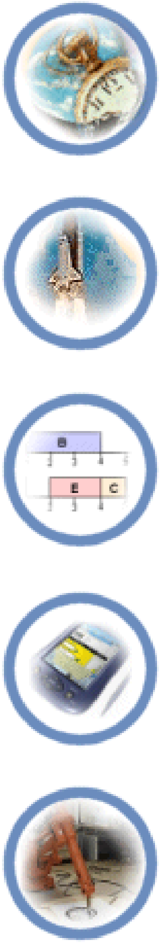


Lecture ”3”
Mobile App Development
(Android, iOS, BlackBerry, Windows Mobile)

<lecturer, date>

Outline

- **Smartphones**
- Developing Mobile Applications
- Android
- iOS
- BlackBerry
- Windows Mobile
- References

Cell phones

- low-end ”feature phones”

- high-end ”smartphones”

Smartphone

- Features
  - More powerful than feature phone
  - With more device capabilities
- Smartphone as a new PC
  - Typically cheaper than computers
  - More convenient because of their portability
  - Will perform many of the routine tasks currently on a desktop/laptop
  - The center of gravity of the software industry will be mobilized

Outline

- Smartphones
- **Developing Mobile Applications**
- Android
- iOS
- BlackBerry
- Windows Mobile
- References

Terminology

- Integrated Development Environment (IDE)
  - A software application that provides comprehensive facilities to developers for software development
  - Normally consists of a source code editor, build automation tools and a debugger
- Application Programming Interface (API)
  - A set of routines/protocols/tools for building software apps
  - Expresses a software component in terms of its operations/inputs/outputs/ underlying types
  - Defines functionalities that are independent of their respective implementations

Terminology

- Software Development Kit (SDK)
  - Allows you to build against the headers and libraries of an OS version other than the one you're running on e.g.; you can build for OS X version 10.4 while running on OS X version 10.6
- Native Development Kit (NDK)
  - A toolset that allows you to implement parts of your app using native-code languages such as C and C++
  - For certain types of apps, this can be helpful so you can reuse existing code libraries written in these languages

Terminology

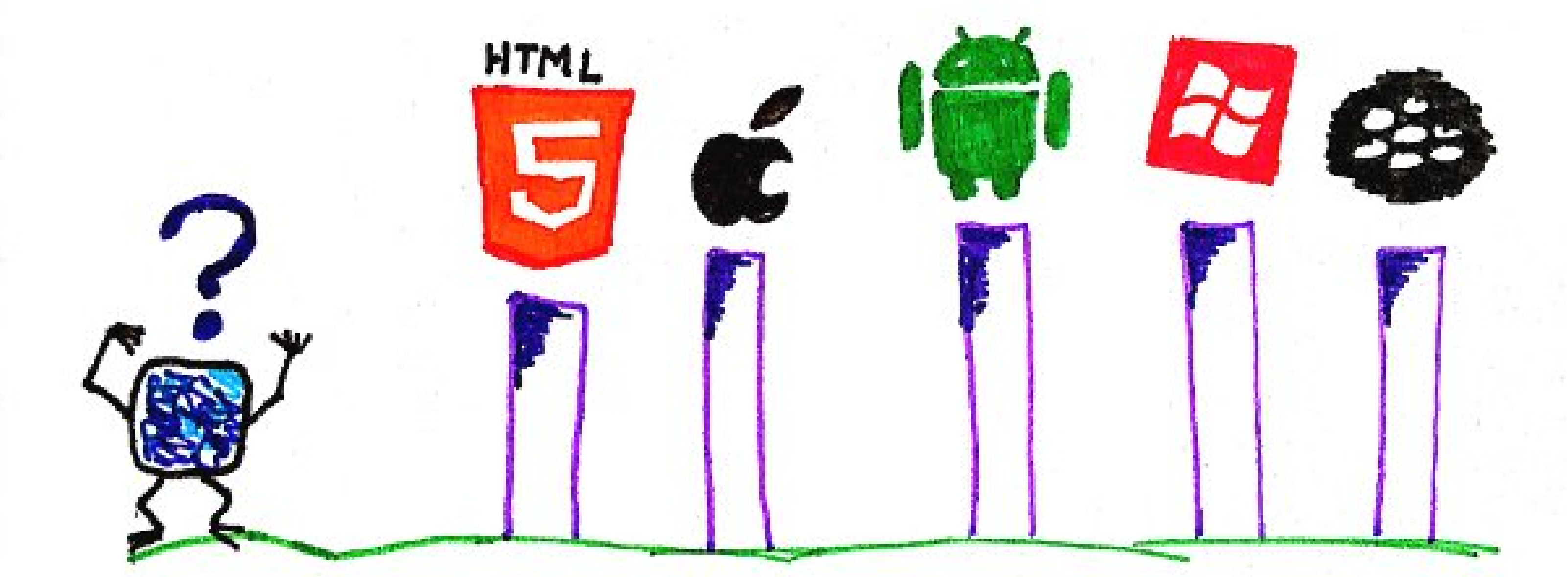
- Emulator
  - E.g.; Android SDK includes a mobile device emulator
  - A virtual mobile device that runs on your computer
  - Lets you develop and test Android apps without using a physical device

