Emerging Technologies

About Me

- My name is Barry Denby
- Will be your lecturer for this module
- My email address is barry.denby@griffith.ie
- Be warned I like my bad puns :)

What is this module about?

- From the module descriptor we find this text
  - “This module aims to give the learner the opportunity to study relatively new developments in Computer Science. The currently suggested topic is Mobile Development”
- Allows us to update the module quickly should a new topic come up
- For now it is focused on Android development

What is this module about?

- There are multiple reasons why I'm sticking with Android development
  - First of all it is the most popular platform by far according to IDC data Q3 of 2016 saw android with an 86.8% share of all devices shipped that quarter
  - http://www.idc.com/promo/smartphone-market-share/os
- My guess from the stats is that the majority of you will have an android device of some description

What is this module about?

- The tools are freely available for all platforms
  - Doesn't matter if you are Windows/Linux/OSX you can get access to the tools easily
  - Apple has a walled garden approach
    - You can only develop on iOS natively if you are in possession of a Mac and have Xcode installed

How will this course be assessed?

- Assessment will consist of 50% CA and a 50% exam
- You will be required to pass both components to pass the course
  - If you fall short in one component by 1 or 2% it is possible to pass by compensation
  - However, fall any further and you will have to repeat the component that you have failed

Continuious Assessment

- Your continious assessment will consist of three programming assignments
  - Assignment 1 will be worth 28%
  - Assignment 2 will be worth 32%
  - Assignment 3 will be worth 40%

Texts to work from

- After this in week 6 I will start to release assignments
  - A01 given in week 6 due in week 7
  - A02 given in week 8 due in week 9
  - A03 given in week 10 in week 12
- The deadline will be the Sunday of the respective week at 23:55

Continuious Assessment

- All CA will be programming based
  - Will consist of a lot of code
  - I will provide you with a program spec
  - You will be required to produce the full code
    - No shell code provided
  - Also required to provide documentation
- You will be challenged with CA
  - Examples will show a task in simple detail
  - But up to you to figure how to combine to make a useful application

Continuious Assessment

- What I will assume about you
  - You know and are familiar with OO development and relevant data structures
    - And are capable of designing and building relatively large programs from scratch
  - You know and are familiar with Relational Databases
  - You know and are familiar with HCI & GUI programming

Lab Sessions

- You are required to attend one lab session a week
  - Monday 12:30 JS101: Griffith + BiB students
  - Monday 14:00 JS001: EPITA
  - Tuesday 16:00 W102: EPITECH

Exam

- The exam is worth the other 50% of your assessment
  - It will be a 3 hour long exam
  - 6 questions do any 5
  - Mostly theory with some programming in there (will be short snippets)

Tools that you will require

- As was stated before Android enables you to develop on either Windows/OSX/Linux
  - You will be able to get the tools that are mentioned here for all three platforms
- There will be no labs running this week
  - Use this first week to get your toolchain up and running

Tools that you will require

- All programming will be Java based
  - Everyone in the room should be fluent in Java by now.
  - Will enable you to get into development very quickly
  - There will be XML as well but as you will likely have dealt with HTML at this stage XML should look familiar and friendly.

Tools that you will require

- You only have one choice of IDE and that is Android Studio
  - Google's official IDE for android development
  - Based on IntelliJ
    - If you use IntelliJ already android studio will look familiar to you
  - Is available on Windows/Linux/OSX
    - Unlike other mobile OS IDEs

Android Studio

- Be aware of the following with Android Studio
  - If you are running Windows you'll need at least 4GB of RAM
    - Can get away with less on Linux/OSX
  - The gradle build system is pretty slow
    - Generally avoid background processes or having 50 windows open at once

Restrictions on App Development

- I will impose a few restrictions and Application development
  - All applications must have a minimum API level of API 15 (Android 4.0 Ice Cream Sandwich)
  - The Android By Example programming text you will be using has all examples written to work with this API level in mind as a minimum
  - Examples work with Android 6.0 (Marshmallow) as well

Running Applications

- When you have developed an application you will have one of three options to run them
  - Option 1: use your own android device (provided it is Android 4.0 or higher)
  - Option 2: use Genymotion (x86 virtual machine implementation of android)
  - Option 3: use the provided android emulator with the SDK (not recommended)

Option 1: using your own device

- Easiest and fastest option
- Need to enable developer options
  - Pre 4.1 devices have the option available in Developer settings under the settings menu
  - Post 4.1 devices have the option hidden by default

Option 1: using your own device

- On Post 4.1 devices do the following
  - Go to Settings >About Phone and hit the build number about 7 or 8 times to enable developer mode
- In the developer options enable USB debugging and switch developer options to on
  - Linux and OSX should be plug and play when you attach your device

Option 1: using your own device

- Windows requires the Google USB driver (find it in the SDK manager)
  - When you connect your device open up the device manager
  - In the unknown devices you should see an android device
  - Update the driver and point it to <android-sdk>/extras/google/usb-driver/ and select the .inf file there

Option 1: using your own device

- It will present you with one of three options
  - Pick the ADB composite interface
- After the driver has been setup on your OS of choice a dialog will appear asking to accept a key from your machine
  - Accept it and you can start developing

Option 2: Using GenyMotion

- If you don't have an android device I would suggest using this option instead
  - Uses android-x86 in a virtual machine that is configured to link up with eclipse and android studio
  - Get it from https://www.genymotion.com/
  - Note you will need to register on their site to get access to the images. (it's free)

Option 2: Using GenyMotion

- Once it is installed and you have downloaded and started an image
  - Android Studio will see it when you go to start and application
  - A little bit slower than native speed
  - But that's purely down to virtualisation

Option 3: Android Emulator

- Absolute last resort solution
  - Emulates an ARM/X86/MIPS chip thus is really slow
  - However, it is the official emulator so it is guarenteed to work
  - If you can avoid this option then avoid it.

How will coding be done in this course?

- Everything will be done by hand and without the use of the provided GUI builder tools
  - Gives you a much better understanding of how android fits together as a whole from a programming perspective
  - Also gives you access to the full functionality
  - Android apps are composed of many files and thus this will show you how they will fit together.

Texts to work from

- I have a series of android example programs written in text on the moodle
  - About 180 pages long, gives you a small insight into android development.
  - I will give you specific examples to look up for each of the individual assignments before you get them.
  - They will explain and show the concepts you need to get them completed.
  - Some of these are a little outdated but I know the changes that need to be made, so please ask

Texts to work from

- In the first few labs I would suggest trying out the programs there
  - To give you an idea of how android fits together
  - And also familiarise yourself with your tools
  - To prepare for the assignments ahead
  - Do not copy paste code: **YOU WILL LEARN NOTHING THIS WAY**

Texts to work from

- If you want a further follow up to android I would suggest the following book
  - Professional Android 4 Application Development
  - By Reto Meier
  - Well out of date now but still a worthwhile read and contains lots of detail.

How lectures will work

- Will get you to do some basic research on a topic for 10 to 15 minutes
  - I will ask you what you come up with
  - Will be directly related to the lecture
  - So be here on time

Plan over the coming weeks

- I intend to cover the following topics
  - Week 2: Differences between mobile and desktop
  - Week 3: Android Application Structure
  - Week 4: Custom Views and Multitouch

Plan over the coming weeks

- Week 5: Activity life cycle and task management
- Week 6: Dalvik
- Week 7: Android RunTime
- Week 8: Android from a security perspective

Plan over the coming weeks

- Week 9: Sensors
- Week 10: The need for multi threading
- Week 11: changes in android 6.0
- Week 12: discussion about the exam