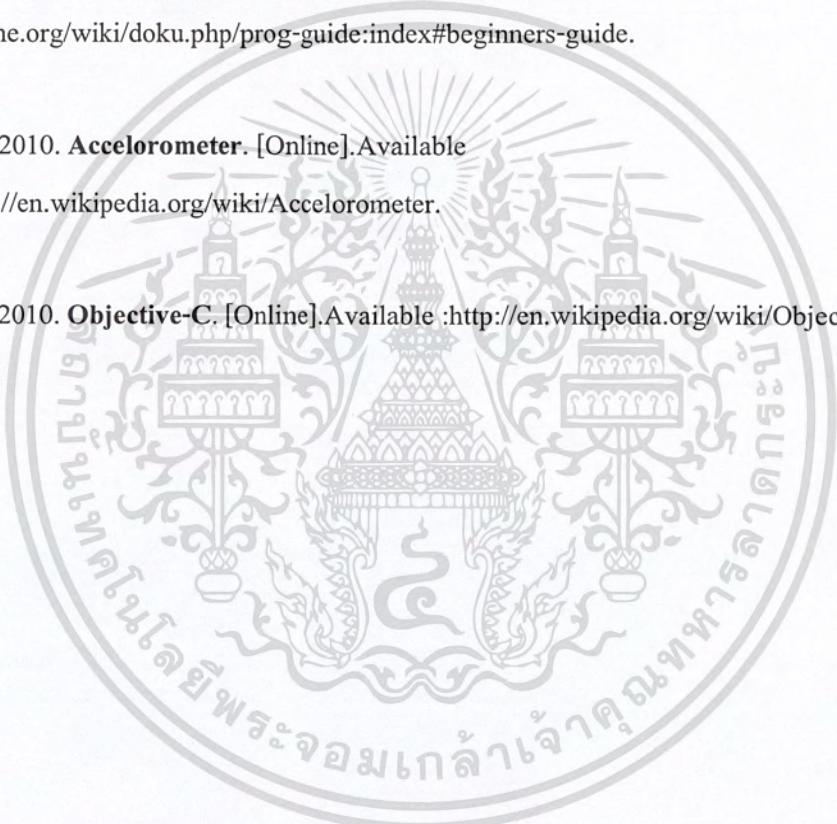
SoftwareDevelopment. 2010. **iPhone Programming**. [Online].Available : <http://xn-1207bif5ahdb4jm1oj5l.com/iPhone-Programming>.

Steffen Itterheim. **Learn Game Development with cocos2d**. [online].Available : <http://www.learn-cocos2d.com>.

Tony Rayo. 2010. **Programming Guide**. [Online].Available :<http://www.cocos2d-iphone.org/wiki/doku.php/prog-guide:index#beginners-guide>.

Wikipedia. 2010. **Accelerometer**. [Online].Available :<http://en.wikipedia.org/wiki/Accelerometer>.

Wikipedia. 2010. **Objective-C**. [Online].Available :<http://en.wikipedia.org/wiki/Objective-C>.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก

# ตัวอย่างโปรแกรมที่ทดลองเขียน

### ก.1 เมนู

เป็นการทดสอบสร้างเมนูแบบง่ายๆด้วย Cocos2D และเราก็ได้ทำการทดสอบภาษาไทยลงไปด้วย ซึ่งผลออกมาคือ Cocos2D สนับสนุนภาษาไทย



รูป ก.1 ตัวอย่างเมนู

#### ก.1.1 โค้ดของโปรแกรม

##### โปรแกรม ก.1 ไฟล์ MenuAppDelegate.h

```
@interfacemenutransitionAppDelegate : NSObject  
<UIApplicationDelegate> {  
    UIWindow *window;  
}  
@property (nonatomic, retain) UIWindow *window;  
@end
```

##### โปรแกรม ก.2 ไฟล์ MenuAppdelegate.m

```
#import "menutransitionAppDelegate.h"  
#import "cocos2d.h"  
#import "MainMenuscene.h"  
@implementation menutransitionAppDelegate  
@synthesize window;  
- (void)  
applicationDidFinishLaunching:(UIApplication*) application  
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.2 ไฟล์ MenuAppDelegate.m (ต่อ)

```

    // Init the window
    window = [[UIWindow alloc]
initWithFrame:[UIScreen mainScreen] bounds]];
    // cocos2d will inherit these values
    [window setUserInteractionEnabled:YES];
    [window setMultipleTouchEnabled:YES];
    // Try to use CADisplayLink director
    // if it fails (SDK < 3.1) use the default director
    if( !
[CCDirector setDirectorType:CCDirectorTypeDisplayLink] )
    [CCDirector setDirectorType:CCDirectorTypeDefault];
    // Use RGBA_8888 buffers
    // Default is: RGB_565 buffers
    [[CCDirector sharedDirector]
setPixelFormat:kPixelFormatRGBA8888];
    // Create a depth buffer of 16 bits
    // Enable it if you are going to use 3D transitions
or 3d objects
    // [[CCDirector sharedDirector]
setDepthBufferFormat:kDepthBuffer16];
    // Default texture format for PNG/BMP/TIFF/JPEG/GIF
images
    // It can be RGBA8888, RGBA4444, RGB5_A1, RGB565
    // You can change anytime.
    [CCTexture2D setDefaultAlphaPixelFormat:kTexture2DPixelFormat_RGBA8888];
    // before creating any layer, set the landscape mode
    [[CCDirector sharedDirector]
setDeviceOrientation:CCDeviceOrientationLandscapeLeft];
    [[CCDirector sharedDirector]
setAnimationInterval:1.0/60];
    [[CCDirector sharedDirector] setDisplayFPS:YES];
    // create an OpenGL view inside a window
    [[CCDirector sharedDirector] attachInView:window];
    [window makeKeyAndVisible];
    //call mainwindow
    MainMenuScene *scene = [MainMenuScene node];
    [[CCDirector sharedDirector] runWithScene:scene ];
}
- (void) applicationWillResignActive: (UIApplication
*) application {
    [[CCDirector sharedDirector] pause];
}
- (void) applicationDidBecomeActive: (UIApplication
*) application {
    [[CCDirector sharedDirector] resume];
}
- (void) applicationDidReceiveMemoryWarning: (UIApplication
*) application {

```

### โปรแกรม ก.2 ไฟล์ MenuAppDelegate.m (ต่อ)

```

    [[CCTextureCache sharedTextureCache]
removeUnusedTextures];
}
- (void)applicationWillTerminate:(UIApplication
*)application {
    [[CCDirector sharedDirector] end];
}
- (void)applicationSignificantTimeChange:(UIApplication
*)application {
    [[CCDirector sharedDirector]
setNextDeltaTimeZero:YES];
}
- (void)dealloc {
    [[CCDirector sharedDirector] release];
    [window release];
    [super dealloc];
}
@end

```

### โปรแกรม ก.3 ไฟล์ MainMenuScene.h

```

#import <UIKit/UIKit.h>
#import "cocos2d.h"
@interface MainMenuScene : CCScene
{
}
@end
@interface MainMenuLayer : CCLayer
{
}
//สร้างfunction สำหรับหน้า new game
-(void)newGame:(id)sender;
//สร้างfunction สำหรับหน้า help
-(void)help:(id)sender;
@end

```

### โปรแกรม ก.4 ไฟล์ MainMenuScene.m

```

#import "MainMenuScene.h"
#import "World1Level1Scene.h"
#import "HelpScene.h"
@implementation MainMenuScene
-(id)init {
    if( (self=[super init]) ) {
        CCSprite *bg =
[CCSpritespriteWithFile:@"background.png"];
        //set to center
        [bg setPosition:ccp(240,160)];
    }
}

```

## โปรแกรม ก.4 ไฟล์ MainMenuScene.m (ต่อ)

```

        //add background
        [selfaddChild:bg z:0];
        //add menulayer on blackground and call mainlayer
        [selfaddChild:[MainMenuLayernode] z:1];
    }
    returnself;
}
@end
@implementation MainMenuLayer
-(id)init {
    if((self=[superinit])) {
        self.isTouchEnabled = YES;
        //set font
        [CCMenuItemFontsetFontSize:48];
        [CCMenuItemFontsetFontName:@"Marker Felt"];
        //create menu item
        CCMenuItem *start =
        [CCMenuItemFontitemFromString:@"new"target:selfselector:@s
        elector(newGame:)];
        CCMenuItem *help =
        [CCMenuItemFontitemFromString:@"HELP"target:selfselector:
        @selector(help:)];
        CCMenuItem *menu =
        [CCMenumenuWithItems:start,help,nil];
        //add menu
        [selfaddChild:menu];
        /** must add menu
        [menualignItemsVertically];
    }
    returnself;
}
-(void)newGame:(id)sender {
    //create object for link to world1level1scene
    World1Level1Scene *worldscene =
    [World1Level1Scenenode];
    //create for link when we click
    [[CCDirectorsharedDirector] replaceScene:worldscene];
}
-(void)help:(id)sender {
    //create object for link to helpscene
    HelpScene *helper = [HelpScenenode];
    [[CCDirectorsharedDirector] replaceScene:helper];
}
@end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.5 ไฟล์ World1Level.h

```
#import <UIKit/UIKit.h>
#import "cocos2d.h"
@interfaceWorld1Level1Scene : CCScene {
}
@end
@interfaceWorld1Level1Layer : CCLayer {
}
@end
```

### โปรแกรม ก.6 ไฟล์ World1Level.m

```
#import "World1Level1Scene.h"
#import "MainMenuscene.h"
@implementation World1Level1Scene
-(id)init {
    if( (self=[superinit] ) ) {
        CCSprite *world =
[CCSpritespriteWithFile:@"world11.jpg"];
        //set position
        world.position = ccp(240,160);
        //add background
        [selfaddChild:world z:0];
        //add World1Level1Layer
        [selfaddChild:[World1Level1Layernode] z:1];
    }
    returnself;
}
@end
@implementation World1Level1Layer
-(id)init {
    if( (self=[superinit]) ) {
        //Set Label
        CCLabel *label = [CCLabellabelWithString:@"This is
World1Level1"fontName:@"Marker Felt"fontSize:48];
        //set position
        [labelsetPosition:ccp(240,160)];
        //Add Label ลงบนlayer
        [selfaddChild:label];
    }
    returnself;
}
@end
```

### โปรแกรม ก.7 ไฟล์ HelpScene.h

```
@interfaceHelpScene : CCScene {} @end
@interfaceHelpLayer : CCLayer {} @end
```

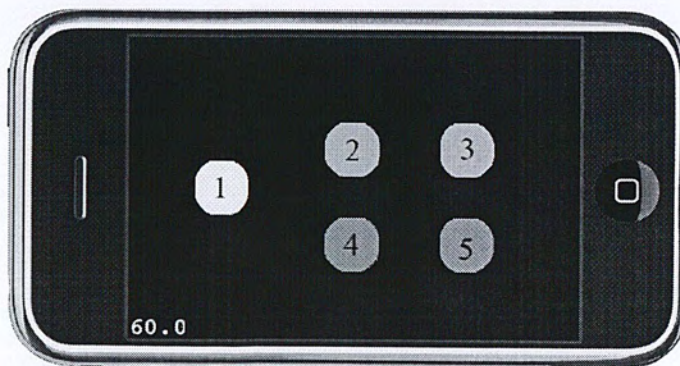
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.8 ไฟล์ HelpScene.m

```
#import "HelpScene.h"
#import "MainMenuScene.h"
@implementation HelpScene
-(id)init {
    if((self=[superinit])) {
        //กำหนด image ที่จะใช้
        CCSprite *help =
[CCSpritespriteWithFile:@"help.jpg"];
        //Set position
        [helpsetPosition:ccp(240,160)];
        [selfaddChild:help z:0];
        [selfaddChild:[HelpLayernode] z:1];
    }
    returnself;
}
@end
@implementation HelpLayer
-(id)init {
    if((self =[superinit])) {
        //สร้างString "HelperScene" โดยใช้Font"Marker Felt"ขนาด48
        CCLabel *labell =
[CCLabellabelWithString:@"HelperScene"fontName:@"Marker
Felt"fontSize:48];
        [labellsetPosition:ccp(240,160)];
        [selfaddChild:labell];
    }
    returnself;
}
@end
```

### ก.2 ทดสอบการสัมผัสและการตอบรับ

เป็นการตรวจจบการสัมผัสจุดที่กำหนด จากนั้นให้โปรแกรมดำเนินการตามที่ได้เขียนโปรแกรมไว้ ในตัวอย่างนี้ได้กำหนดในลูกบอลเคลื่อนที่เมื่อเราสัมผัสไปบนจุดที่กำหนดของจอ



รูป ก.2 ทดสอบการสัมผัสและการตอบรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ก.2.1 โค้ดของโปรแกรม

### โปรแกรม ก.9 ไฟล์ MoveImage3AppDelegate.h

```
#import <UIKit/UIKit.h>
@interface MoveImage3AppDelegate : NSObject
<UIApplicationDelegate> {
    UIWindow *window;
}
@property (nonatomic, retain) UIWindow *window;
@end
```

### โปรแกรม ก.10 ไฟล์ MoveImage3AppDelegate.m

```
#import "MoveImage3AppDelegate.h"
#import "cocos2d.h"
#import "HelloWorldScene.h"
@implementation MoveImage3AppDelegate
@synthesize window;
- (void)
applicationDidFinishLaunching:(UIApplication*)application
{
    // Init the window
    window = [[UIWindow alloc]
initWithFrame:[UIScreen mainScreen] bounds]];
    // cocos2d will inherit these values
    [window setUserInteractionEnabled:YES];
    [window setMultipleTouchEnabled:YES];
    // Try to use CADisplayLink director
    // if it fails (SDK < 3.1) use the default director
    if( !
[CCDirector setDirectorType:CCDirectorTypeDisplayLink] )
        [CCDirector setDirectorType:CCDirectorTypeDefault];
    // Use RGBA_8888 buffers
    // Default is: RGB_565 buffers
    [[CCDirector sharedDirector]
setPixelFormat:kPixelFormatRGBA8888];
    // Create a depth buffer of 16 bits
    // Enable it if you are going to use 3D transitions
or 3d objects
    // [[CCDirector sharedDirector]
setDepthBufferFormat:kDepthBuffer16];
    // Default texture format for PNG/BMP/TIFF/JPEG/GIF
images
    // It can be RGBA8888, RGBA4444, RGB5_A1, RGB565
    [CCTexture2D setDefaultAlphaPixelFormat:kTexture2DPixelFormat_RGBA8888];
    // before creating any layer, set the landscape mode
    [[CCDirector sharedDirector]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.10 ไฟล์ MoveImage3AppDelegate.m (ต่อ)

```

setDeviceOrientation:CCDeviceOrientationLandscapeLeft];
    [[CCDirectorsharedDirector]
setAnimationInterval:1.0/60];
    [[CCDirectorsharedDirector] setDisplayFPS:YES];
    // create an OpenGL view inside a window
    [[CCDirectorsharedDirector] attachInView:window];
    [windowmakeKeyAndVisible];
    [[CCDirectorsharedDirector] runWithScene:
[HelloWorldscene]];
}
- (void)applicationWillResignActive:(UIApplication
*)application {
    [[CCDirectorsharedDirector] pause];
}
- (void)applicationDidBecomeActive:(UIApplication
*)application {
    [[CCDirectorsharedDirector] resume];
}
- (void)applicationDidReceiveMemoryWarning:(UIApplication
*)application {
    [[CCTextureCachesharedTextureCache]
removeUnusedTextures];
}
- (void)applicationWillTerminate:(UIApplication
*)application {
    [[CCDirectorsharedDirector] end];
}
- (void)applicationSignificantTimeChange:(UIApplication
*)application {
    [[CCDirectorsharedDirector]
setNextDeltaTimeZero:YES];
}
- (void)dealloc {
    [[CCDirectorsharedDirector] release];
    [windowrelease];
    [superdealloc];
}
@end

```

### โปรแกรม ก.11 ไฟล์ HelloWorldScene.h

```

#import "cocos2d.h"
// HelloWorld Layer
@interface HelloWorld : CCLayer
{
    CCSprite *chick;
    CCSprite

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.11 ไฟล์ HelloWorldScene.h (ต่อ)

```
*Ballone,*Balltwo,*Ballthree,*Ballfour,*Ballfive;
}
// returns a Scene that contains the HelloWorld as the
only child
+(id) scene;
@end
```

### โปรแกรม ก.12 ไฟล์ HelloWorldScene.m

```
#import "HelloWorldScene.h"
int globalseries = 1;
// HelloWorld implementation
@implementation HelloWorld

+(id) scene {
    // 'scene' is an autorelease object.
    CCGScene *scene = [CCScenenode];
    // 'layer' is an autorelease object.
    HelloWorld *layer = [HelloWorldnode];
    // add layer as a child to scene
    [sceneaddChild:layer];
    // return the scene
    return scene;
}
// on "init" you need to initialize your instance
-(id) init {
    // always call "super" init
    // Apple recommends to re-assign "self" with the
"super" return value
    if( (self=[superinit]) ) {
        //Create Menu item
        CCMenuItemImage *item1 =
[CCMenuItemImageitemFromNormalImage:@"ghostbutton.png"sel
ectedImage:@"ghostbutton.png"target:selfselector:@selecto
r(doThis:)];
        //You can set position of image (if not set button,
button will position at center of iphone)
        //item1.position = ccp(100,100);
        CCMenu *menu = [CCMenuWithItems:item1, nil];
        // Add menu item ลงบนLayer
        [selfaddChild:menu z:1];
        // Can use touch
        self.isTouchEnabled =YES;
        //STEP2 : intilazile object that you declare
        //Ballone,*Balltwo,*Ballthree,*Ballfour,*Ballfive
        Ballone = [CCSpritespriteWithFile:@"ball1.png"];
        Ballone.position = ccp(100,160);
        Balltwo = [CCSpritespriteWithFile:@"ball2.png"];