Developing Mobile Apps

- A tricky business
- Needs platform specific tools/APIs

Developing Mobile Apps

- Write code in different languages/different platforms



Platforms

1. iOS for iPhone/iPad/iPod Touch apps
2. Android open source platform by Google
3. BlackBerry by RIM
4. Windows Mobile from Microsoft

Developing Outline

- Building a simple app
- Running in the simulator
- Adding a browser control
- Building for the device
- Distribution options and requirements

Outline

- Smartphones
- Developing Mobile Applications
- **Android**
- iOS
- BlackBerry
- Windows Mobile
- References

Android

- Released under the open source Apache License
- Built on Linux kernel version 2.6
- A project of the Open Handset Alliance (OHA)
- Founded by Google

Rich Set of Features

- 2D and 3D graphics
- Good media support for common audio/video/image formats
- Animated transitions and high-resolution
- Colorful graphics are integrated in OS
- Web browser is based on the powerful WebKit engine
- Multitasking of applications

Android Development

Android	
IDE	Android Studio
SDK	Java/C,C++ since Android NDK

Android Development

- On Windows/Linux/Mac platforms
- No Java Virtual Machine on the platform
- Java classes are recompiled in to Dalvik bytecode and run on a Dalvik virtual machine
- C/C++
    - To reuse existing code
    - To gain performance

Outline

- Smartphones
- Developing Mobile Applications
- Android
- **iOS**
- BlackBerry
- Windows Mobile
- References

iOS Development

- iOS
  - Advanced OS
  - With iOS SDK and Xcode IDE creates revolutionary mobile apps

iOS	
IDE	Xcode
SDK	Objective-C/C/C++/Fortran/Java/Objective-C++ AppleScript/Python/Ruby

Xcode

- Xcode suite includes Interface Builder and Instruments
  - Interface Builder helps you create user interfaces for your app
  - Instruments provides a thorough analysis of your app’s
    - Runtime performance
    - Memory usage
    - Allowing you to efficiently find memory leaks and bottlenecks to help improve the user experience

Design Patterns

1. Model-View-Controller (MVC) pattern

    - Central to a good design for any iOS app

    - a way to separate your code into three functionally independent areas

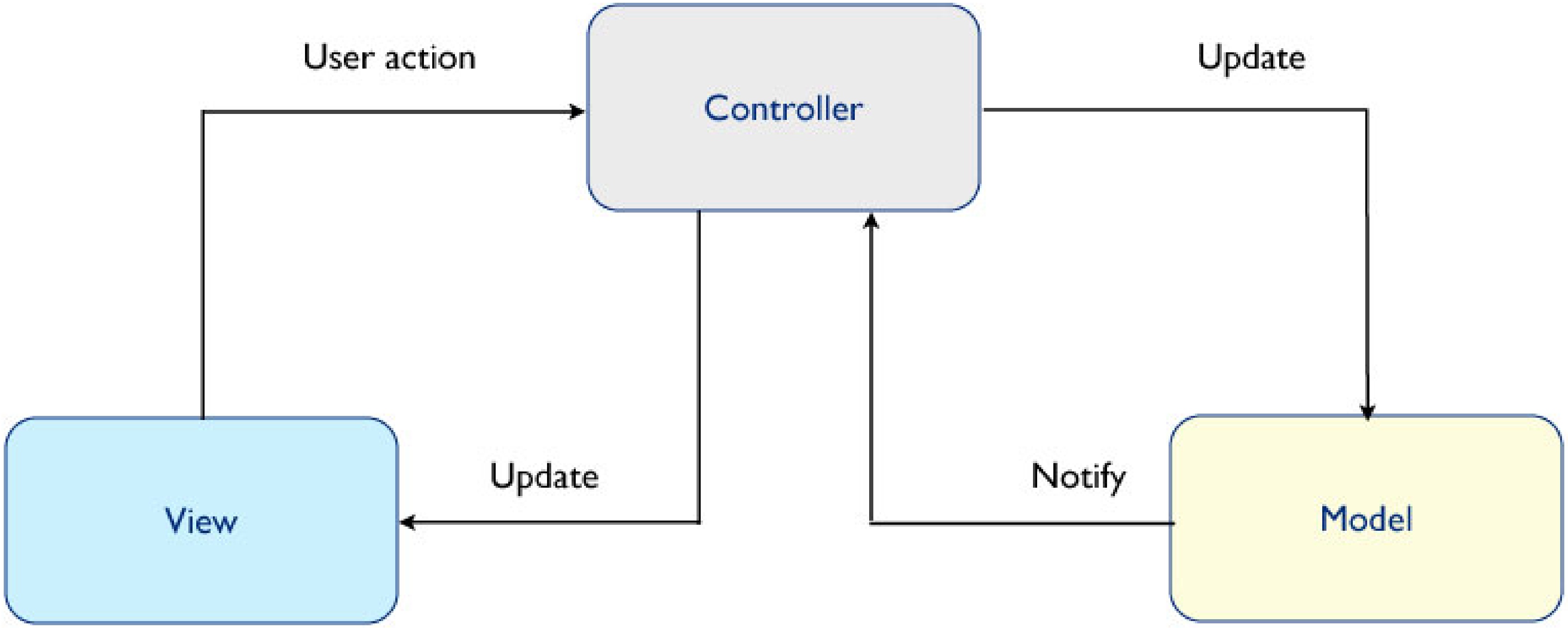
    - Assigns the objects in an app to one of three roles: model, view, or controller

