

Introduction to Mobile Application

Suphot Sawattiwong
tohpus@hotmail.com

การพัฒนาโปรแกรมบน โทรศัพท์เคลื่อนที่

- สิ่งที่ต้องคำนึงถึงในการพัฒนาโปรแกรมบนโทรศัพท์เคลื่อนที่ มีจุดใหญ่ๆ 2 จุดคือ
 - Platform ของโทรศัพท์เคลื่อนที่
 - Specification ของโทรศัพท์เคลื่อนที่

Platform

Java Base	C Base
J2ME	Symbian
Doja	Brew
Android	.Net Compact Framework
	iPhone OS

J2ME

- J2ME (Java 2 Micro Edition) เป็นการย่อส่วนมาจาก J2SE (Java 2 Standard Edition) โดยเน้นไปยังอุปกรณ์พกพา ไม่ว่าจะเป็น PDA, โทรศัพท์เคลื่อนที่ และอื่นๆ

CLDC

- CLDC (Connected Limited Device Configuration) เป็น ส่วนของการตั้งค่า J2ME ที่ออกแบบสำหรับอุปกรณ์ ที่มีความเร็ว Processor ที่ต่ำ และ มีหน่วยความจำที่จำกัด

MIDP

- MIDP (Mobile Information Device Profile)
เป็นโปรไฟล์ (API) กำหนดคุณสมบัติต่างๆ เช่น
 - High level user interface elements
 - Application lifecycle management
 - Local data storage
 - Connectivity
 - Low level Graphics APIs
 - Input handling
 - Media (Sound)

MIDP

- MIDP profile จะทำส่วนกับ CLDC configuration เพื่อสร้างระบบ KVM (Kilo Virtual Machine) ซึ่งเป็นส่วน Run Time ที่เรา
จะใช้พัฒนา J2ME

Specification ของโทรศัพท์เคลื่อนที่

- การพัฒนามาบนโทรศัพท์เคลื่อนที่จำเป็นต้องรู้เกี่ยวกับ ความสามารถและข้อมูลทางเทคนิคของโทรศัพท์เคลื่อนที่ เช่น
 - ความเร็วของ CPU
 - ความเร็วการวาดใน 1 หน้าจอ
 - การสนับสนุนไฟล์ Format ต่างๆ
 - เป็นต้น

ปัญหาในการพัฒนาโปรแกรมในโทรศัพท์ เคลื่อนที่

- ปัญหาที่เกิดจากข้อจำกัดของ Hardware
- ปัญหาที่เกิดจากความหลากหลายของ Hardware
- ปัญหาที่เกิดจากการสนับสนุนเทคโนโลยีไม่ตรงกัน
- ปัญหาที่เกิดจากตัวโทรศัพท์เคลื่อนที่เอง

ปัญหาที่เกิดจากข้อจำกัดของ Hardware

- จะเป็นปัญหาที่เกี่ยวข้องกับ Hardware มีข้อจำกัด เช่น
 - Memory ที่มีจำกัด
 - Data Storage มีจำกัด
 - เป็นต้น

ปัญหาที่เกิดจากความหลากหลายของ Hardware

- ปัญหาเหล่านี้หลีกเลี่ยงไม่ได้มากนัก หากต้องสนับสนุนโทรศัพท์เคลื่อนที่ที่หลากหลาย ยี่ห้อ เช่น
- ปัญหาที่เกิดจากโทรศัพท์เคลื่อนที่ที่มีความสามารถที่ไม่เท่ากัน มาตรฐานของ MIDP และ CLDC ที่แตกต่างกัน เป็นต้น

ปัญหาที่เกิดจากการสนับสนุนเทคโนโลยีไม่ตรงกัน

- การสนับสนุน Platform ไม่ตรงกัน เช่น บางรุ่นสนับสนุน J2ME ของ Nokia, J2ME ของ Sony Ericsson
- iPhone สนับสนุนระบบ Multi-Touch และรันบน iPhone OS
- Google G1 สนับสนุน Android
- iPhone 3GS สนับสนุน OpenGL ES2.0 แต่ iPhone และ iPod Touch สนับสนุน OpenGL ES1.1 เป็นต้น