```

## โปรแกรม ก.12 ไฟล์ HelloWorldScene.m (ต่อ)

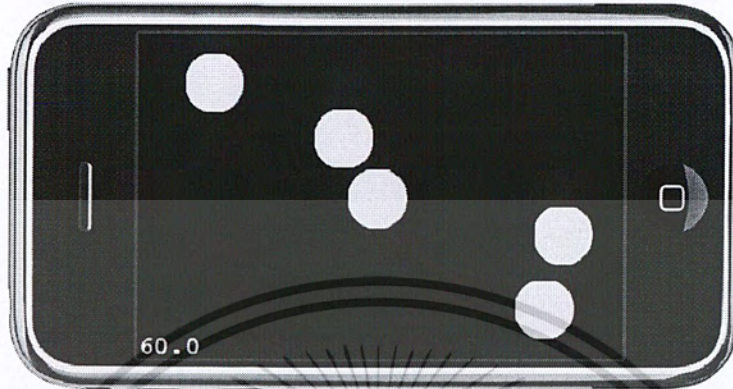
```

    Balltwo.position = ccp(240,200);
    Ballthree = [CCSpritespriteWithFile:@"ball3.png"];
    Ballthree.position = ccp(360,200);
    Ballfour = [CCSpritespriteWithFile:@"ball4.png"];
    Ballfour.position = ccp(240,100);
    Ballfive = [CCSpritespriteWithFile:@"ball5.png"];
    Ballfive.position = ccp(360,100);
    //addลูกบอลลงบนLayer
    [selfaddChild:Ballone];
    [selfaddChild:Balltwo];
    [selfaddChild:Ballthree];
    [selfaddChild:Ballfour];
    [selfaddChild:Ballfive];
}
returnself;
}
//Let's the action begin
-(void)doThis:(id)sender {
    //[chick runAction:[CCMoveBy actionWithDuration:.3
position:ccp(40,20)]];
    if(globalseries==1) {
        [BallonerunAction:[CCMoveByactionWithDuration:.3posit
ion:ccp(-50,0)]];
        globalseries++;
    }
    elseif(globalseries==2) {
        [BalltworunAction:[CCMoveByactionWithDuration:.3posit
ion:ccp(0,70)]];
        globalseries++;
    }
    elseif(globalseries==3) {
        [BallthreerunAction:[CCMoveByactionWithDuration:.3pos
ition:ccp(0,70)]];
        globalseries++;
    }
    elseif(globalseries==4) {
        [BallfourrunAction:[CCMoveByactionWithDuration:.3posi
tion:ccp(0,-70)]];
        globalseries++;
    }
    elseif(globalseries==5) {
        [BallfiverunAction:[CCMoveByactionWithDuration:.3posi
tion:ccp(0,-70)]];
        globalseries++;
    }
}
-(void) dealloc {
    [superdealloc];
}
@end

```

### ก.3 มัลติทัช

เป็นการสร้างลูกบอลขึ้นมา 5 ลูก โดยจะสามารถเคลื่อนลูกบอลทั้ง 5 ได้โดยอิสระในช่วงเวลาเดียวกัน ด้วยการสัมผัสและลากลูกบอลนั้นๆ ไปมา



รูป ก.3 การทดสอบมัลติทัช

#### ก.3.1 โค้ดของโปรแกรม

##### โปรแกรม ก.13 ไฟล์ Multiple3AppDelegate.h

```
#import <UIKit/UIKit.h>
#import "cocos2d.h"
@interface Multiple3AppDelegate : NSObject
<UIApplicationDelegate> {
    UIWindow *window;
}
@property (nonatomic, retain) UIWindow *window;
@end
```

##### โปรแกรม ก.14 ไฟล์ Multiple3AppDelegate.m

```
#import "Multiple3AppDelegate.h"
#import "cocos2d.h"
#import "MainScene.h"
@implementation Multiple3AppDelegate
@synthesize window;
- (void)
applicationDidFinishLaunching:(UIApplication*) application
{
    window = [[UIWindow alloc]
initWithFrame:[UIScreen mainScreen] bounds]];
// cocos2d will inherit these values
[window setUserInteractionEnabled:YES];
[window setMultipleTouchEnabled:YES];
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.14 ไฟล์ Multiple3Appdelegate.m (ต่อ)

```

    if( !
[CCDirectorsetDirectorType:CCDirectorTypeDisplayLink] )
    [CCDirectorsetDirectorType:CCDirectorTypeDefault];
    [[CCDirectorsharedDirector]
setPixelFormat:kPixelFormatRGBA8888];
    [CCTexture2DsetDefaultAlphaPixelFormat:kTexture2DPixelFormat_RGBA8888];
    // before creating any layer, set the landscape mode
    [[CCDirectorsharedDirector]
setDeviceOrientation:CCDeviceOrientationLandscapeLeft];
    [[CCDirectorsharedDirector]
setAnimationInterval:1.0/60];
    [[CCDirectorsharedDirector] setDisplayFPS:YES];
    // create an openGL view inside a window
    [[CCDirectorsharedDirector] attachInView:window];
    [windowmakeKeyAndVisible];
    [[CCDirectorsharedDirector] runWithScene:
[MainScenenode]];
}
- (void)applicationWillResignActive:(UIApplication
*)application {
    [[CCDirectorsharedDirector] pause];
}
- (void)applicationDidBecomeActive:(UIApplication
*)application {
    [[CCDirectorsharedDirector] resume];
}
- (void)applicationDidReceiveMemoryWarning:(UIApplication
*)application {
    [[CCTextureCachesharedTextureCache]
removeUnusedTextures];
}
- (void)applicationWillTerminate:(UIApplication
*)application {
    [[CCDirectorsharedDirector] end];
}
- (void)applicationSignificantTimeChange:(UIApplication
*)application {
    [[CCDirectorsharedDirector]
setNextDeltaTimeZero:YES];
}
- (void)dealloc {
    [[CCDirectorsharedDirector] release];
    [windowrelease];
    [superdealloc];
}
@end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.15 ไฟล์ mainscene.h

```
#import "cocos2d.h"
@interface MainScene : CCScene {
@private
}

@end

@interface MainLayer : CCLayer {
@private
    NSMutableArray *ballarray;
}
@end
```

## โปรแกรม ก.16 ไฟล์ mainscene.m

```
#import "MainScene.h"
#import "Ball.h"
@interface MainLayer ()
@end
@implementation MainScene
-(id)init {
    if ( (self = [superinit]) == nil ) return nil;
    [self addChild:[MainLayerNode]];
    return self;
}
-(void)onExit {
    [super onExit];
}
@end
@implementation MainLayer
-(id)init {
    if ( (self = [superinit]) == nil ) return nil;
    NSMutableArray *ballMutable =
    [NSMutableArray arrayWithCapacity:5];
    //สร้าง Texture2D ของ image ที่จะใช้งาน
    CCTexture2D *ballTexture =
    [[CCTextureCache sharedTextureCache] addImage:@"ballY.png"];
    ;
    Ball *ball = [BallBallWithTexture:ballTexture];
    //กำหนดตำแหน่งของลูกบอล
    ball.position = CGPointMake(arc4random()%480,
    arc4random()%320);
    [ballMutable addObject:ball];
    ball = [BallBallWithTexture:ballTexture];
    ball.position = CGPointMake(arc4random()%480,
    arc4random()%320);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.16 ไฟล์ mainscene.m (ต่อ)

```

    [ballMutableaddObject:ball];
    ball = [BallBallWithTexture:ballTexture];
    ball.position = CGPointMake(arc4random()%480,
arc4random()%320);
    [ballMutableaddObject:ball];

    ball = [BallBallWithTexture:ballTexture];
    ball.position = CGPointMake(arc4random()%480,
arc4random()%320);
    [ballMutableaddObject:ball];
    ball = [BallBallWithTexture:ballTexture];
    ball.position = CGPointMake(arc4random()%480,
arc4random()%320);
    [ballMutableaddObject:ball];
    ballarray = [ballMutable copy];
    //add ลูกบอลลงบนLayer
    for (Ball *ball inballarray) {
        [selfaddChild:ball];
    }
    returnself;
}
-(void)dealloc {
    [superdealloc];
}
@end

```

## โปรแกรม ก.17 ไฟล์ ball.h

```

#import "cocos2d.h"
//ใช้สำหรับcheck การTouch
typedefenum BallTouch {
    kBallstateGrabbed,
    kBallstateUnGrabbed,
}BallState;
@interfaceBall : CCSprite<CCTargetedTouchDelegate> {
@private
    BallStateballstate;
}
@property(nonatomic, readonly)CGRect rect;
+(id)BallWithTexture:(CCTexture2D *)aTexture;
@end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.18 ไฟล์ ball.m

```

#import "cocos2d.h"
#import "Ball.h"
@implementation Ball
//การรูปสี่เหลี่ยมให้กับSprite
-(CGRect)rect {
    CGSize s = [self.textureContentSize];
    returnCGRectMake(-s.width/2, -s.height/2, s.width,
s.height);
}
+(id)BallWithTexture:(CCTexture2D *)aTexture {
    return [[selfalloc] initWithTexture:aTexture]
autorelease];
}
-(id)initWithTexture:(CCTexture2D *)aTexture {
    if ((self = [superinitWithTexture:aTexture]) ) {
        ballstate = kBallstateUnGrabbed;
    }
    returnself;
}
-(void)onEnter {
[[CCTouchDispatchersharedDispatcher]addTargetedDelegate:s
elfpriority:0swallowsTouches:YES];
    [superonEnter];
}
-(void)onExit {
[[CCTouchDispatchersharedDispatcher]removeDelegate:se
lf];
    [superonExit];
}
-(BOOL)containTouchLocation:(UITouch *)touch {
    returnCGRectContainsPoint(self.rect,
[selfconvertTouchToNodeSpaceAR:touch]);
}
-(BOOL)ccTouchBegan:(UITouch *)touch withEvent:(UIEvent
*)event {
    if (ballstate != kBallstateUnGrabbed) {
        returnNO;
    }
    if (![selfcontainTouchLocation:touch]) {
        returnNO;
    }
    ballstate = kBallstateGrabbed;
    returnYES;
}
-(void)ccTouchMoved:(UITouch *)touch withEvent:(UIEvent
*)event {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.18 ไฟล์ ball.m (ต่อ)

```

CGPoint touchPoint = [touch locationInView:[touch
view]];
touchPoint =
[[CCDirectorsharedDirector]convertToGL:touchPoint];
self.position = ccp(touchPoint.x,touchPoint.y);
}
-(void)ccTouchEnded:(UITouch *)touch withEvent:(UIEvent
*)event {
ballstate = kBallstateUnGrabbed;
}
@end

```

### ก.4 การเล่นไฟล์เสียง

เป็นการทดสอบนำไฟล์เสียงไปเล่นบนไอโฟน โดยได้ใช้ไฟล์เสียงสกุล wav เป็นตัวทดสอบ



รูป ก.4 โปรแกรมเล่นไฟล์เสียงแคะร้อง พร้อมแสดงภาพแคะ

#### ก.4.1 โค้ดของโปรแกรม

### โปรแกรม ก.19 ไฟล์ SoundAppDelegate.h

```

#import <UIKit/UIKit.h>
@class sound_iphoneViewController;
@interface sound_iphoneAppDelegate : NSObject
<UIApplicationDelegate> {
UIWindow *window;
sound_iphoneViewController *viewController;}
@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet
sound_iphoneViewController *viewController;
@end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.20 ไฟล์ SoundAppDelegate.m

```
#import "sound_iphoneAppDelegate.h"
#import "sound_iphoneViewController.h"
@implementation sound_iphoneAppDelegate
@synthesize window;
@synthesize viewController;
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary
*)launchOptions {
// Override point for customization after app launch
[window addSubview:viewController.view];
[window makeKeyAndVisible];
return YES;
}
- (void)dealloc {
[viewController release];
[window release];
[super dealloc];
}
@end
```

### โปรแกรม ก.21 ไฟล์ sound\_iphone.h

```
#import <UIKit/UIKit.h>
#import <AudioToolbox/AudioToolbox.h>
@interface sound_iphoneViewController : UIViewController
{
}
@end
```

### โปรแกรม ก.22 ไฟล์ Sound\_iphone.m

```
#import "sound_iphoneViewController.h"
static void MySoundFinishedPlayingCallback(SystemSoundID
sound_id, void* user_data) {
    AudioServicesDisposeSystemSoundID(sound_id);
}
@implementation sound_iphoneViewController
// Implement viewDidLoad to do additional setup after
loading the view, typically from a nib.
- (void)viewDidLoad {
    [super viewDidLoad];
    NSString *path = [[NSBundle mainBundle]
pathForResource:@"Sheep" ofType:@"wav"];
    SystemSoundID soundID;
    AudioServicesCreateSystemSoundID((CFURLRef)[NSURLfile
URLWithPath:path], &soundID);
    AudioServicesAddSystemSoundCompletion(soundID, NULL,
NULL, MySoundFinishedPlayingCallback, NULL);
}
```

### โปรแกรม ก.22 ไฟล์ Sound\_iphone.m (ต่อ)

```

//Play Sound
AudioServicesPlaySystemSound(soundID);
}

- (void)didReceiveMemoryWarning {
// Releases the view if it doesn't have a superview.
[superdidReceiveMemoryWarning];
}

- (void)viewDidUnload {
// Release any retained subviews of the main view.
// e.g. self.myOutlet = nil;
}

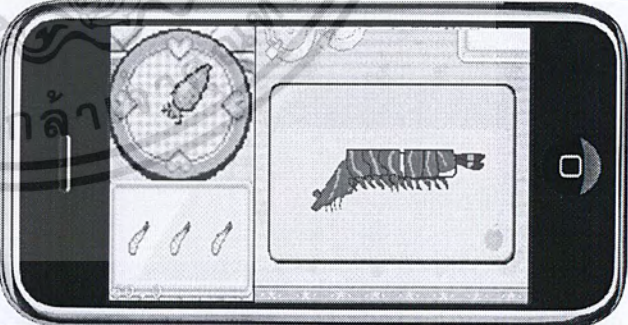
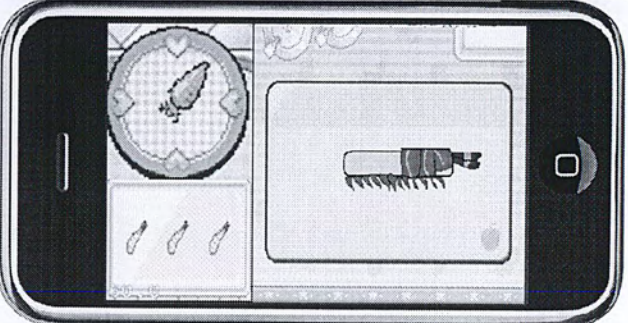
- (void)dealloc {
[superdealloc];
}
@end

```

### ก.5 ทดสอบเขียนเกม

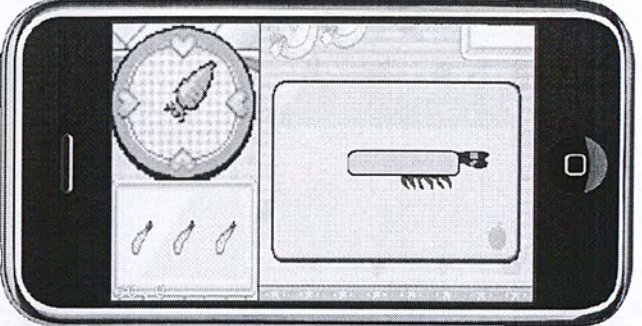
หลังจากได้ทำการศึกษาการใช้เครื่องมือต่างๆมาพอสมควร จึงได้ลองเขียนโปรแกรมเล็กๆ เพื่อจำลองการเล่นเกมนิรันดร์ในการปลูกกล้วยให้เห็นเป็นรูปร่างพอสังเขป ตามขั้นตอนในตารางต่อไปนี้

#### ตาราง ก.1 ขั้นตอนการปลูกกล้วย

ขั้นตอน	วิธีการเล่น	ภาพประกอบ
1. ปลูกหัวกล้วย	ใช้นิ้วแตะไปบนหัวกล้วย แล้วโยกหัวกล้วยขึ้นลงไปเรื่อยๆ หัวกล้วยจะหลุดออกมาในที่สุด	
2. ปลูกเปลือกกล้วย	ใช้นิ้วแตะไปที่เปลือกกล้วย แล้วสไลด์ออกด้านบน	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตาราง ก.1 ขั้นตอนการปลอกุ้ง (ต่อ)

3.ปลอกุ้ง	ใช้นิ้วแตะไปที่ขากุ้ง แล้วสไลด์ออกด้านล่าง	
4.เสร็จสิ้น	จะเห็นว่าเหลือเพียงแค่เนื้อและหางกุ้งเท่านั้น	

### ก.5.1 โค้ดของโปรแกรม

#### โปรแกรม ก.23 ไฟล์ PeelShrimp4AppDelegate.h

```
#import <UIKit/UIKit.h>
@interface PeelShrimp4AppDelegate : NSObject
<UIApplicationDelegate> {
    UIWindow *window;
}
@property (nonatomic, retain) UIWindow *window;
@end
```

#### โปรแกรม ก.24 ไฟล์ PeelShrimp4AppDelegate.m

```
#import "PeelShrimp4AppDelegate.h"
#import "cocos2d.h"
#import "HelloWorldScene.h"
@implementation PeelShrimp4AppDelegate
@synthesize window;
- (void)
applicationDidFinishLaunching:(UIApplication*) application
{
    // Init the window
    window = [[UIWindow alloc]
initWithFrame:[UIScreen mainScreen] bounds]];
    // cocos2d will inherit these values
    [window setUserInteractionEnabled:YES];
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.24 ไฟล์ PeelShimp4AppDelegate.m (ต่อ)

```

    [windowsetMultipleTouchEnabled:YES];
    // Try to use CADisplayLink director
    // if it fails (SDK < 3.1) use the default director
    if( !
[CCDirectorsetDirectorType:CCDirectorTypeDisplayLink] )
    [CCDirectorsetDirectorType:CCDirectorTypeDefault];

    // Use RGBA_8888 buffers
    // Default is: RGB_565 buffers
    [[CCDirectorsharedDirector]
setPixelFormat:kPixelFormatRGBA8888];
    // Create a depth buffer of 16 bits
    // Enable it if you are going to use 3D transitions
or 3d objects
    //    [[CCDirector sharedDirector]
setDepthBufferFormat:kDepthBuffer16];
    // Default texture format for PNG/BMP/TIFF/JPEG/GIF
images
    // It can be RGBA8888, RGBA4444, RGB5_A1, RGB565
    // You can change anytime.
    [CCTexture2DsetDefaultAlphaPixelFormat:kTexture2DPixe
lFormat_RGBA8888];
    // before creating any layer, set the landscape mode
    [[CCDirectorsharedDirector]
setDeviceOrientation:CCDeviceOrientationLandscapeLeft];
    [[CCDirectorsharedDirector]
setAnimationInterval:1.0/60];
    [[CCDirectorsharedDirector] setDisplayFPS:YES];
    [[CCDirectorsharedDirector] attachInView>window];
    [windowmakeKeyAndVisible];
    [[CCDirectorsharedDirector] runWithScene:
[HelloWorldscene]];
}
- (void)applicationWillResignActive:(UIApplication
*)application {
    [[CCDirectorsharedDirector] pause];
}
- (void)applicationDidBecomeActive:(UIApplication
*)application {
    [[CCDirectorsharedDirector] resume];
}
- (void)applicationDidReceiveMemoryWarning:(UIApplication
*)application {
    [[CCTextureCachesharedTextureCache]
removeUnusedTextures];
}
- (void)applicationWillTerminate:(UIApplication
*)application {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.24 ไฟล์ PeelShimp4AppDelegate.m (ต่อ)

```

    [[CCDirectorsharedDirector] end];
}
- (void)applicationSignificantTimeChange:(UIApplication
*)application {
    [[CCDirectorsharedDirector]
setNextDeltaTimeZero:YES];
}
- (void)dealloc {
    [[CCDirectorsharedDirector] release];
    [windowrelease];
    [superdealloc];
}
@end

```

### โปรแกรม ก.25 ไฟล์ HelloWorldScene.h

```

#import "cocos2d.h"
@interface HelloWorld : CCLayer
{
    // สร้างobject ของclass CCSprite
    CCSprite
*Ballone,*Balltwo,*Ballthree,*Ballfour,*Ballfive;
    CCSprite *Ballin;
    CCSprite *BodyShrimp;
    CCSprite *Board,*Bar;
}
// returns a Scene that contains the HelloWorld as the
only child
+(id) scene;
@end

```

### โปรแกรม ก.26 ไฟล์ HelloWorldScene.m

```

@implementation HelloWorld
int globalseries = 1;
int headHP =5; //init HP of head. If headHP=0, the head
will fly away.
int counthitupdown = 1; //for count hit up and down. 1
mean already hitup.2 mean already hitdown
+(id) scene
{
    // 'scene' is an autorelease object.
    CCScene *scene = [CCScenenode];
    // 'layer' is an autorelease object.
    HelloWorld *layer = [HelloWorldnode];
    // add layer as a child to scene
    [sceneaddChild: layer];
    // return the scene

```

## โปรแกรม ก.26 ไฟล์ HelloWorldScene.m (ต่อ)

```

    return scene;
}
// on "init" you need to initialize your instance
-(id) init {
    // always call "super" init
    // Apple recommends to re-assign "self" with the
"super" return value
    if( (self=[superinit] ) ) {
        self.isEnabled = YES;
        //กำหนดรูปที่จะใช้งาน
        Board = [CCSpritespriteWithFile:@"board.png"];
        //กำหนดตำแหน่ง
        Board.position = ccp(325,160);
        //กำหนดให้เป็นbackground
        [self addChild:Boardz:0];
        BodyShrimp = [CCSpritespriteWithFile:@"body.png"];
        BodyShrimp.position = ccp(350,160);
        // add object ลงบนlayer
        [self addChild:BodyShrimp];

        //BallIn mean Invisible Ball
        BallIn =
[CCSpritespriteWithFile:@"ghostbutton.png"];
        BallIn.position = ccp(260,160);
        [self addChild:BallIn];
        Ballone = [CCSpritespriteWithFile:@"head2.png"];
        Ballone.position = ccp(280,165);
        Balltwo = [CCSpritespriteWithFile:@"upper
shell.png"];
        Balltwo.position = ccp(304,160);
        Ballthree = [CCSpritespriteWithFile:@"lower
shell.png"];
        Ballthree.position = ccp(367,160);
        Ballfour = [CCSpritespriteWithFile:@"upper
leg.png"];
        Ballfour.position = ccp(304,138);
        Ballfive = [CCSpritespriteWithFile:@"lower
leg.png"];
        Ballfive.position = ccp(360,138);
        [self addChild:Ballonez:1]; //I don't know but don't
forget it
        [self addChild:Balltwoz:1];
        [self addChild:Ballthreez:1];
        [self addChild:Ballfourz:1];
        [self addChild:Ballfivez:1];
        Bar = [CCSpritespriteWithFile:@"Timebar.png"];
        Bar.position = ccp(84,160);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.26 ไฟล์ HelloWorldScene.m (ต่อ)

```

        [self addChild:Barz:2];
    }
    return self;
}
// เป็น function เมื่อทำการลากนิ้วจะถูกเรียกใช้งาน
-(void) ccTouchesMoved:(NSSet *)touches
withEvent:(UIEvent *)event{

    if(globalseries == 1){
        //init point to touch
        UITouch *touch = [touches anyObject];
        //เก็บตำแหน่งที่ทำการสัมผัสบนหน้าจอ
        CGPoint point = [touch locationInView:[touch view]];
        point = [[CCDirector sharedDirector]
convertToGL:point];
        //control rotation of shrimp's head
        float angleRadians = atanf((float)(point.y-
Ballone.position.y)/(float)(point.x-Ballone.position.x));
        float angleDegrees =
CC_RADIANS_TO_DEGREES(angleRadians);
        float cocosAngle = -1 * angleDegrees;
        if(cocosAngle >= -60 && cocosAngle <= 60 &&
point.x < 270)
            Ballone.rotation = cocosAngle;
        if(cocosAngle <= -40 && counthitupdown == 1) {
            headHP--;
            counthitupdown = 2;
        }
        elseif(cocosAngle >= 40 && counthitupdown == 2) {
            counthitupdown = 1;
        }
        if(headHP <= 0) {
            Ballone.rotation = 0;
            [Ballone runAction:[CCMoveBy actionWithDuration:.5 position:ccp(-200,0)]];
            BallIn.position = ccp(304,160);
            globalseries++;
        }
    }
    elseif(globalseries == 2) {
        //BallIn.position = ccp(50, 160);
        UITouch *touch = [touches anyObject];
        CGPoint point = [touch locationInView:[touch view]];
        point = [[CCDirector sharedDirector]
convertToGL:point];
        BallIn.position = ccp(point.x, point.y);
        if(BallIn.position.y >= 230 &&
(BallIn.position.x > 272 && BallIn.position.x < 336)) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.26 ไฟล์ HelloWorldScene.m (ต่อ)

```

    [BalltwoRunAction:[CCMoveByactionWithDuration:.5posit
ion:ccp(0,200)]];
        globalseries++;
        BallIn.position =ccp(367,160);
    }
    else
        BallIn.position =ccp(304,160);
}
elseif(globalseries == 3)
{
    ///Balltwo.position = ccp(240, 250);
    UITouch *touch = [touches anyObject];
    CGPoint point = [touch locationInView:[touch view]];
    point = [[CCDirectorsharedDirector]
convertToGL:point];
    BallIn.position = ccp(point.x, point.y);
    if(BallIn.position.y>= 230&&
(BallIn.position.x>337&&BallIn.position.x<397)) {
        [BallthreeRunAction:[CCMoveByactionWithDuration:.5posi
tion:ccp(0,200)]];
        globalseries++;
        BallIn.position =ccp(304,138);
    }
    else
        BallIn.position =ccp(367,160);
}
elseif(globalseries == 4)
{
    //Ballfour.position = ccp(240, 250);
    UITouch *touch = [touches anyObject];
    CGPoint point = [touch locationInView:[touch view]];
    point = [[CCDirectorsharedDirector]
convertToGL:point];
    BallIn.position = ccp(point.x, point.y);
    if(BallIn.position.y<= 80&&
(BallIn.position.x>276&&BallIn.position.x<330)) {

        [BallfourRunAction:[CCMoveByactionWithDuration:.5posi
tion:ccp(0,-200)]];
        globalseries++;
        BallIn.position =ccp(360,138);
    }
    else
        BallIn.position =ccp(304,138);
}
elseif(globalseries == 5)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โปรแกรม ก.26 ไฟล์ HelloWorldScene.m (ต่อ)

```

        //Ballfive.position = ccp(240, 250);
        UITouch *touch = [touches anyObject];
        CGPoint point = [touch locationInView:[touch view]];
        point = [[CCDirectorsharedDirector]
convertToGL:point];
        BallIn.position = ccp(point.x, point.y);
        if(BallIn.position.y<= 80&&
(BallIn.position.x>330&&BallIn.position.x<390)) {

            [BallfiverunAction:[CCMoveByactionWithDuration:.5posi
tion:ccp(0,-200)]];
            globalseries++;
        }
        else
            BallIn.position =ccp(360,138);
    }
}
// on "dealloc" you need to release all your retained
objects
- (void) dealloc {
    // in case you have something to dealloc, do it in
this method
    // in this particular example nothing needs to be
released.

// cocos2d will automatically release all the children
(Label)
    [Ballonerelease];
    Ballone = nil;
    [Balltworelease];
    Balltwo = nil;
    [Ballthreerelease];
    Ballthree = nil;
    [Ballfourrelease];
    Ballfour = nil;
    [Ballfiverelease];
    Ballfive = nil;
    [BallInrelease];
    BallIn = nil;
    [BodyShrimprelease];
    BodyShrimp = nil;
    [Boardrelease];
    Board = nil;
    [Barrelease];
    Bar = nil;
    [superdealloc];
}
@end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.27 ไฟล์ Multiplayer.h

```

// Multiplayer.h
// ThaiARoi
//
// Created by admin on 2/5/11.
// Copyright 2011 __MyCompanyName__. All rights
reserved.
//

#import <Foundation/Foundation.h>
#import "cocos2d.h"
#import <GameKit/GameKit.h>

#define MAX_PACKET_SIZE 1024

typedef enum {
    RoleServer,
    RoleClient
} LinkRole;
#define PACKET_COINTOSS -1 // decide who is going to
be the server

typedef enum {
    MUL_CREATED,
    MUL_STAGE1,
    MUL_STAGE2,
    MUL_STAGE3,
    MUL_STAGE4,
    MUL_STAGE5,
    MUL_STAGE6,
    MUL_STAGE_FIN_SCORES,
    MUL_STAGE_FIN_RESULT
} PacketCodes;

typedef enum{
    RESULT_LOSE,
    RESULT_WIN
}GameResult;

@protocol DataReceiver

- (void)receivePacket:(int)packetID
objectIndex:(int)objectIndex data:(void*)data;
@end

@protocol LinkDelegate
-(void) linkConnected: (LinkRole) role;
-(void) linkDisconnected;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.27 ไฟล์ Multiplayer.h (ต่อ)

```

@end
@interface Multiplayer :
CCLayer<GKPeerPickerControllerDelegate,GKSessionDelegate>
{
    /*****Multiplayer part*****/
    id<LinkDelegate> delegate;
    id<DataReceiver> dataReceiver;
    NSString *sessionID;
    NSString *name;
    GKSession *session;
    UIAlertView *connectionAlert;
    NSInteger state;
    NSInteger role;
    int
        uniqueID;
    int
        peerUniqueID;
    int
        packetNumber;
    NSString *peerID;
    NSString *peerName;
    CCLabel *SendPacket,*RecPacket;
    CCSprite *BGmulti;
    CCSprite *spred;

    int StageNumber;
    int isWin;

    /*****Stage1 part*****/
    CCSprite *background;
    CCMenuItemImage *item1;
    CCSprite *head;
    CCSprite *body;
    CCSprite *shield;
    CCSprite *leg;
    int number;
    int headHP;
    int count;
    int counthitupdown;
    BOOL stepOne;
    BOOL stepTwo;
    BOOL stepThree;

    //Progress Timer
    CCProgressTimer *timer;
    float timer1;

    //Progress Bar
    CCProgressTimer *bar;
    CCProgressTimer *barstage;
    CCSprite *whitecircle;

```

## โปรแกรม ก.27 ไฟล์ Multiplayer.h (ต่อ)

```

//***** Stage2 part *****
CCSprite *knife,*lemongrass,*arrow;
int healthValue;
int globalseries;
int lemongrassHP;
int counthitupdown_st2;
//***** Stage3 part *****
CCSprite *galingale,*rightgal;
int healthPoint;
int globalserie;
int lemon_grassHP;
int counthupdown; //for count hit up and down. 1
mean already hitup.2 mean already hitdown
int cutseq;

//***** Stage4 part *****
CCSprite
*Ballone,*Balltwo,*Ballthree,*Ballfour,*Ballfive,*Ballsix
;
CCSprite *BallIn;
CCSprite *BodyShrimp;

int counthitupdown_st4;
int bodyHP;
int frameCountt;
int Counttt;
int smalll;

//***** Stage5 part *****
CCSprite *background2;
CCSprite *potSprite1,*lemon,*waterLemon;

float oldDistance;
float newDistance;
int count_st5;
int check;

//***** Stage6 part *****
CCSprite *bar1,*arrow1,*bar2;
CCSprite *Bar[10];
CCSpriteSheet *rectSheet1;
CCSprite *potSprite1_st6;
CCSprite *inputfood[6],*inputfood2;
CCSprite *slicelemongrass[4];
CCSprite *slicemushroom[4];
CCSprite *shrimp[3];
CCSprite *chilly[3];
CCSprite *potSprite2;

```

## โปรแกรม ก.27 ไฟล์ Multiplayer.h (ต่อ)

```

    CCSprite *KaffirLime[3]
    CCParticleSystem *splasheffect2,*splasheffect3;
    int step;
    BOOL success[6];
    CCParticleSystem *FireEffect[2];

    /***** Stage result part *****/
    CCSprite *BGresult;
    CCLabel *GameResultLabel;
    NSString *resultchar;
    CCSprite *backbutton,*backbutton2;
}
// returns a Scene that contains the HelloWorld as the
// only child
+(id) scene;
@property(nonatomic) NSInteger
state;
@property(nonatomic, copy) NSString *name;
@property(nonatomic) NSInteger
role;
@property(nonatomic) NSInteger
uniqueID;
@property(nonatomic, copy) NSString
*sessionId;
@property(nonatomic, retain) GKSession
*session;
// remote peer
@property(nonatomic, copy) NSString
*peerID;
@property(nonatomic) NSInteger
peerUniqueID;
@property(nonatomic, copy) NSString
*peerName;
@property(nonatomic, retain) UIAlertView
*connectionAlert;
@property(nonatomic, retain) id<DataReceiver>
dataReceiver;
- (void)startPicker;
- (void)reset;
- (void)resync;
- (void)sendPacket:(int)packetID
objectIndex:(int)objectIndex data:(void *)data
length:(int)length reliable:(bool)howtosend;
- (void)invalidateSession;
- (void)stagemull;
@end