ปัญหาที่เกิดจากตัวโทรศัพท์เคลื่อนที่เอง

- ปัญหาเหล่านี้เกิดจาก OS ของโทรศัพท์เคลื่อนที่ หรือในชุดพัฒนา มีปัญหา หรือมี Bug เกิดขึ้น รวมถึงปัญหาของ Hardware รุ่นนั้นด้วยเช่นกัน

iPhone

- iPhone และ iPod Touch เป็น Platform ที่นำเสนอนวัตกรรมใหม่ๆ มากมาย ไม่ว่าจะเป็นระบบ Touch/Multi-Touch, Accelerometer, Core-Location เป็นต้น

สิ่งที่ต้องเตรียม

- เครื่อง Mac หรือ เครื่องมีระบบปฏิบัติการ Leopard หรือ Snow Leopard
- iPhone SDK (developer.apple.com/iphone)
- iPhone หรือ iPod Touch
- internet

iPhone SDK

- iPhone SDK ที่เลือกใช้ในการเรียนการสอน
ครั้งนี้ ให้ใช้ iPhone SDK 3.1.2

ภาษาที่ใช้ในการพัฒนา

- ในการพัฒนา Software บน iPhone นั้น สามารถทำได้ โดยการใช้ Objective C เป็นตัวพัฒนา และอนุญาตให้ใช้ C+ ได้ด้วยเช่นกัน
- แต่ส่วนใหญ่นิยมใช้ Objective C ในการพัฒนา

iPhone Specification

- iPhone สนับสนุนการวาด Graphic ด้วย OpenGL ES และ Quartz2D
- iPhone มี Resolution ในแนวตั้งหรือ Portrait 320x480 และในแนวนอนหรือ Landscape
- สามารถเชื่อมต่อข้อมูลผ่าน Internet โดยผ่าน wifi หรือ GPRS หรือ 3G

เครื่องมือในการพัฒนา

- iPhone SDK 3.1.2
- XCODE
- Instrument
- Dashcode
- iPhone Simulator
- Interface Builder

ข้อจำกัดของ Platform

- ข้อจำกัดด้านพื้นที่เก็บข้อมูล
- ข้อจำกัดด้านการเข้าถึงข้อมูล
- ข้อจำกัดด้านหน่วยความจำ
- ข้อจำกัดเงื่อนไขการ Download iPhone Application ผ่าน Internet

ข้อจำกัดของ Platform

- ข้อจำกัดด้านวิธีการป้อนข้อมูล
- ข้อจำกัดด้านพลังงาน
- ข้อจำกัดของโปรแกรม
- ข้อจำกัดด้านพฤติกรรมของผู้ใช้งาน

Sandbox

- iPhone OS จะมีการจำกัดสิทธิ์ของโปรแกรม
ทุกตัวที่พัฒนาด้วย iPhone SDK ให้ทำงาน
แบบ Sandbox เพื่อความปลอดภัยของระบบ
Sandbox ของ iPhone จะจำกัดไม่ให้
โปรแกรมเข้าถึงระบบได้โดยตรง เข้าถึงได้
แค่บางส่วนที่ตัว iPhone OS อนุญาต

ข้อจำกัดของ SDK

- ไม่มี Garbage Collector
- Library มี Bug และยังไม่สมบูรณ์
- Framework ที่มีใน iPhone OS นั้น ไม่ update เร็วเท่าใน iPhone SDK

Model-View-Controller

- MVC เป็นเฟรมเวิร์คที่เกิดขึ้นมาในสมัย Smalltalk-80 เมื่อกว่ายี่สิบปีมาแล้ว โดยแยก ออบเจกต์ที่เก็บข้อมูล (model) ออบเจกต์ที่ แสดงข้อมูล (view) และออบเจกต์ที่ติดต่อกับผู้ใช้ (controller) ออกจากกันอย่างชัดเจน