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m

```

// Multiplayer.m
// ThaiARoi
//
#import "Multiplayer.h"
#import "SinglePlayer.h"
#import "SimpleAudioEngine.h"
#import "CDAudioManager.h"
#import "CocosDenshion.h"
#import "SinglePlayer.h"
typedef enum {
    StateDisconnected,
    StatePicker,
    StateReset,
    StateCointoss,
    StateConnected,
    StateReconnect
} LinkStates;

@implementation Multiplayer
@synthesize state, role, session, name, sessionID,
peerID, peerName, connectionAlert, dataReceiver,
uniqueID, peerUniqueID;

+(id) scene
{
    CCScene *scene = [CCScene node];
    Multiplayer *layer = [Multiplayer node];
    [scene addChild:layer];
    return scene;
}

-(id) init
{
    if ((self = [super init])) {
        self.isTouchEnabled = YES;

        StageNumber=0;
        isWin = RESULT_WIN;

        BGmulti = [CCSprite
spriteWithFile:@"BGmulti.png"];
        BGmulti.position = ccp(240,160);
        [self addChild:BGmulti];

        spred = [CCSprite spriteWithFile:@"red.png"];
        spred.position = ccp(240,160);
        [spred runAction:[CCHide action] ];
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [self addChild:spreed z:5];
        SendPacket = [CCLabel labelWithString:@"Send
Packet" fontName:@"Marker Felt" fontSize:64];
        SendPacket.position = ccp(200,200);
        [self addChild:SendPacket];
        [SendPacket runAction:[CCHide action]];
        RecPacket = [CCLabel labelWithString:@"Received
Packet" fontName:@"Marker Felt" fontSize:64];
        RecPacket.position = ccp(280,50);
        [self addChild:RecPacket];
        [RecPacket runAction:[CCHide action]];
        item1 = [CCMenuItemImage
itemFromNormalImage:@"bally.png"
selectedImage:@"bally.png" target:self
selector:@selector(doThis:)];
        item1.position = ccp(-50,0);
        CCMenu *menu = [CCMenu menuWithItems:item1,
nil];
        [self addChild:menu z:1];
        backButton = [CCSprite
spriteWithFile:@"back_button.png"];
        backButton.position = ccp(430,290);
        [self addChild:backbutton z:3];
        [self startPicker];
    }
    return self;
}
-(void)doThis:(id)sender
{
    if(session != nil)
    {
        NSLog(@"doThis function");
        int testnumber = 1;
        if (self.role == RoleServer) {
            [self sendPacket:MUL_CREATED objectIndex:0
data:&testnumber length:sizeof(int) reliable:YES];
            [SendPacket runAction:[CCShow action]];
            [self stagemull];
        }
    }
}

-(void)onEnter
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    [super onEnter];
}
-(void)onEnterTransitionDidFinish
{
    [super onEnterTransitionDidFinish];
}
-(void)onExit
{
    [super onExit];
}
// on "dealloc" you need to release all your retained
objects
- (void)dealloc {
    if(self.connectionAlert &&
self.connectionAlert.visible) {
        [self.connectionAlert
dismissWithClickedButtonIndex:-1 animated:NO];
    }
    self.connectionAlert = nil;
    [self invalidateSession];
    self.peerName = nil;
    self.peerID = nil;
    self.name = nil;
    self.sessionID = nil;
    NSLog(@">>> dealloc at Multiplayer.m");

    [super dealloc];
}
- (void)invalidateSession {
    if(session != nil) {
        NSLog(@">>> invalidateSession at
Multiplayer.m");
        self.dataReceiver = nil;
        [session disconnectFromAllPeers];
        session.available = NO;
        [session setDataReceiveHandler: nil
withContext: NULL];
        session.delegate = nil;
        self.session = nil;
    }
}
- (void)setGameState:(NSInteger)newState {
    if(newState == StateDisconnected) {
        if(self.session) {
            // invalidate session and release it.
            [self invalidateSession];
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        state = newState;
    }

-(void)startPicker {
    GKPeerPickerController* picker;

    self.state = StatePicker;
    picker = [[GKPeerPickerController alloc] init]; //
note: picker is released in various picker delegate
methods when picker use is done.
    picker.delegate = self;
    //picker.disconnectTimeout = 30;
    [picker show]; // show the Peer Picker
}

-(void)peerPickerControllerDidCancel:(GKPeerPickerController *)picker {
    NSLog(@" >>> peerPickerControllerDidCancel");
    // Peer Picker automatically dismisses on user
cancel. No need to programmatically dismiss.
    // autorelease the picker.
    picker.delegate = nil;
    [picker autorelease];
    // invalidate and release game session if one is
around.
    if(self.session != nil) {
        [self invalidateSession];
    }
    state = StateDisconnected;
    [delegate linkDisconnected];
}

// Provide a custom session that has a custom session ID.
This is also an opportunity to provide a session with a
custom display name.
- (GKSession *)peerPickerController:(GKPeerPickerController *)picker
sessionForConnectionType:(GKPeerPickerConnectionType)type
{
    NSLog(@" >>> sessionForConnectionType");

    GKSession *_session = [[GKSession alloc]
initWithSessionID:sessionID displayName:name
sessionMode:GKSessionModePeer];
    _session.disconnectTimeout = 40;
    return [_session autorelease]; // peer picker
retains a reference, so autorelease ours so we don't
leak.
}

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

- (void)peerPickerController:(GKPeerPickerController
*)picker didConnectPeer:(NSString *)_peerID
toSession:(GKSession *)_session {
    NSLog(@" >>> didConnectPeer");

    // Remember the current peer.
    self.peerID = _peerID;

    // Make sure we have a reference to the game session
and it is set up
    self.session = _session; // retain
    self.session.delegate = self;

    self.peerName = [session displayNameForPeer:peerID];

    [self.session setDataReceiveHandler:self
withContext:nil];

    // Done with the Peer Picker so dismiss it.
    [picker dismiss];
    picker.delegate = nil;
    [picker autorelease];

    // Start Multiplayer game by entering a cointoss
state to determine who is server/client.
    self.state = StateCointoss;
    [NSTimer scheduledTimerWithTimeInterval:0.033
target:self selector:@selector(cointoss) userInfo:nil
repeats:NO];
}
- (void)cointoss {
    NSLog(@" >>> cointoss");
    [self sendPacket:PACKET_COINTOSS objectIndex:0
data:&uniqueID length:sizeof(int) reliable:YES];
}
- (void)reset {
    NSLog(@">>> At reset");
    self.dataReceiver = nil;
    self.state = StateReset;
}

- (void)resync {
    NSLog(@">>> At resync");
    [self cointoss];

    if (self.state == StateReset) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        NSString *message = [NSString
stringWithFormat:@"Waiting for %@", [session
displayNameForPeer:peerID]];
        UIAlertView *alert = [[UIAlertView alloc]
initWithTitle:@"Replay" message:message delegate:self
cancelButtonTitle:@"End Game" otherButtonTitles:nil];

        self.connectionAlert = alert;
        [alert show];
        [alert release];
        state = StateCointoss;
    } else {
        // cointoss packet already received
        [delegate linkConnected:role];
    }
}

- (void)receiveData:(NSData *)data fromPeer:(NSString
*)peer inSession:(GKSession *)session context:(void
*)context {
    NSLog(@" >>> receiveData");
    const unsigned int packetHeaderSize = 3 *
sizeof(int); // we have two "ints" for our header
    static int lastPacketTime = -1;
    unsigned char *incomingPacket = (unsigned char
*)[data bytes];
    int *pIntData = (int *)&incomingPacket[0];
    // check the network time and make sure packers are
in order
    int packetTime = pIntData[0];
    int packetID = pIntData[1];
    int objectIndex = pIntData[2];
    lastPacketTime = packetTime;
    switch( packetID ) {
        case PACKET_COINTOSS:
            {
                NSLog(@" >>> ReceivePacketID =
PACKET_COINTOSS");
                // coin toss to determine roles of the two
players
                peerUniqueID = pIntData[3];
                // if other player's coin is higher than
ours then that player is the server
                if(peerUniqueID > uniqueID) {
                    self.role = RoleClient;
                }
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        if (self.state == StateCointoss) {
            if(self.connectionAlert &&
self.connectionAlert.visible) {[self.connectionAlert
dismissWithClickedButtonIndex:-1 animated:NO];
            }
            [delegate linkConnected: self.role];
        }
        self.state = StateConnected;
    }
    break;
case MUL_CREATED:
    NSLog(@">>> I got MUL_CREATED");
    [self stagemull];
    break;
case MUL_STAGE1:
    NSLog(@">>> I got MUL_STAGE1");
    if (pIntData[3] ==0) {
        NSLog(@">>> Start peel shirmp");
    }
    else if(pIntData[3] ==1)
    {
        NSLog(@">>> First shirmp is
finished");
    }
    else if(pIntData[3] ==2)
    {
        NSLog(@">>> Second shirmp is
finished");
    }
    else if(pIntData[3] ==3)
    {
        NSLog(@">>> Third shirmp is
finished");
        barstage.percentage += 17;
    } break;
case MUL_STAGE2:
    NSLog(@">>> I got MUL_STAGE2");
    if (pIntData[3] ==0)
    {
        NSLog(@">>> Start stage2");
    }
    else if(pIntData[3] ==1)
    {
        NSLog(@">>> First piece");
    }
    else if(pIntData[3] ==2)
    {
        NSLog(@">>> Second piece");
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    }
    else if(pIntData[3] ==3)
    {
        NSLog(@">>> Third piece");
    }
    else if(pIntData[3] ==4)
    {
        NSLog(@">>> Fourth piece");
    }
    else if(pIntData[3] ==5)
    {
        NSLog(@">>> Fifth piece");
    }
    else if(pIntData[3] ==6)
    {
        NSLog(@">>> Sixth piece");
    }
    else if(pIntData[3] ==7)
    {
        NSLog(@">>> Seven piece");
    }
    else if(pIntData[3] ==8)
    {
        NSLog(@">>> Eighth piece");
    }
    else if(pIntData[3] ==9)
    {
        NSLog(@">>> Nineth piece");
    }
    else if(pIntData[3] ==10)
    {
        NSLog(@">>> Tenth piece");
        barstage.percentage += 17;
    }
    break;
case MUL_STAGE3:
    NSLog(@">>> I got MUL_STAGE3");
    if (pIntData[3] ==0)
    {
        NSLog(@">>> Start stage3");
    }
    else if(pIntData[3] ==2)
    {
        NSLog(@">>> First piece");
    }
    else if(pIntData[3] ==4)
    {
        NSLog(@">>> Second piece");
    }
    else if(pIntData[3] ==6)
    {
        NSLog(@">>> Third piece");
    }
    else if(pIntData[3] ==8)
    {
        NSLog(@">>> Fourth piece");
    }
    else if(pIntData[3] ==10)
    {
        NSLog(@">>> Fifth piece");
        barstage.percentage += 17;
    }

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        break;
    case MUL_STAGE4:
        NSLog(@">>> I got MUL_STAGE4");
        if (pIntData[3] ==0)
        {
            NSLog(@">>> Start stage4");
        }
        else if(pIntData[3] ==1)
        {
            NSLog(@">>> First Mushroom");
        }
        else if(pIntData[3] ==2)
        {
            NSLog(@">>> Second Mushroom");
        }
        else if(pIntData[3] ==3)
        {
            NSLog(@">>> Third Mushroom");
            barstage.percentage += 17;
        }
        break;
    case MUL_STAGE5:
        NSLog(@">>> I got MUL_STAGE5");
        if (pIntData[3] ==0)
        {
            NSLog(@">>> Start stage5");
        }
        else if(pIntData[3] ==1)
        {
            NSLog(@">>> First try");
        }
        else if(pIntData[3] ==2)
        {
            NSLog(@">>> Second try");
        }
        else if(pIntData[3] ==3)
        {
            NSLog(@">>> Third try");
        }
        else if(pIntData[3] ==4)
        {
            NSLog(@">>> Fourth try");
            barstage.percentage += 17;
        }
        break;
    case MUL_STAGE6:
        NSLog(@">>> I got MUL_STAGE6");
        if (pIntData[3] ==0)
        {
            NSLog(@">>> Ready for input 1");
        }
        else if(pIntData[3] ==1)
        {
            NSLog(@">>> Ready for input 2");
        }
        else if(pIntData[3] ==2)

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        {
            NSLog(@">>> Ready for input 3");
        }
        else if(pIntData[3] ==3)
        {
            NSLog(@">>> Ready for input 4");
        }
        else if(pIntData[3] ==4)
        {
            NSLog(@">>> Ready for input 5");
        }
        else if(pIntData[3] ==5)
        {
            NSLog(@">>> Ready for input 6");
        }
        else if(pIntData[3] ==6)
        {
            NSLog(@">>> Out of input food");
            barstage.percentage += 15;
        }
        if (StageNumber <7)
        {
            [self fadeinandout:spreed];
        } break;
        case MUL_STAGE_FIN_RESULT:
            NSLog(@">>> I got MUL_STAGE_FIN_RESULT.");
            if (pIntData[3]==RESULT_WIN) {
                NSLog(@">>> You LOSE");
                isWin = RESULT_LOSE;
            } break;
        case MUL_STAGE_FIN_SCORES:
            NSLog(@">>> I got MUL_STAGE_FIN_SCORES.
Your enemy score is %d",pIntData[3]);
            break;
        default:
            if (dataReceiver) {
                [dataReceiver receivePacket:packetID
objectIndex:objectIndex
data:&incomingPacket[packetHeaderSize]];
            } else {
                NSLog(@" !!! receiveData PACKET
BEFORE COINTOSS: %d", packetID);
            }
        }
    }
- (void)sendPacket:(int)packetID objectIndex:(int)objectIndex
data:(void *)data length:(int)length reliable:(bool)howtosend
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    NSLog(@">>> sendPacket");
    // the packet we'll send is resued
    static unsigned char networkPacket[MAX_PACKET_SIZE];
    const unsigned int packetHeaderSize = 3 *
sizeof(int); // we have three "ints" for our header

    if(length < (MAX_PACKET_SIZE - packetHeaderSize)) {
// our networkPacket buffer size minus the size of the
header info
        int *pIntData = (int *)&networkPacket[0];
        // header info
        pIntData[0] = packetNumber++;
        pIntData[1] = packetID;
        pIntData[2] = objectIndex;

        memcpy( &networkPacket[packetHeaderSize], data,
length );

        NSData *packet = [NSData dataWithBytes:
networkPacket length: (length+packetHeaderSize)];
        if(howtosend == YES) {
            [session sendDataToAllPeers:packet
withDataMode:GKSendDataReliable error:nil];
        } else {
            [session sendDataToAllPeers:packet
withDataMode:GKSendDataUnreliable error:nil];
        }
    }
}

- (void)session:(GKSession *)_session peer:(NSString
*)_peerID didChangeState:(GKPeerConnectionState)_state {

    NSLog(@" >>> session:didChangeState");

    if(self.state == StatePicker) {
        return; // only do stuff if
we're in multiplayer, otherwise it is probably for Picker
    }

    if(_state == GKPeerStateDisconnected) {
        NSString *message = [NSString
stringWithFormat:@"%@" has disconnected.", [_session
displayNameForPeer:_peerID]];
        if ((self.state == StateCointoss) &&
self.connectionAlert && self.connectionAlert.visible) {
            self.connectionAlert.message = message;

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    }
    else {
        UIAlertView *alert = [[UIAlertView alloc]
initWithTitle:@"Lost Connection" message:message
delegate:self cancelButtonTitle:@"End Game"
otherButtonTitles:nil];
        self.connectionAlert = alert;
        [alert show];
        [alert release];
    }

    self.state = StateDisconnected;

    [delegate linkDisconnected];
}
}
- (void)alertView:(UIAlertView *)alertView
clickedButtonAtIndex:(NSInteger)buttonIndex {
    if(buttonIndex == 0) {
        if (self.state == StateCointoss) {
            [delegate linkDisconnected];
        }
        self.state = StateDisconnected;
    }
    if (StageNumber==1) {
        NSLog(@"Time to stage2");
        [self stagemul2];
    }
    else if (StageNumber==2) {
        NSLog(@"Time to stage3");
        [self stagemul3];
    }
    else if (StageNumber==3) {
        NSLog(@"Time to stage4");
        [self stagemul4];
    }
    else if (StageNumber==4) {
        NSLog(@"Time to stage5");
        [self stagemul5];
    }
    else if (StageNumber==5) {
        NSLog(@"Time to stage6");
        [self stagemul6];
    }
    else if (StageNumber==6)
    {
        NSLog(@"Time to result stage");
        [self resultstage];
    }
}

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    }
}

-(void) stagemull
{
    NSLog(@">>> stagemull function");

    [[SimpleAudioEngine
sharedEngine]playEffect:@"quickpop.mp3"];

    //[[CCDirector
sharedDirector]replaceScene:[StageMull scene]];

    [SendPacket runAction:[CCHide action]];
    [RecPacket runAction:[CCHide action]];
    item1.position = ccp(500,500);
    [BGmulti runAction:[CCHide action]];
    backbutton.position = ccp(1430,1290);

    /***init stagel
    StageNumber=1;
    background = [CCSprite
spriteWithFile:@"background1_3.png"];
    background.position = ccp(240,160);
    [self addChild:background];
    stepOne = FALSE;
    stepTwo = FALSE;
    stepThree = FALSE;
    counthitupdown = 1;
    headHP = 5;
    count =0;
    number=0;
    [self initPeel];
    [self schedule:@selector(checkLeg:)];
    //Progress Timer
    timer = [CCProgressTimer
progressWithFile:@"circle2.png"];
    timer.type = kCCProgressTimerTypeRadialCCW;
    timer.percentage = 0;
    timer.position = ccp(60,240);
    [self addChild:timer z:3];
    [self schedule:@selector(TimerTest:) interval:1];
    bar = [CCProgressTimer
progressWithFile:@"gauge2.png"];
    bar.type = kCCProgressTimerTypeHorizontalBarLR;
    bar.percentage = 0;
    bar.position = ccp(300,300);
    [self addChild:bar z:3];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        whitecircle = [CCSprite
spriteWithFile:@"WhiteCircle.png"];
        whitecircle.position = ccp(60,160);
        [self addChild:whitecircle z:3];

        //progressbar
        barstage = [CCProgressTimer
progressWithFile:@"RedCircle.png"];
        barstage.type = kCCProgressTimerTypeRadialCW;
        barstage.percentage = 0;
        barstage.position = ccp(60,160);
        [self addChild:barstage z:3];

        [self sendPacket:MUL_STAGE1 objectIndex:0
data:&count length:sizeof(int) reliable:YES];

        [[SimpleAudioEngine
sharedEngine]playBackgroundMusic:@"playmusic4.mp3"];
    }
    //Schedule Update Timer
    -(void)TimerTest:(ccTime)delta
    {
        timer.percentage += (delta*1.67);
        CCLOG(@" percent %f",timer.percentage);

        timer1 += delta;
        CCLOG(@" second %f",timer1);

        if (timer1>=60) {
            timer1 =0;
        }

        if (timer.percentage>=100) {
            timer.percentage =0;
        }
        CCLOG(@"%f",delta);
    }
    -(void)initPeel
    {
        body = [CCSprite spriteWithFile:@"peel_body.png"];
        head = [CCSprite spriteWithFile:@"peel_head.png"];
        shield = [CCSprite spriteWithFile:@"peel_sl.png"];
        leg = [CCSprite spriteWithFile:@"peel_leg.png"];
        body.position = ccp(600,164);
    }

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    head.position = ccp(620,180);
    head.anchorPoint = ccp(0.9,0.5);
    shield.position = ccp(600,188);
    leg.position = ccp(600,160);

    [self addChild:body];
    [self addChild:head];
    [self addChild:shield z:1];
    [self addChild:leg];

    [self jumpPeel];
}

-(void) jumpPeel
{
    [body runAction:[CCJumpTo actionWithDuration:0.5
position:ccp(350,80) height:100 jumps:1]];
    [head runAction:[CCJumpTo actionWithDuration:0.5
position:ccp(300,95) height:100 jumps:1]];
    [shield runAction:[CCJumpTo actionWithDuration:0.5
position:ccp(335,105) height:100 jumps:1]];
    [leg runAction:[CCJumpTo actionWithDuration:0.5
position:ccp(335,78) height:100 jumps:1]];
}

-(void) checkLeg:(ccTime)delta
{
    //CCLOG(@"%f",leg.position.y);
    if (leg.position.y<0&&body.position.x==--400) {
        [self Finish];
    }
}

-(void) Finish
{
    body.position = ccp(600,164);
    head.position = ccp(620,180);
    shield.position = ccp(600,188);
    leg.position = ccp(600,170);
    stepOne = FALSE;
    stepTwo = FALSE;
    stepThree = FALSE;
    counthitupdown = 1;
    headHP = 5;
    count++;
    if (count==1) {
        bar.percentage +=30;

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [self sendPacket:MUL_STAGE1 objectIndex:0
data:&count length:sizeof(int) reliable:YES];
    }
    if (count==2) {
        bar.percentage +=30;
        [self sendPacket:MUL_STAGE1 objectIndex:0
data:&count length:sizeof(int) reliable:YES];
    }
    if (count==3) {
        bar.percentage +=40;
        [self sendPacket:MUL_STAGE1 objectIndex:0
data:&count length:sizeof(int) reliable:YES];
        [[SimpleAudioEngine
sharedEngine]stopBackgroundMusic];
        UIAlertView *alert = [[UIAlertView alloc]init];
        [alert setTitle:@"à,-à,³à,,à,±à¹%à,™à,•à,à,™ 1
à¹Cà,ªà,£à¹+à,^à¹ à,¥à¹%à,$"];
        [alert setDelegate:self];
        [alert addButtonWithTitle:@"à,•à, à,¥à,±"];
        [alert show];
        [alert release];
    }
    [self jumpPeel];
}
-(void)ccTouchesBegan:(NSSet *)touches withEvent:(UIEvent
*)event
{
    if (StageNumber==1) {
        UITouch *touch = [touches anyObject];
        CGPoint point = [touch locationInView:[touch
view]];
        point = [[CCDirector
sharedDirector]convertToGL:point];

        if
((point.x>=150) && (point.x<=280) && (point.y>=90) && (point.y<
=200)) {
            number = 1;
            CCLOG(@"head");
        }
        else
if ((point.x>=280) && (point.x<=400) && (point.y>=90) && (point.
y<=180)) {
            number = 2;
            CCLOG(@"shield");
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        else
if ((point.x>=280) && (point.x<=370) && (point.y>=40) && (point.
y<=90)) {
            number = 3;
            CCLOG(@"leg");
        }
    }
else if(StageNumber ==5)
{
    NSSet *set = [event allTouches];

    if(set.count == 2)
    {
        UITouch *one = [[set
allObjects]objectAtIndex:0];
        UITouch *two = [[set
allObjects]objectAtIndex:1];
        CGPoint ptOne = [one locationInView:[one view]];
        CGPoint ptTwo = [two locationInView:[two view]];
        ptOne = [[CCDirector sharedDirector]convertToGL:ptOne];
        ptTwo = [[CCDirector sharedDirector]convertToGL:ptTwo];

        oldDistance = ccpDistance(ptOne, ptTwo);
        CCLOG(@" oldDistance = %f",oldDistance);
    }

}
else if(StageNumber ==7 || StageNumber==0)
{
    UITouch *touch = [touches anyObject];
    CGPoint point = [touch locationInView:[touch
view]];
    point = [[CCDirector
sharedDirector]convertToGL:point];

    if ((point.x>=400) && (point.x<=450) && (point.y>=270) && (
point.y<=320)) {
        CCFadeTransition *tran = [CCFadeTransition
transitionWithDuration:2 scene:[SinglePlayer scene]
withColor:ccWHITE];
        [[CCDirector
sharedDirector]replaceScene:tran];
    }
}
}
}

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

-(void) ccTouchesMoved: (NSSet *) touches withEvent: (UIEvent
*) event
{
    if (StageNumber==1) {
        if (number==1&&!stepOne) {
            UITouch *touch = [touches anyObject];
            CGPoint point = [touch
locationInView:[touch view]];
            point = [[CCDirector sharedDirector]
convertToGL:point];
            float angleRadians =
atanf((float) (point.y-head.position.y)/(float) (point.x-
head.position.x));
            float angleDegrees =
CC_RADIANS_TO_DEGREES(angleRadians);
            float cocosAngle = -1 * angleDegrees;
            if(cocosAngle >= -60 && cocosAngle <= 60
&& point.x <= 280)
                head.rotation = cocosAngle;
            if(cocosAngle <= -40 && counthitupdown ==
1)
                {
                    headHP--;
                    counthitupdown = 2;
                    [[SimpleAudioEngine sharedEngine]
playEffect:@"march22.mp3"];
                }
            else if(cocosAngle >= 40 && counthitupdown
== 2)
                {
                    counthitupdown = 1;
                    [[SimpleAudioEngine sharedEngine]
playEffect:@"march22.mp3"];
                }

            if(headHP <= 0)
                {
                    head.rotation = 0;
                    [[SimpleAudioEngine sharedEngine]
playEffect:@"guncock.mp3"];
                    [head runAction:[CCMoveBy
actionWithDuration:0.5 position:ccp(-400,0)]];
                    stepOne = TRUE;
                }
        }
        else if (number==2 && stepOne) {
            UITouch *touch = [touches anyObject];
            CGPoint point = [touch
locationInView:[touch view]];

```



## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

[[SimpleAudioEngine
sharedEngine] playEffect:@"sliceup.mp3"];
    }
    }
    else if(point.y < 50)
    {
        knife.position =
ccp(knife.position.x,50);
        if(counthitupdown_st2 ==1)
            counthitupdown_st2 =2;
        [[SimpleAudioEngine sharedEngine]
playEffect:@"slicedown.mp3"];
    }

    if(lemongrassHP ==0)
    {
        [knife runAction:[CCMoveBy
actionWithDuration:.2 position:ccp(-15,0)]];
        [[SimpleAudioEngine sharedEngine]
playEffect:@"shortchop.mp3"];
        //à¹€à,žà,à¹`à;ProgressBar
        healthValue++;
        [self stepup:healthValue];
        [self lemonGiscut:healthValue];
        //NSLog(@"HealthValue
%d",healthValue);
        globalserie++;
        lemongrassHP =2;
    }
    }
    else if(StageNumber ==3)
    {
        UITouch *touch = [touches anyObject];
        CGPoint point = [touch locationInView:[touch
view]];
        point = [[CCDirector sharedDirector]
convertToGL:point];
        //knife can move only 90>point.y>130
        if(globalserie <= 10)
        {
            knife.position =
ccp(knife.position.x,point.y);
            if(point.y > 130)
            {
                knife.position =
ccp(knife.position.x,130);
                if(counthupdown ==2)

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        {
            counthupdown = 1;
            lemon_grassHP--;
            [[SimpleAudioEngine
sharedEngine] playEffect:@"sliceup.mp3"];
        }
    }
    else if(point.y < 70)
    {
        knife.position =
ccp(knife.position.x, 70);
        if(counthupdown ==1)
            counthupdown =2;
        [[SimpleAudioEngine sharedEngine]
playEffect:@"slicedown.mp3"];
    }

    if(lemon_grassHP ==0)
    {
        if(cutseq ==1)
        {
            id actionHide = [CCHide action];
            [galingale
runAction:actionHide];
            [self cutgalingale:1];

            id actionShow = [CCShow action];
            id actiondelaytime =
[CCDelayTime actionWithDuration:2*0.4f];
            [galingale runAction:[CCSequence
actions: actiondelaytime,actionShow,nil]];
            galingale.texture
=[[CCTextureCache sharedTextureCache]
addImage:@"galB.png"];

            [[SimpleAudioEngine
sharedEngine] playEffect:@"shortchop.mp3"];
            cutseq++;
        }
        else if(cutseq==2)
        {
            id actionHide = [CCHide action];
            [galingale
runAction:actionHide];

            [self cutgalingale:2];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        id actionShow = [CCShow action];
        id actiondelaytime =
[CCDelayTime actionWithDuration:2*0.4f];
        [galingale runAction:[CCSequence
actions: actiondelaytime,actionShow,nil]];
        galingale.texture
=[ [CCTextureCache sharedTextureCache]
addImage:@"galC.png"];
        [[SimpleAudioEngine
sharedEngine] playEffect:@"shortchop.mp3"];
        cutseq++;
    }
    else if(cutseq==3)
    {
        id actionHide = [CCHide action];
        [galingale
runAction:actionHide];
        [self cutgalingale:3];
        id
actionShow = [CCShow action];
        id actiondelaytime =
[CCDelayTime actionWithDuration:2*0.4f];
        [galingale runAction:[CCSequence
actions: actiondelaytime,actionShow,nil]];
        galingale.texture
=[ [CCTextureCache sharedTextureCache]
addImage:@"galD.png"];
        [[SimpleAudioEngine
sharedEngine] playEffect:@"shortchop.mp3"];
        cutseq++;
    }
    else if(cutseq==4)
    {
        id actionHide = [CCHide action];
        [galingale
runAction:actionHide];
        [self cutgalingale:4];
        id actionShow = [CCShow action];
        id actiondelaytime =
[CCDelayTime actionWithDuration:2*0.4f];
        [galingale runAction:[CCSequence
actions: actiondelaytime,actionShow,nil]];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        galingale.texture
    =[[CCTextureCache sharedTextureCache]
    addImage:@"galE.png"];
        [[SimpleAudioEngine
    sharedEngine] playEffect:@"shortchop.mp3"];
        cutseq++;
    }
    else if(cutseq==5)
    {
        id actionHide = [CCHide action];
        [galingale
    runAction:actionHide];

        [self cutgalingale:5];

        id actionShow = [CCShow action];
        id actiondelaytime1 =
    [CCDelayTime actionWithDuration:2*0.4f];
        id actiondelaytime2 =
    [CCDelayTime actionWithDuration:2*0.4f];
        galingale.texture
    =[[CCTextureCache sharedTextureCache]
    addImage:@"galangal6left.png"];
        rightgal.texture
    =[[CCTextureCache sharedTextureCache]
    addImage:@"galangal6right.png"];

        [[SimpleAudioEngine
    sharedEngine] playEffect:@"shortchop.mp3"];

        [galingale runAction:[CCSequence
    actions: actiondelaytime1,actionShow,nil]];
        [rightgal runAction:[CCSequence
    actions: actiondelaytime2,nil]];

        cutseq++;
    }
    else if(cutseq==6)
    {
        CCLOG(@"");
    }
    else {
        CCLOG(@"Integer out of range");
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

                                                                    [knife
runAction:[CCMoveBy actionWithDuration:.2 position:ccp(-
20,0)]];

                                                                    healthPoint =
healthPoint +2;
                                                                    [self stepup_st3:healthPoint];
                                                                    globalserie = globalserie +2;
                                                                    lemon_grassHP =5;
                                                                    if(globalserie >=10)
                                                                    [knife runAction:[CCFadeOut
actionWithDuration:0.1]];
                                                                    }
                                                                    }
                                                                    }
                                                                    else if(StageNumber==4)
                                                                    {
                                                                    UITouch *touch = [touches anyObject];
                                                                    CGPoint point = [touch locationInView:[touch
view]];
                                                                    point =[[[CCDirector sharedDirector]
convertToGL:point]];
                                                                    //for (int yy = 0; yy <=2; yy++)
                                                                    if(Counttt < 3 ) {
                                                                    id actionshow=[CCShow action];
                                                                    if(frameCountt > 0 )
                                                                    {
                                                                    Balltwo.position = ccp(540,200);
                                                                    [Balltwo runAction:actionshow];
                                                                    [BallIn runAction:actionshow];
                                                                    bodyHP=3;
                                                                    [Balltwo runAction:[CCJumpBy
actionWithDuration:.5 position:ccp(-240,-60) height:100
jumps:1]];
                                                                    BallIn.position =ccp(300,140);
                                                                    //CCLOG(@"test000");
                                                                    frameCountt = 0;
                                                                    //break;
                                                                    }
                                                                    //CCLOG(@"test9999999");
                                                                    BallIn.position = ccp(point.x, point.y);
                                                                    if((BallIn.position.y <=110 &&
BallIn.position.y >=100) && (BallIn.position.x>280 &&
BallIn.position.x < 320) && bodyHP == 3)
                                                                    {
                                                                    bodyHP--;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        counthitupdown_st4 = 1;
    }
    if((BallIn.position.y <=100 &&
BallIn.position.y >=80) && (BallIn.position.x>280 &&
BallIn.position.x < 320) && bodyHP == 2)
    {
        bodyHP--;
        counthitupdown_st4 = 1;
    }
    if((BallIn.position.y < 60) &&
(BallIn.position.x>280 && BallIn.position.x < 320) &&
bodyHP == 1)
    {
        bodyHP--;
        counthitupdown_st4 = 0;
    }

    if(bodyHP == 0 && counthitupdown_st4 == 0)
    {
        CCSpriteSheet *galSheet1_st4 =
[CCSpriteSheet spriteSheetWithFile:@"mushroomm.png"];
[self addChild:galSheet1_st4 z:2];
        counthitupdown_st4 = 1;
        frameCounttt++;
        Countttt++;
        id actionHide = [CCHide action];
        [Balltwo runAction:actionHide];

        // create the sprite
        CCSprite *galSpritel1_st4 = [CCSprite
spriteWithTexture:galSheet1_st4.texture
rect:CGRectMake(0, 450, 225, 150)];
        [galSheet1_st4
addChild:galSpritel1_st4 z:2];
        [galSpritel1_st4
runAction:actionshow];
        // position the sprite in the center
of the screen
        // position the sprite in the center
of the screen
        //CGSize s = [[CCDirector
sharedDirector] winSize];
        //galSpritel1_st4.position =
ccp(s.width/2,s.height/2);
        galSpritel1_st4.position =
ccp(300,140);

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        // create the animation
        CCAnimation *galAnimation1_st4 =
[CCAnimation animationWithName:@"cutgal" delay:0.1f];

        int frameCount = 0;
        for (int y = 0; y <=3; y++) {
            CCSpriteFrame *frame1 =
[CCSpriteFrame frameWithTexture:galSheet1_st4.texture
rect:CGRectMake(0,y*150,225,150) offset:ccp(0,0)];
            [galAnimation1_st4
addFrame:frame1];

            frameCount++;

            if (frameCount == 4)
                break;
        }
        [[SimpleAudioEngine sharedEngine]
playEffect:@"METLHIT.mp3"];

        // create the action
        CCAnimate *galAction1_st4 =
[CCAnimate actionWithAnimation:galAnimation1_st4];
        // run the action
        //[galSprite1_st4
runAction:galAction1_st4];

        //[Balltwo runAction:[CCMoveBy
actionWithDuration:.5 position:ccp(0,200)]];
        //globalseries++;

        [self
sendPacket:MUL_STAGE4 objectIndex:0 data:&Counttt
length:sizeof(int) reliable:YES];

        if (small1 == 1)
        {
            Ballone.position =
ccp (small1*25,25);
            //Ballfive.position =
ccp ((small1+1)*25,25);
        }
        if (small1 == 2)
        {
            Ballthree.position =
ccp (small1*25,25);
            //Ballfive.position =
ccp ((small1+1)*25,25);

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    }
    if (small1 == 3)
    {
        Ballfour.position =
ccp(small1*25,25);
        //[Ballfive
runAction:actionHide];
        //Counttt = 0;
        /*counthitupdown_st4 = 1; //for
count hit up and down. 1 mean already hitup.2 mean
already hitdown
        bodyHP = 3; //for count hit up
and down. 1 mean already hitup.2 mean already hitdown
        frameCountt = 0;
        Counttt = 0;
        small1 = 0;*/
        [[SimpleAudioEngine
sharedEngine]stopBackgroundMusic];
        UIAlertView *alert =
[[UIAlertView alloc]init];
        [alert setTitle:@"à,-
à,³à, ,à,±à¹%à,™à,•à,à,™ 4 à¹Cà,ªà,fà¹#à,^à¹ à,¥à¹%à,S"];
        [alert setDelegate:self];
        [alert
addButtonWithTitle:@"à,•à, à,¥à,†"];
        [alert show];
        [alert release];
    }
        small1++;
        //[galSprite1_st4
runAction:actionHide];
        [galSprite1_st4 runAction:[CCSequence
actions:galAction1_st4,actionHide, nil]];

        //Balltwo = [CCSprite
spriteWithTexture:galSheet1.texture rect:CGRectMake(0, 0,
225, 150)];

    }

    else
        BallIn.position =ccp(300,140);

    }
}
else if (StageNumber==5){
    NSSet *set = [event allTouches];

    if(set.count == 2)

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        {
            UITouch *one = [[set
allObjects]objectAtIndex:0];
            UITouch *two = [[set
allObjects]objectAtIndex:1];

            CGPoint pt1 = [one locationInView:[one
view]];
            CGPoint pt2 = [two locationInView:[two
view]];

            pt1 = [[CCDirector
sharedDirector]convertToGL:pt1];
            pt2 = [[CCDirector
sharedDirector]convertToGL:pt2];

            newDistance = ccpDistance(pt1, pt2);
            CCLOG(@"newDistance %f",newDistance);
            [self checkposition];
        }
    }
}

-(void)ccTouchesEnded:(NSSet *)touches withEvent:(UIEvent
*)event
{
    if (StageNumber==5)
    {
        CCLOG(@"End");
        check=1;
    }
}

-(void)stagemul2
{
    NSLog(@">>> stagemul2 function");
    body.position = ccp(1000,1640);
    head.position = ccp(1000,1000);
    shield.position = ccp(1000,1880);
    leg.position = ccp(1000,1600);

    //à¹€à, à¹%à,² stage2
    StageNumber=2;

    //Progress Timer
    timer.percentage = 0;

    //progressbar
    bar.percentage = 0;
    timer1 =0;

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    healthValue =0;
    globalseries = 1;
    lemongrassHP =2;
    counthitupdown_st2 = 1; //for count hit up and down.
1 mean already hitup.2 mean already hitdown

    self.isTouchEnabled = YES;

    knife =[CCSprite spriteWithFile:@"knife.PNG"];
    knife.position =ccp(440,110);
    [self addChild:knife z:2];

    lemongrass = [CCSprite
spriteWithFile:@"lemong0.png"];
    lemongrass.position = ccp(311,110);
    [self addChild:lemongrass z:1];

    arrow = [CCSprite spriteWithFile:@"upanddown.png"];
    arrow.position = ccp(400,110);
    [self addChild:arrow z:3];

    [self sendPacket:MUL_STAGE2 objectIndex:0
data:&healthValue length:sizeof(int) reliable:YES];

    [self fadeInandout_st2:arrow];

    [[SimpleAudioEngine
sharedEngine]playBackgroundMusic:@"playmusic1.mp3"];
}

-(void) fadeInandout_st2:(CCSprite *)obj {

    id fadeout1 = [CCFadeOut actionWithDuration:0.5];

    id fadeIn1 = [CCFadeIn actionWithDuration:0.5];
    [obj runAction:[CCSequence actions:fadeout1,
fadeIn1,fadeout1, fadeIn1,fadeout1, fadeIn1,fadeout1,
nil]];
}

-(void) stepup:(int)num {
    if(num ==1)
    {
        bar.percentage +=10;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 2)
    {
        bar.percentage +=10;
        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 3)
    {
        bar.percentage +=10;
        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 4)
    {
        bar.percentage +=10;
        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 5)
    {
        bar.percentage +=10;
        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 6)
    {
        bar.percentage +=10;
        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 7)
    {
        bar.percentage +=10;
        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 8)
    {
        bar.percentage +=10;
        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 9)
    {
        bar.percentage +=10;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
    }
    else if(num == 10)
    {
        bar.percentage +=10;
        [self sendPacket:MUL_STAGE2 objectIndex:0
data:&num length:sizeof(int) reliable:YES];
        [[SimpleAudioEngine
sharedEngine]stopBackgroundMusic];
        UIAlertView *alert = [[UIAlertView alloc]init];
        [alert setTitle:@"à,-à,³à,,à,±à¹%à,™à,•à,à,™ 2
à¹Cà,ªà,£à¹#à,^à¹ à,¥à¹%à,$"];
        [alert setDelegate:self];
        [alert addButtonWithTitle:@"à,•à, à,¥à,±"];
        [alert show];
        [alert release];
    }
}

-(void) lemongiscut:(int)num2{
    if(num2 == 1)
    {
        lemongrass.texture =[[CCTextureCache
sharedTextureCache] addImage:@"lemong1.png"];
    }
    else if(num2 == 2)
    {
        lemongrass.texture =[[CCTextureCache
sharedTextureCache] addImage:@"lemong2.png"];
    }
    else if(num2 == 3)
    {
        lemongrass.texture =[[CCTextureCache
sharedTextureCache] addImage:@"lemong3.png"];
    }
    else if(num2 == 4)
    {
        lemongrass.texture =[[CCTextureCache
sharedTextureCache] addImage:@"lemong4.png"];
    }
    else if(num2 == 5)
    {
        lemongrass.texture =[[CCTextureCache
sharedTextureCache] addImage:@"lemong5.png"];
    }
    else if(num2 == 6)
    {

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        lemongrass.texture = [[CCTextureCache
sharedTextureCache] addImage:@"lemong6.png"];
    }
    else if(num2 == 7)
    {
        lemongrass.texture = [[CCTextureCache
sharedTextureCache] addImage:@"lemong7.png"];
    }
    else if(num2 == 8)
    {
        lemongrass.texture = [[CCTextureCache
sharedTextureCache] addImage:@"lemong8.png"];
    }
    else if(num2 == 9)
    {
        lemongrass.texture = [[CCTextureCache
sharedTextureCache] addImage:@"lemong9.png"];
    }
    else if(num2 == 10)
    {
        lemongrass.texture = [[CCTextureCache
sharedTextureCache] addImage:@"lemong10.png"];
    }
}
-(void)stagemul3
{
    //Progress Timer
    timer.percentage = 0;

    //progressbar
    bar.percentage = 0;
    timer1 = 0;

    //à¹€à,à¹%à,² stage3
    StageNumber=3;

    //à¹€à,à,²à,,à¹^à,²à¹€à, à¹^à,²à¹,,à, >à,<à¹^à,à,™
    lemongrass.position = ccp(3110,1100);

    healthPoint = 0;
    globalserie = 1;
    lemon_grassHP = 5;
    counthupdown = 1; //for count hit up and down. 1
mean already hitup.2 mean already hitdown
    cutseq = 1;

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    galingale = [CCSprite
spriteWithFile:@"galanga0_test.png"];
    galingale.position = ccp(311,200);
    [self addChild:galingale z:1];

    rightgal = [CCSprite
spriteWithFile:@"galblank.png"];
    rightgal.position = ccp(311,200);
    [self addChild:rightgal z:1];

    knife.position =ccp(galingale.position.x+49,110);

    arrow.position = ccp(400,150);

    [self sendPacket:MUL_STAGE3 objectIndex:0
data:&healthPoint length:sizeof(int) reliable:YES];

    [self fadeinandout_st3:arrow];

    [[SimpleAudioEngine
sharedEngine]playBackgroundMusic:@"playmusic2.mp3"];
}

-(void) fadeinandout_st3:(CCSprite *)obj {
    id fadeout1 = [CCFadeOut actionWithDuration:0.5];
    id fadein1 = [CCFadeIn actionWithDuration:0.5];
    [obj runAction:[CCSequence actions:fadeout1,
fadein1,fadeout1, fadein1,fadeout1, fadein1,fadeout1,
nil]];
}

-(void) cutgalingale: (int)num{
    switch (num) {
        case 1:
            [self cutgalingale1];
            break;
        case 2:
            [self cutgalingale2];
            break;
        case 3:
            [self cutgalingale3];
            break;
    }
}

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        case 4:

            [self cutgalingale4];
            break;
        case 5:

            [self cutgalingale5];
            break;
        default:
            break;
    }
}

-(void) cutgalingale1
{

    CCSpriteSheet *galSheet1 = [CCSpriteSheet
spriteSheetWithFile:@"gal1.png"];
    [self addChild:galSheet1];

    // create the sprite

    CCSprite *galSprite1 = [CCSprite
spriteWithTexture:galSheet1.texture rect:CGRectMake(0,
502, 220, 251)];
    [galSheet1 addChild:galSprite1];

    // position the sprite in the center of the screen
    galSprite1.position = ccp(311,200);

    // create the animation
    CCAanimation *galAnimation1 = [CCAanimation
animationWithName:@"cutgal" delay:0.4f];

    int frameCount = 0;
    for (int y = 0; y <=2; y++) {
        CCSpriteFrame *frame1 = [CCSpriteFrame
frameWithTexture:galSheet1.texture
rect:CGRectMake(0,y*251,220,251) offset:ccp(0,0)];
        [galAnimation1 addFrame:frame1];

        frameCount++;

        if (frameCount == 2)
            break;
    }

    // create the action

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    CCAanimate *galAction1 = [CCAanimate
actionWithAnimation:galAnimation1];

    id actionHide = [CCHide action];

    [galSprite1 runAction:[CCSequence
actions:galAction1,actionHide,nil]];
}

-(void) cutgalingale2
{

    CCSpriteSheet *galSheet1 = [CCSpriteSheet
spriteSheetWithFile:@"gal2.png"];
    [self addChild:galSheet1];

    // create the sprite
    CCSprite *galSprite1 = [CCSprite
spriteWithTexture:galSheet1.texture rect:CGRectMake(0,
502, 220, 251)];
    [galSheet1 addChild:galSprite1];

    // position the sprite in the center of the screen
    galSprite1.position = ccp(311,200);

    // create the animation
    CCAanimation *galAnimation1 = [CCAanimation
animationWithName:@"cutgal" delay:0.4f];

    int frameCount = 0;
    for (int y = 0; y <=2; y++) {
        CCSpriteFrame *frame1 = [CCSpriteFrame
frameWithTexture:galSheet1.texture
rect:CGRectMake(0,y*251,220,251) offset:ccp(0,0)];
        [galAnimation1 addFrame:frame1];

        frameCount++;

        if (frameCount == 2)
            break;
    }

    // create the action
    CCAanimate *galAction1 = [CCAanimate
actionWithAnimation:galAnimation1];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        id actionHide = [CCHide action];

        [galSprite1 runAction:[CCSequence
actions:galAction1,actionHide,nil]];
    }

-(void) cutgalingale3
{

    CCSpriteSheet *galSheet1 = [CCSpriteSheet
spriteSheetWithFile:@"gal3.png"];
    [self addChild:galSheet1];

    CCSprite *galSprite1 = [CCSprite
spriteWithTexture:galSheet1.texture rect:CGRectMake(0,
502, 220, 251)];
    [galSheet1 addChild:galSprite1];

    // position the sprite in the center of the screen
    galSprite1.position = ccp(311,200);

    // create the animation
    CCAAnimation *galAnimation1 = [CCAAnimation
animationWithName:@"cutgal" delay:0.4f];

    int frameCount = 0;
    for (int y = 0; y <=2; y++) {
        CCSpriteFrame *frame1 = [CCSpriteFrame
frameWithTexture:galSheet1.texture
rect:CGRectMake(0,y*251,220,251) offset:ccp(0,0)];
        [galAnimation1 addFrame:frame1];

        frameCount++;

        if (frameCount == 2)
            break;
    }

    // create the action
    CCAAnimate *galAction1 = [CCAAnimate
actionWithAnimation:galAnimation1];

    id actionHide = [CCHide action];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    [galSprite1 runAction:[CCSequence
actions:galAction1,actionHide,nil]];
}

-(void) cutgalingale4
{
    CCSpriteSheet *galSheet1 = [CCSpriteSheet
spriteSheetWithFile:@"gal4.png"];
    [self addChild:galSheet1];

    // create the sprite

    CCSprite *galSprite1 = [CCSprite
spriteWithTexture:galSheet1.texture rect:CGRectMake(0,
502, 220, 251)];
    [galSheet1 addChild:galSprite1];

    // position the sprite in the center of the screen
    galSprite1.position = ccp(311,200);

    // create the animation
    CCAnimation *galAnimation1 = [CCAnimation
animationWithName:@"cutgal" delay:0.4f];

    int frameCount = 0;
    for (int y = 0; y <=2; y++) {
        CCSpriteFrame *frame1 = [CCSpriteFrame
frameWithTexture:galSheet1.texture
rect:CGRectMake(0,y*251,220,251) offset:ccp(0,0)];
        [galAnimation1 addFrame:frame1];

        frameCount++;

        if (frameCount == 2)
            break;
    }

    // create the action
    CCAnimate *galAction1 = [CCAnimate
actionWithAnimation:galAnimation1];

    id actionHide = [CCHide action];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    [galSprite1 runAction:[CCSequence
actions:galAction1,actionHide,nil]];
}

-(void) cutgalingale5
{
    CCSpriteSheet *galSheet1 = [CCSpriteSheet
spriteSheetWithFile:@"gal5.png"];
    [self addChild:galSheet1];

    CCSprite *galSprite1 = [CCSprite
spriteWithTexture:galSheet1.texture rect:CGRectMake(0,
502, 220, 251)];
    [galSheet1 addChild:galSprite1];

    // position the sprite in the center of the screen
    galSprite1.position = ccp(311,200);

    // create the animation
    CCAAnimation *galAnimation1 = [CCAAnimation
animationWithName:@"cutgal" delay:0.4f];

    int frameCount = 0;
    for (int y = 0; y <=2; y++) {
        CCSpriteFrame *frame1 = [CCSpriteFrame
frameWithTexture:galSheet1.texture
rect:CGRectMake(0,y*251,220,251) offset:ccp(0,0)];
        [galAnimation1 addFrame:frame1];

        frameCount++;

        if (frameCount == 2)
            break;
    }

    // create the action
    CCAanimate *galAction1 = [CCAanimate
actionWithAnimation:galAnimation1];

    id actionHide = [CHide action];

    [galSprite1 runAction:[CCSequence
actions:galAction1,actionHide,nil]];
}

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

-(void) stepup_st3:(int)num {

    [self sendPacket:MUL_STAGE3 objectIndex:0 data:&num
length:sizeof(int) reliable:YES];
    if(num ==1)
    {
        bar.percentage +=20;
    }
    else if(num == 2)
    {
        bar.percentage +=20;
    }
    else if(num == 3)
    {
        bar.percentage +=20;
    }
    else if(num == 4)
    {
        bar.percentage +=20;
    }
    else if(num == 5)
    {
        bar.percentage +=20;
    }
    else if(num == 6)
    {
        bar.percentage +=20;
    }
    else if(num == 7)
    {
        bar.percentage +=20;
    }
    else if(num == 8)
    {
        bar.percentage +=20;
    }
    else if(num == 9)
    {
        bar.percentage +=20;
    }
    else if(num == 10)
    {
        bar.percentage +=20;
        [[SimpleAudioEngine
sharedEngine]stopBackgroundMusic];
        UIAlertView *alert = [[UIAlertView alloc]init];
        [alert setTitle:@"-à,³à,à,±à¹%à,™à,•à,à,™ 3
à¹Eà,ªà,£à¹#à,^à¹ à,¥à¹%à,$"];
        [alert setDelegate:self];
    }
}

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [alert addButtonWithTitle:@"à,•à, à,¥à,+"];
        [alert show];
        [alert release];
    }
}
-(void)stagemul4
{
    //Progress Timer
    timer.percentage = 0;

    //progressbar
    bar.percentage = 0;
    timer1 =0;

    //à¹€à, ,à¹%à,² stage4
    StageNumber=4;

    galingale.position = ccp(1311,1200);
    rightgal.position = ccp(1311,1200);
    knife.position =ccp(1409,1100);
    arrow.position = ccp(1400,1500);
    [background runAction:[CCHide action]];

    //init stage 4
    counthitupdown_st4 = 1; //for count hit up and down.
1 mean already hitup.2 mean already hitdown
    bodyHP = 3; //for count hit up and down. 1 mean
already hitup.2 mean already hitdown
    frameCountt = 0;
    Counttt = 0;
    smalll = 1;

    self.isTouchEnabled =YES;

    //BallIn mean Invisble Ball
    Ballsix = [CCSprite
spriteWithFile:@"background1_3.png"];
    Ballsix.position = ccp(240,160);
    [self addChild:Ballsix z:1];
    Ballone = [CCSprite
spriteWithFile:@"smallmush.png"];
    Ballone.position = ccp(540,200);
    [self addChild:Ballone z:3];
    Ballfive = [CCSprite
spriteWithFile:@"smallmushgrey.png"];
    Ballfive.position = ccp(50,25);
    [self addChild:Ballfive z:2];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    Ballthree = [CCSprite
spriteWithFile:@"smallmush.png"];
    Ballthree.position = ccp(540,200);
    [self addChild:Ballthree z:3];
    Ballfour = [CCSprite
spriteWithFile:@"smallmush.png"];
    Ballfour.position = ccp(540,200);
    [self addChild:Ballfour z:3];
    BallIn = [CCSprite
spriteWithFile:@"ghostbutton.png"];
    BallIn.position = ccp(300,140);
    [self addChild:BallIn z:3];
    CCSpriteSheet *galSheet1_st4 = [CCSpriteSheet
spriteSheetWithFile:@"mushroomm.png"];
    [self addChild:galSheet1_st4 z:2];

    Balltwo = [CCSprite
spriteWithTexture:galSheet1_st4.texture
rect:CGRectMake(0, 0, 225, 150)];
    Balltwo.position = ccp(540,200);

    [self sendPacket:MUL_STAGE4 objectIndex:0
data:&Counttt length:sizeof(int) reliable:YES];

    [self addChild:Balltwo z:2];
    [Balltwo runAction:[CCJumpBy actionWithDuration:.5
position:ccp(-240,-60) height:100 jumps:1]];

    [[SimpleAudioEngine
sharedEngine]playBackgroundMusic:@"playmusic3.mp3"];
}
-(void)stagemul5
{
    //Progress Timer
    timer.percentage = 0;

    //progressbar
    bar.percentage = 0;
    timer1 =0;

    //à¹€à, ,à¹%à,² stage5
    StageNumber=5;

    [Ballsix runAction:[CCHide action]];
    Ballsix.position = ccp(1240,1160);
    Ballone.position = ccp(1540,1200);

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

Ballfive.position = ccp(1500,1250);
Ballthree.position = ccp(1540,1200);
Ballfour.position = ccp(1540,1200);
BallIn.position = ccp(1300,1140);
//galSprite1.position = ccp(1311,1200);
//self

//CCSpriteSheet *galSheet1 = [CCSpriteSheet
spriteSheetWithFile:@"mushroomm.png"];
//[self addChild:galSheet1 z:2];

background2 = [CCSprite
spriteWithFile:@"background4_6.png"];
background2.position = ccp(240,160);
[self addChild:background2 z:-1];

//add lemon
lemon = [CCSprite spriteWithFile:@"lemon.png"];
lemon.position = ccp(311,160);
[self addChild:lemon];

[self sendPacket:MUL_STAGE5 objectIndex:0
data:&count_st5 length:sizeof(int) reliable:YES];

//draw waterlemon
[self DrawWaterLemon];

//add WaterLemon
waterLemon = [CCSprite
spriteWithFile:@"water_lemon.png"];
[self addChild:waterLemon z:1];
[self DrawWaterLemon];
//playAnimation
[self AnimatePod];

//check position
[self schedule:@selector(checkposition:)];

[[SimpleAudioEngine
sharedEngine]playBackgroundMusic:@"bat.mp3"];
}

-(void) DrawWaterLemon
{
    waterLemon.position = ccp(311,160);
    [waterLemon runAction:[CCHide action]];
}

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

}

-(void) AnimatePod
{
    CCSpriteSheet *potSheet1 = [CCSpriteSheet
spriteSheetWithFile:@"potter.png"];
    [self addChild:potSheet1];

    potSprite1 = [CCSprite
spriteWithTexture:potSheet1.texture rect:CGRectMake(0, 0,
250, 125)];
    [potSheet1 addChild:potSprite1];

    // position the sprite in the center of the screen
    potSprite1.position = ccp(311,30);

    // create the animation
    CCAnimation *potAnimation1 = [CCAnimation
animationWithName:@"boil" delay:0.2f];

    int frameCount= 0;
    for (int y = 0; y <2; y++) {
        CCSpriteFrame *frame1 = [CCSpriteFrame
frameWithTexture:potSheet1.texture
rect:CGRectMake(0,y*125,250,125) offset:ccp(0,0)];
        [potAnimation1 addFrame:frame1];

        frameCount++;

        if (frameCount == 2)
            break;
    }

    // create the action
    CCAnimate *potAction1 = [CCAnimate
actionWithAnimation:potAnimation1];
    CCRepeatForever *repeat = [CCRepeatForever
actionWithAction:potAction1];

    // run the action
    [potSprite1 runAction:repeat];
}

-(void) checkposition
{
    if (oldDistance<=200) {
        if (newDistance<oldDistance) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        if ((newDistance>=50)&(newDistance<=150))
    {
        check--;
        if (check==0) {
            [waterLemon runAction:[CCShow
action]]];
            [self MoveWaterLemon];
        }
    }
}

-(void) MoveWaterLemon
{
    [waterLemon runAction:[CCMoveTo actionWithDuration:1
position:ccp(311,70)]];

    if (count_st5==4) {
        [[SimpleAudioEngine
sharedEngine] stopBackgroundMusic];
        UIAlertView *alert = [[UIAlertView alloc] init];
        [alert setTitle:@"à,-à,³à,à,†à¹%à,™à,•à,à,™ 5
à¹€à,ªà,£à¹+à,^à¹ à,¥à¹%à,S"];
        [alert setDelegate:self];
        [alert addButtonWithTitle:@"à,•à, à,¥à,†"];
        [alert show];
        [alert release];
    }
}

-(void) checkposition:(ccTime) delta
{
    if (waterLemon.position.y==70) {
        count_st5++;
        [self sendPacket:MUL_STAGE5 objectIndex:0
data:&count_st5 length:sizeof(int) reliable:YES];
        bar.percentage +=20;
        CCLOG(@"%i",count_st5);
        [self DrawWaterLemon];
        [[SimpleAudioEngine
sharedEngine]playEffect:@"watersplash.mp3"];
    }
}

-(void) stagemul6
{
    //Progress Timer
    timer.percentage = 0;

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

//progressbar
bar.percentage = 0;
timer1 =0;
bar.position = ccp(3000,3000);

StageNumber=6;

lemon.position = ccp(1311,1160);

//*****init stage6
step = 0;
[self sendPacket:MUL_STAGE6 objectIndex:0 data:&step
length:sizeof(int) reliable:YES];

//load music
[[SimpleAudioEngine
sharedEngine]playBackgroundMusic:@"playmusic1.mp3"];

rectSheet1 = [CCSpriteSheet
spriteSheetWithFile:@"Allbar.png"];
[self addChild:rectSheet1];

Bar[0] = [CCSprite
spriteWithTexture:rectSheet1.texture rect:CGRectMake(0,
200, 300, 25)];
Bar[0].position = ccp(550,290);
//[Bar[0] setColor:ccc3(150,0,0)];
[self addChild:Bar[0]];

Bar[1] = [CCSprite
spriteWithTexture:rectSheet1.texture rect:CGRectMake(0,
150, 250, 25)];
Bar[1].position = ccp((Bar[0].position.x)+400,290);
//[Bar[1] setColor:ccc3(150,150,0)];
[self addChild:Bar[1]];

Bar[2] = [CCSprite
spriteWithTexture:rectSheet1.texture rect:CGRectMake(0,
150, 250, 25)];
Bar[2].position = ccp((Bar[1].position.x)+380,290);
//[Bar[2] setColor:ccc3(150,150,150)];
[self addChild:Bar[2]];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    Bar[3] = [CCSprite
spriteWithTexture:rectSheet1.texture rect:CGRectMake(0,
100, 200, 25)];
    Bar[3].position = ccp((Bar[2].position.x)+360,290);
    //[Bar[3] setColor:ccc3(255,0,0)];
    [self addChild:Bar[3]];

    Bar[4] = [CCSprite
spriteWithTexture:rectSheet1.texture rect:CGRectMake(0,
100, 200, 25)];
    Bar[4].position = ccp((Bar[3].position.x)+360,290);
    //[Bar[4] setColor:ccc3(255,255,0)];
    [self addChild:Bar[4]];

    Bar[5] = [CCSprite
spriteWithTexture:rectSheet1.texture rect:CGRectMake(0,
50, 150, 25)];
    Bar[5].position = ccp((Bar[4].position.x)+360,290);
    //[Bar[5] setColor:ccc3(255,255,255)];
    [self addChild:Bar[5]];

    CCSpriteSheet *potSheet1_st6 = [CCSpriteSheet
spriteSheetWithFile:@"boilsheet.png"];
    [self addChild:potSheet1_st6];

    potSprite1_st6 = [CCSprite
spriteWithTexture:potSheet1_st6.texture
rect:CGRectMake(0, 0, 336, 206)];
    [potSheet1_st6 addChild:potSprite1_st6];

    // position the sprite in the center of the screen
    potSprite1_st6.position = ccp(311,70);

    // create the animation
    CCAAnimation *potAnimation1 = [CCAAnimation
animationWithName:@"boil" delay:0.2f];

    int frameCount = 0;
    for (int y = 0; y <2; y++) {
        CCSpriteFrame *frame1 = [CCSpriteFrame
frameWithTexture:potSheet1_st6.texture
rect:CGRectMake(0,y*206,336,206) offset:ccp(0,0)];
        [potAnimation1 addFrame:frame1];

        frameCount++;

        if (frameCount == 2)

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        break;
    }

    // create the action
    CCAnimate *potAction1 = [CCAnimate
actionWithAnimation:potAnimation1];
    CCRepeatForever *repeat = [CCRepeatForever
actionWithAction:potAction1];

    CCSpriteSheet *potSheet2 = [CCSpriteSheet
spriteSheetWithFile:@"boilsheetred.png"];
    [self addChild:potSheet2];

    potSprite2 = [CCSprite
spriteWithTexture:potSheet2.texture rect:CGRectMake(0, 0,
336, 206)];
    [potSheet2 addChild:potSprite2];

    // position the sprite in the center of the screen
    potSprite2.position = ccp(311,70);

    // create the animation
    CCAnimation *potAnimation2 = [CCAnimation
animationWithName:@"boilred" delay:0.2f];

    int frameCount2 = 0;
    for (int y = 0; y < 2; y++) {
        CCSpriteFrame *frame2 = [CCSpriteFrame
frameWithTexture:potSheet2.texture
rect:CGRectMake(0,y*206,336,206) offset:ccp(0,0)];
        [potAnimation2 addFrame:frame2];

        frameCount2++;

        if (frameCount2 == 2)
            break;
    }

    // create the action
    CCAnimate *potAction2 = [CCAnimate
actionWithAnimation:potAnimation2];
    CCRepeatForever *repeat2 = [CCRepeatForever
actionWithAction:potAction2];
    [potSprite2 runAction:repeat2];
    [potSprite2 runAction:[CCHide action]];

    inputfood[0] = [CCSprite
spriteWithFile:@"small_gal.png"];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

inputfood[1] = [CCSprite
spriteWithFile:@"slice_lemongrass2.png"];
inputfood[2] = [CCSprite
spriteWithFile:@"slice_mushroom2.png"];
inputfood[3] = [CCSprite
spriteWithFile:@"shrimp1.png"];
inputfood[4] = [CCSprite
spriteWithFile:@"chilly3.png"];
inputfood[5] = [CCSprite
spriteWithFile:@"Kaffir_lime1.png"];

for (int i=0; i<6; i++) {
inputfood[i].position = ccp(500,500);
[self addChild:inputfood[i]];
[inputfood[i] runAction:[CCHide action]];
}

inputfood2 = [CCSprite
spriteWithFile:@"small_gal2.png"];
inputfood2.position =
ccp(potSprite1_st6.position.x,potSheet1_st6.position.y+20
0);
[self addChild:inputfood2];

[inputfood2 runAction:[CCHide action]];

for (int i=0; i<4; i++) {
slicelemongrass[i] = [CCSprite
spriteWithFile:@"slice_lemongrass.png"];
slicelemongrass[i].position =
ccp(potSprite1_st6.position.x +((i*23)-
47),potSheet1_st6.position.y+180+(i*5));
slicelemongrass[i].rotation = i*60;
[self addChild:slicelemongrass[i]];
[slicelemongrass[i] runAction:[CCHide action]];
}

for (int i=0; i<4; i++) {
//slicemushroom[i] = [CCSprite
spriteWithFile:@"slice_mushroom.png" rect:CGRectMake(0,
0, 22, 10)];
slicemushroom[i] = [CCSprite
spriteWithFile:@"slice_mushroom.png"];
slicemushroom[i].position =
ccp(potSprite1_st6.position.x +((i*23)-
47),potSheet1_st6.position.y+180+(i*5));
slicemushroom[i].rotation = i*61;
[self addChild:slicemushroom[i]];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [slicemushroom[i] runAction:[CCHide action]];
    }

    for (int i=0; i<3; i++) {
        shrimp[i] = [CCSprite
spriteWithFile:@"shrimp1.png"];
        shrimp[i].position =
ccp(potSprite1_st6.position.x +((i*23)-
47),potSheet1_st6.position.y+180+(i*5));
        shrimp[i].rotation = i*50;
        [self addChild:shrimp[i]];
        [shrimp[i] runAction:[CCHide action]];
    }

    for (int i=0; i<3; i++) {
        chilly[i] = [CCSprite
spriteWithFile:@"chilly2.png"];
        chilly[i].position =
ccp(potSprite1_st6.position.x +((i*23)-
47),potSheet1_st6.position.y+180+(i*5));
        chilly[i].rotation = i*50;
        [self addChild:chilly[i]];
        [chilly[i] runAction:[CCHide action]];
    }

    for (int i=0; i<3; i++) {
        KaffirLime[i] = [CCSprite
spriteWithFile:@"Kaffir_lime1.png"];
        KaffirLime[i].position =
ccp(potSprite1_st6.position.x +((i*23)-
47),potSheet1_st6.position.y+180+(i*5));
        KaffirLime[i].rotation = i*50;
        [self addChild:KaffirLime[i]];
        [KaffirLime[i] runAction:[CCHide action]];
    }

    // run the à,«à,;à¹%à, action
    [potSprite1_st6 runAction:repeat];

    //update
    [self schedule:@selector(updateRunningBar:)];

    //init array success
    for (int i=0; i<6; i++) {
        success[i] =NO;
    }

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

//Effectà¹,,à,ÿ
for (int i = 0; i<2; i++) {
    FireEffect[i] = [[CCParticleFire alloc] init];
    FireEffect[i].texture = [[CCTextureCache
sharedTextureCache] addImage:@"firen1.png"];

    FireEffect[0].position = ccp(250,-20);
    FireEffect[1].position = ccp(360,-20);

    //gravity
    FireEffect[i].gravity = CGPointMake(0,-40);

    //[FireEffect[i] setScaleX:2.0];

    //size in pixel
    FireEffect[i].startSize = 25.0f;
    FireEffect[i].startSizeVar =1.0f;
    FireEffect[i].endSize =
kParticleStartSizeEqualToEndSize;

    [self addChild:FireEffect[i]];
}

[[CCTouchDispatcher
sharedDispatcher]addTargetedDelegate:self priority:0
swallowsTouches:YES];
}
-(BOOL) ccTouchBegan:(UITouch *)touch withEvent:(UIEvent
*)event
{
    if (StageNumber==6) {
        return YES;
    }
}

-(void) ccTouchEnded:(UITouch *)touch withEvent:(UIEvent
*)event
{
    if (StageNumber==6)
    {
        CGPoint location = [touch locationInView:
[touch view]];
        location = [[CCDirector
sharedDirector]convertToGL:location];
        CGRect myRect[6];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        for (int i=0; i<6; i++) {
            myRect[i] =
CGRectMake(inputfood[i].position.x-
((inputfood[i].contentSize.width)/2),
inputfood[i].position.y-
((inputfood[i].contentSize.height)/2),
inputfood[i].position.x+((inputfood[i].contentSize.width)
/2),
inputfood[i].position.y+((inputfood[i].contentSize.height
)/2));
        }

        if(CGRectContainsPoint(myRect[0], location) &&
(step==0))
        {
            id actionFadeout = [CCFadeOut
actionWithDuration:0.5];
            [inputfood[0] runAction:actionFadeout];

            id actionfalldown = [CCSpawn actions:
                                [CCMoveTo
actionWithDuration:0.5 position:ccp(311,120)],
                                [CCRotateBy
actionWithDuration:0.5 angle: 180],
                                nil];

            id actionHide = [CHide action];
            [inputfood2 runAction:[CCSequence actions:
[CCShow action],
                                [CCDelayTime
actionWithDuration:0.3f],
                                actionfalldown,
                                actionHide,
                                nil]];

            id moveToActionDone = [CCCallFunc
actionWithTarget:self selector:@selector(watersplash)];
            id actiondelaytime = [CCDelayTime
actionWithDuration:0.8];
            [self runAction:[CCSequence
actions:actiondelaytime,moveToActionDone,nil]];

            //Bar[0]
            if(step==0)
            {

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        success[step] = YES;

        step=1;
        NSLog(@"Bar[0] %d",step);
        [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
        inputfood[0].position = ccp(500,500);
    }
}

else if(CGRectContainsPoint(myRect[1],
location) &&(step==1))
{
    //NSLog(@"Collision Detected");
    id actionFadeout = [CCFadeOut
actionWithDuration:0.5];
    [inputfood[1] runAction:actionFadeout];

    id actionfalldown0 = [CCSpawn actions:
                        [CCMoveTo
actionWithDuration:0.5
position:ccp(slicelemongrass[0].position.x+35,120)],
                        [CCRotateBy
actionWithDuration:0.5 angle: 620],
                        nil];

    id actionfalldown1 = [CCSpawn actions:
                        [CCMoveTo
actionWithDuration:0.5
position:ccp(slicelemongrass[1].position.x,120)],
                        [CCRotateBy
actionWithDuration:0.5 angle: 720],
                        nil];

    id actionfalldown2 = [CCSpawn actions:
                        [CCMoveTo
actionWithDuration:0.5
position:ccp(slicelemongrass[2].position.x+50,120)],
                        [CCRotateBy
actionWithDuration:0.5 angle: 180],
                        nil];

    id actionfalldown3 = [CCSpawn actions:
                        [CCMoveTo
actionWithDuration:0.5
position:ccp(slicelemongrass[3].position.x-30,120)],

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

                                                                    [CCRotateBy
actionWithDuration:0.5 angle: 50],
                                                                    nil];

        id actionHide = [CCHide action];

        [slicelemongrass[0] runAction:[CCSequence
actions: [CCShow action],

[CCDelayTime actionWithDuration:0.3f],

actionfalldown0,

actionHide,

nil]];

        [slicelemongrass[1] runAction:[CCSequence
actions: [CCShow action],

[CCDelayTime actionWithDuration:0.3f],

actionfalldown1,

actionHide,

nil]];

        [slicelemongrass[2] runAction:[CCSequence
actions: [CCShow action],

[CCDelayTime actionWithDuration:0.3f],

actionfalldown2,

actionHide,

nil]];

        [slicelemongrass[3] runAction:[CCSequence
actions: [CCShow action],

[CCDelayTime actionWithDuration:0.3f],

actionfalldown3,

actionHide,

nil]];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        id moveToActionDone = [CCCallFunc
actionWithTarget:self selector:@selector(watersplash)];
        id actiondelaytime = [CCDelayTime
actionWithDuration:0.8];
        [self runAction:[CCSequence
actions:actiondelaytime,moveToActionDone,nil]];

//Bar[1]
if(step==1)
{
    success[step] = YES;

    step=2;
    NSLog(@"step %d",step);
    [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
    inputfood[1].position = ccp(500,500);
}
}
else if(CGRectContainsPoint(myRect[2],
location) && (step==2))
{
    //NSLog(@"Collision Detected MUSHROOM");
    id actionFadeout = [CCFadeOut
actionWithDuration:0.5];
    [inputfood[2] runAction:actionFadeout];

    id actionfalldown0 = [CCSpawn actions:
        [CCMoveTo
actionWithDuration:0.5
position:ccp(slicelemongrass[0].position.x+35,120)],
        [CCRotateBy
actionWithDuration:0.5 angle: 620],
        nil];

    id actionfalldown1 = [CCSpawn actions:
        [CCMoveTo
actionWithDuration:0.5
position:ccp(slicelemongrass[1].position.x,120)],
        [CCRotateBy
actionWithDuration:0.5 angle: 720],
        nil];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        id actionfalldown2 = [CCSpawn actions:
                                [CCMoveTo
actionWithDuration:0.5
position:ccp(slicelemongrass[2].position.x+5,120)],
                                [CCRotateBy
actionWithDuration:0.5 angle: 180],
                                nil];
        id actionfalldown3 = [CCSpawn actions:
                                [CCMoveTo
actionWithDuration:0.5
position:ccp(slicelemongrass[3].position.x-30,120)],
                                [CCRotateBy
actionWithDuration:0.5 angle: 50],
                                nil];

        id actionHide = [CCHide action];
        [slicemushroom[0] runAction:[CCSequence
actions: [CCShow action],
[CCDelayTime actionWithDuration:0.3f],
actionfalldown0,
actionHide,
nil]];
        [slicemushroom[1] runAction:[CCSequence
actions: [CCShow action],
[CCDelayTime actionWithDuration:0.3f],
actionfalldown1,
actionHide,
nil]];
        [slicemushroom[2] runAction:[CCSequence
actions: [CCShow action],
[CCDelayTime actionWithDuration:0.3f],
actionfalldown2,
actionHide,
nil]];
        [slicemushroom[3] runAction:[CCSequence
actions: [CCShow action],
[CCDelayTime actionWithDuration:0.3f],

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

actionfalldown3,
actionHide,
nil]];

        id moveToActionDone = [CCCallFunc
actionWithTarget:self selector:@selector(watersplash)];
        id actiondelaytime = [CCDelayTime
actionWithDuration:0.8];
        [self runAction:[CCSequence
actions:actiondelaytime,moveToActionDone,nil]];

        if(step==2)
        {
            success[step] = YES;
            step=3;
            NSLog(@"success %d",step);
            [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
            inputfood[2].position = ccp(500,500);
        }
        else if(CGRectContainsPoint(myRect[3],
location) && (step==3))
        {
            //NSLog(@"Collsion Detected MUSHROOM");
            id actionFadeout = [CCFadeOut
actionWithDuration:0.5];
            [inputfood[3] runAction:actionFadeout];

            id actionfalldown0 = [CCSpawn actions:
                [CCMoveTo
actionWithDuration:0.5
position:ccp(shrimp[0].position.x+35,120)],
                [CCRotateBy
actionWithDuration:0.5 angle: 620],
                nil];

            id actionfalldown1 = [CCSpawn actions:
                [CCMoveTo
actionWithDuration:0.5
position:ccp(shrimp[1].position.x,120)],

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

                                                                    [CCRotateBy
actionWithDuration:0.5 angle: 50],
                                                                    nil];

        id actionfalldown2 = [CCSpawn actions:
                                                                    [CCMoveTo
actionWithDuration:0.5
position:ccp(shrimp[2].position.x+5,120)],
                                                                    [CCRotateBy
actionWithDuration:0.5 angle: -180],
                                                                    nil];

        id actionHide = [CCHide action];

        [shrimp[0] runAction:[CCSequence actions:
[CCShow action],
                                                                    [CCDelayTime
actionWithDuration:0.3f],
actionfalldown0,
                                                                    actionHide,
                                                                    nil]];
        [shrimp[1] runAction:[CCSequence actions:
[CCShow action],
                                                                    [CCDelayTime
actionWithDuration:0.3f],
actionfalldown1,
                                                                    actionHide,
                                                                    nil]];
        [shrimp[2] runAction:[CCSequence actions:
[CCShow action],
                                                                    [CCDelayTime
actionWithDuration:0.3f],
actionfalldown2,
                                                                    actionHide,
                                                                    nil]];

        id moveToActionDone = [CCCallFunc
actionWithTarget:self selector:@selector(watersplash)];
        id actiondelaytime = [CCDelayTime
actionWithDuration:0.8];
        [self runAction:[CCSequence
actions:actiondelaytime,moveToActionDone,nil]];

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        if (step==3)
        {
            success[step] = YES;

            step=4;
            NSLog(@"success %d",step);
            [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
            inputfood[3].position = ccp(500,500);
        }

    }
    else if(CGRectContainsPoint(myRect[4],
location) && (step==4))
    {
        //NSLog(@"Collsion Detected CHILLY");
        id actionFadeout = [CCFadeOut
actionWithDuration:0.5];
        [inputfood[4] runAction:actionFadeout];

        id actionfalldown0 = [CCSpawn actions:
                             [CCMoveTo
actionWithDuration:0.5
position:ccp(chilly[0].position.x+35,130)],
                             [CCRotateBy
actionWithDuration:0.5 angle: 620],
                             nil];

        id actionfalldown1 = [CCSpawn actions:
                             [CCMoveTo
actionWithDuration:0.5
position:ccp(chilly[1].position.x,130)],
                             [CCRotateBy
actionWithDuration:0.5 angle: 50],
                             nil];

        id actionfalldown2 = [CCSpawn actions:
                             [CCMoveTo
actionWithDuration:0.5
position:ccp(chilly[2].position.x+5,130)],
                             [CCRotateBy
actionWithDuration:0.5 angle: -180],
                             nil];

        id actionHide = [CCHide action];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [chilly[0] runAction:[CCSequence actions:
[CCShow action],
                                [CCDelayTime
actionWithDuration:0.3f],
                                actionfalldown0,
                                actionHide,
                                nil]];
        [chilly[1] runAction:[CCSequence actions:
[CCShow action],
                                [CCDelayTime
actionWithDuration:0.3f],
                                actionfalldown1,
                                actionHide,
                                nil]];
        [chilly[2] runAction:[CCSequence actions:
[CCShow action],
                                [CCDelayTime
actionWithDuration:0.3f],
                                actionfalldown2,
                                actionHide,
                                nil]];
        [potSprite2 runAction:[CCSequence
actions:[CCDelayTime actionWithDuration:0.8f],
                                [CCShow
action],
                                nil]];
        [potSprite1_st6 runAction:[CCSequence
actions:[CCDelayTime actionWithDuration:0.8f],
                                [CCHide
action],
                                nil]];

        id moveToActionDone = [CCCallFunc
actionWithTarget:self selector:@selector(redsplash)];
        id actiondelaytime = [CCDelayTime
actionWithDuration:0.8];
        [self runAction:[CCSequence
actions:actiondelaytime,moveToActionDone,nil]];

        if(step==4)
        {

                success[step] = YES;

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        step=5;
        NSLog(@"success %d",step);
        [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
        inputfood[4].position = ccp(500,500);
    }
}

else if(CGRectContainsPoint(myRect[5],
location) && (step==5))
{
    NSLog(@"Collsion Detected Kaffir Lime");
    id actionFadeout = [CCFadeOut
actionWithDuration:0.5];
    [inputfood[5] runAction:actionFadeout];

    id actionfalldown0 = [CCSpawn actions:
[CCMoveTo
actionWithDuration:0.5
position:ccp(chilly[0].position.x+35,120)],
[CCRotateBy
actionWithDuration:0.5 angle: 620],
nil];

    id actionfalldown1 = [CCSpawn actions:
[CCMoveTo
actionWithDuration:0.5
position:ccp(chilly[1].position.x,120)],
[CCRotateBy
actionWithDuration:0.5 angle: 50],
nil];

    id actionfalldown2 = [CCSpawn actions:
[CCMoveTo
actionWithDuration:0.5
position:ccp(chilly[2].position.x+5,120)],
[CCRotateBy
actionWithDuration:0.5 angle: -180],
nil];

    id actionHide = [CCHide action];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [KaffirLime[0] runAction:[CCSequence
actions: [CCShow action],

[CCDelayTime actionWithDuration:0.3f],

actionfalldown0,

actionHide,

nil]];

        [KaffirLime[1] runAction:[CCSequence
actions: [CCShow action],

[CCDelayTime actionWithDuration:0.3f],

actionfalldown1,

actionHide,

nil]];

        [KaffirLime[2] runAction:[CCSequence
actions: [CCShow action],

[CCDelayTime actionWithDuration:0.3f],

actionfalldown2,

actionHide,

nil]];

        id moveToActionDone = [CCCallFunc
actionWithTarget:self selector:@selector(redsplash)];
        id actiondelaytime = [CCDelayTime
actionWithDuration:0.8];
        [self runAction:[CCSequence
actions:actiondelaytime,moveToActionDone,nil]];

        if(step==5)
        {

                success[step] = YES;

                step=6;
                NSLog(@"success %d",step);
                [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];

                inputfood[5].position = ccp(500,500);

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

        [self resultscore];
    }

    }

}

-(void)updateRunningBar:(ccTime) dt
{
    for(int i=0;i<6;i++)
    {
        Bar[i].position = ccp(Bar[i].position.x -
100*dt, Bar[i].position.y);

        if((Bar[i].position.x -
((Bar[i].contentSize.width)/2)) <=arrow1.position.x)
        {
            if(success[i]==NO)
                [Bar[i] setColor:ccc3(0,255,0)];
            else
                [Bar[i] setColor:ccc3(255,150,0)];

            if((i==0)&& (step==i))
            {
                [inputfood[0] runAction:[CCShow
action]];
                inputfood[0].position = ccp(65,100);
            }

            else if((i==1) && (step==i))
            {
                NSLog(@"change pic 0 to 1");

                [inputfood[1] runAction:[CCShow
action]];
                inputfood[1].position = ccp(65,100);
            }

            else if(i==2)
            {
                if(step==i)
                {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

                                NSLog(@"Bar 2 running
:step=%d",step);
                                [inputfood[2] runAction:[CCShow
action]];
                                inputfood[2].position =
ccp(65,100);
                                }
                                }
                                else if((i==3)&&(step==i))
                                {
                                [inputfood[3] runAction:[CCShow
action]];
                                inputfood[3].position = ccp(65,100);
                                }
                                else if((i==4)&&(step==i))
                                {
                                [inputfood[4] runAction:[CCShow
action]];
                                inputfood[4].position = ccp(65,100);
                                }
                                else if((i==5)&&(step==i))
                                {
                                [inputfood[5] runAction:[CCShow
action]];
                                inputfood[5].position = ccp(65,100);
                                }
                                }
                                if((Bar[i].position.x +
((Bar[i].contentSize.width)/2)) <=arrow1.position.x)
                                {
                                if(success[i]==NO)
                                {
                                [Bar[i] setColor:ccc3(255,0,0)];
                                if((i==0) &&
(step==0))
                                {
                                inputfood[0].position =
ccp(500,500);
                                step=1;
                                //NSLog(@"fail: %d",step);
                                [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    }
    else if((i==1) &&
(step==1))
    {
        inputfood[1].position =
ccp(500,500);
        step=2;
        NSLog(@"fail: step %d",step);
        [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
    }
    else if((i==2) &&
(step==2))
    {
        inputfood[2].position =
ccp(500,500);
        step=3;
        NSLog(@"fail: step %d",step);
        [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
    }
    else if((i==3) && (step==3))
    {
        inputfood[3].position =
ccp(500,500);
        step=4;
        NSLog(@"fail: step %d",step);
        [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
    }
    else if((i==4) && (step==4))
    {
        inputfood[4].position =
ccp(500,500);
        step=5;
        NSLog(@"fail: step %d",step);
        [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    }

    else if((i==5) && (step==5))
    {
        inputfood[5].position =
ccp(500,500);
        step=6;
        NSLog(@"fail: step %d",step);
        [self sendPacket:MUL_STAGE6
objectIndex:0 data:&step length:sizeof(int)
reliable:YES];

        [self resultscore];
    }
}
}
//NSLog(@"%d",step);
}
-(void) watersplash
{
    splasheffect2 = [[CCParticleFireworks alloc] init];
    splasheffect2.texture = [[CCTextureCache
sharedTextureCache] addImage:@"water4.png"];
    splasheffect2.position = ccp(300,100);
    //duration
    splasheffect2.duration =0.05;
    //gravity
    splasheffect2.gravity = CGPointMake(0,-270);
    splasheffect2.life =1;
    splasheffect2.lifeVar =4;
    //color
    ccColor4F startColor ={1.0f, 1.0f, 1.0f, 1.0f};
    splasheffect2.startColor = startColor;
    ccColor4F startColorVar = {0.0f, 0.1f, 0.5f, 1.0f};
    splasheffect2.startColorVar = startColorVar;
    ccColor4F endColor = {1.0f, 1.0f, 1.0f, 1.0f};
    splasheffect2.endColor = endColor;
    ccColor4F endColorVar = {0.0f, 0.1f, 0.1f, 1.0f};
    splasheffect2.endColorVar = endColorVar;

    //size in pixel
    splasheffect2.startSize = 25.0f;
    splasheffect2.startSizeVar =1.0f;
    splasheffect2.endSize =
kParticleStartSizeEqualToEndSize;

    [self addChild:plasheffect2];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    [[SimpleAudioEngine
sharedEngine]playEffect:@"watersplash.mp3"];
}
-(void) redsplash
{
    splasheffect3 = [[CCParticleFireworks alloc] init];
    splasheffect3.texture = [[CCTextureCache
sharedTextureCache] addImage:@"water4.png"];
    splasheffect3.position = ccp(300,100);
    //duration
    splasheffect3.duration =0.05;
    //gravity
    splasheffect3.gravity = CGPointMake(0,-270);
    splasheffect3.life =1;
    splasheffect3.lifeVar =4;
    //color
    ccColor4F startColor ={1.0f, 0.45f, 0.36f, 1.0f};
    splasheffect3.startColor = startColor;
    ccColor4F startColorVar = {0.5f, 0.0f, 0.0f, 1.0f};
    splasheffect3.startColorVar = startColorVar;
    ccColor4F endColor = {1.0f, 0.81f, 0.78f, 1.0f};
    splasheffect3.endColor = endColor;
    ccColor4F endColorVar = {0.1f, 0.01f, 0.01f, 1.0f};
    splasheffect3.endColorVar = endColorVar;

    //size in pixel
    splasheffect3.startSize = 25.0f;
    splasheffect3.startSizeVar =1.0f;
    splasheffect3.endSize =
kParticleStartSizeEqualToEndSize;

    //id actiondelaytime = [CCDelayTime
actionWithDuration:0.5];
    //id ActionDone = [CCCallFunc actionWithTarget:self
selector:@selector(destroy)];
    //[[splasheffect2 runAction:[CCSequence
actions:actiondelaytime,ActionDone,nil]];

    [self addChild:splasheffect3];

    [[SimpleAudioEngine
sharedEngine]playEffect:@"watersplash.mp3"];
}
-(void) resultscore
{
    int result=0;

```

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

for (int i=0; i<6; i++) {
    if(success[i]==NO){
        //NSLog(@"stage %d : fail",i);
    }
    else if(success[i]==YES)
    {
        //NSLog(@"stage %d : success",i);
        result++;
    }
}
//NSLog(@"result : %d",result);
[self sendPacket:MUL_STAGE_FIN_SCORES objectIndex:0
data:&result length:sizeof(int) reliable:YES];
[self sendPacket:MUL_STAGE_FIN_RESULT objectIndex:0
data:&isWin length:sizeof(int) reliable:YES];

[[SimpleAudioEngine
sharedEngine]stopBackgroundMusic];
UIAlertView *alert = [[UIAlertView alloc]init];
[alert setTitle:@"-à,³à,,à,±à¹%à,™à,•à,à,™ 6
à¹Èà,³à,£à¹+à,^à¹ à,¥à¹%à,S"];
[alert setDelegate:self];
[alert addButtonWithTitle:@"à,•à,à,¥à,±"];
[alert show];
[alert release];
}
-(void)resultstage
{

//Progress Timer
timer.percentage = 0;

//progressbar
bar.percentage = 0;
timer1 =0;

barstage.position =ccp(1000,1000);
whitecircle.position =ccp(1000,1000);

StageNumber =7;
timer.position = ccp(1160,1240);

[background2 runAction:[CCHide action]];
potSprite1_st6.position = ccp(1311,1170);

NSLog(@">>> Start hiding");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

// position the sprite in the center of the screen
potSprite2.position = ccp(1311,1170);
[potSprite2 runAction:[CCHide action]];

//inputfood[0] = [CCSprite
spriteWithFile:@"small_gal.png"];
//inputfood[1] = [CCSprite
spriteWithFile:@"slice_lemongrass2.png"];
//inputfood[2] = [CCSprite
spriteWithFile:@"slice_mushroom2.png"];
//inputfood[3] = [CCSprite
spriteWithFile:@"shrimpl.png"];
//inputfood[4] = [CCSprite
spriteWithFile:@"chilly3.png"];
//inputfood[5] = [CCSprite
spriteWithFile:@"Kaffir_lime1.png"];

for (int i=0; i<6; i++) {
    inputfood[i].position = ccp(1500,1500);
}

inputfood2.position = ccp(1500,1500);
for (int i=0; i<4; i++) {
    slicelemongrass[i].position = ccp(1000,1000);
}

for (int i=0; i<4; i++) {
    slicemushroom[i].position = ccp(1000,1000);
}

for (int i=0; i<3; i++) {
    shrimp[i].position = ccp(1000,1000);
}

for (int i=0; i<3; i++) {
    chilly[i].position = ccp(1000,1000);
}

for (int i=0; i<3; i++) {
    KaffirLime[i].position = ccp(1000,1000);
}
//Effectà¹,,à,ÿ
for (int i = 0; i<2; i++) {
    FireEffect[0].position = ccp(1250,1020);
    FireEffect[1].position = ccp(1360,1020);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม ก.28 ไฟล์ Multiplayer.m (ต่อ)

```

    }
    NSLog(@">>> End hiding");

    NSLog(@">>> Start new init");

    //init resultstage
    BGresult = [CCSprite
spriteWithFile:@"backgrounddd.png"];
    BGresult.position = ccp(240,160);
    [self addChild:BGresult];

    //isWin = RESULT_WIN;

    if (isWin == RESULT_WIN) {
        resultchar= @"à,šà,™à,°";
    }
    else if(isWin == RESULT_LOSE)
    {
        resultchar=@"à¹ à,žà¹%";
    }
    GameResultLabel = [CCLabel
labelWithString:resultchar fontName:@"Marker Felt"
fontSize:64];
    GameResultLabel.position = ccp(150,200);
    [self addChild:GameResultLabel];

    backbutton2 = [CCSprite
spriteWithFile:@"back_button.png"];
    backbutton2.position = ccp(430,290);
    [self addChild:backbutton2 z:3];

    NSLog(@">>> End new init");
}

-(void) fadeinandout:(CCSprite *)obj {

    id fadeout1 = [CCFadeOut actionWithDuration:0.1];
    id fadein1 = [CCFadeIn actionWithDuration:0.1];
    [obj runAction:[CCSequence actions:[CCDelayTime
actionWithDuration:0.7f],[CCShow action],fadeout1,
fadein1,fadeout1, fadein1,fadeout1, fadein1,fadeout1,
nil]];
}
@end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้