    - The main purpose for MVC is reusability where you can reuse the same model for different views

MVC

- Models
  - Keep track of your app’s data
- Views
  - Display your user interface and make up the content of an app
- Controllers
  - Manage your views by responding to user actions and populating views with content from the data model
  - Serve as a gateway for communication between the model and views

MVC



Design Patterns

2. Target-Action

   - Conceptually simple design in which one object sends a message to another object when a specific event occurs

   - Action message is a selector defined in source code

   - Target (object that receives the message) is an object capable of performing the action, typically a view controller

   - Object that sends the action message is usually

     - a control e.g.; a button/ slider/switch

     - can trigger an event in response to user interaction such as tap, drag, or value change

Design Patterns

3. Delegation

   - A simple and powerful pattern in which one object in an app acts on behalf of/or in coordination with another object

   - Delegating object

     - Keeps a reference to the other object (the delegate)
     - The delegating object sends a message to the delegate at appropriate time
     - The message informs the delegate of an event that the delegating object is about to handle/has just handled
     - The delegate may respond to the message by updating the appearance/state of itself or of other objects in the app
       - ☞ In some cases it will return a value that affects how an impending event is handled

iPhone Development

- You will need an Intel-based Macintosh computer running OS X v10.5.7 or later for development
- The latest version of the iPhone SDK
- Verify that your device OSs are up-to-date
- Download iPhone SDK which includes the Xcode IDE/iPhone simulator and a suite of additional tools

Outline

- Smartphones
- Developing Mobile Applications
- Android
- iOS
- **BlackBerry**
- Windows Mobile
- References

BlackBerry

- A product of Research in Motion (RIM)
- Runs a proprietary multitasking OS

BlackBerry	
IDE	Eclips/BlackBerry SDK Plug-in
SDK	Java/Widget SDK

BlackBerry Development

- BlackBerry Web Development
  - The newest offering from RIM using the Widget SDK
  - BlackBerry Widgets are small, discrete, standalone web applications that use HTML/CSS/JavaScript
- Java Application Development
  - Classic way in which BlackBerry apps are developed in Java using MIDP 2.0/CLDC 1.1/RIM’s proprietary APIs
  - Although the BlackBerry tools are based on Java, only the Windows 32-bit OS is really supported for development

BlackBerry Enterprise Server (BES)

- Provides advanced functionality for IT administrators e.g.,
  - ☞ Deploy and update applications
  - ☞ Set policies for devices
  - ☞ Most importantly, synchronize email/calendar entries/contacts/tasks wirelessly using push technology
  - ☼ BES is one of the reasons the BlackBerry is so dominant in enterprise market

Outline

- Smartphones
- Developing Mobile Applications
- Android
- iOS
- BlackBerry
- **Windows Mobile**
- References

Windows Mobile

- Provides a more desktop-like user experience than other smartphones

Windows Mobile	
IDE	Microsoft Visual Studio/Expression Blend
SDK	C++/C#, Silverlight/XNA since Windows Phone 7

Windows Phone 7

- Forthcoming Windows Mobile platform
- Provides a user experience better suited to mobile use patterns
- Provides support for app/game development using Silverlight/XNA respectively in addition to C++/C#-based apps with .NET Compact Framework
- Microsoft Visual Studio 2010/Expression Blend 4 for Windows Phone: Primary tools for Windows Phone 7 development

Comparison of Platforms

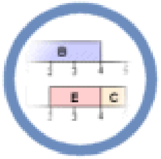
Platforms	iOS	Android	Windows Mobile	BlackBerry
IDE	Xcode	Android Studio	Microsoft Visual Studio Expression Blend	Eclipse BlackBerry SDK Plugin
SDK	Objective-C C/C++/fortran/Java/Objective-C++ AppleScript/Python Ruby	Java C/C++ since Android NDK	C++/C# Silverlight/XNA since Windows Phone 7	Java Widget SDK (HTML/CSS/JavaScript)

Outline

- Smartphones
- Developing Mobile Applications
- Android
- iOS
- BlackBerry
- Windows Mobile
- **References**

References

- Lundrigan, L., Graupera, V.,, Allen, S. (2010). Pro smartphone cross-platform development - iPhone, BlackBerry, Windows Mobile, and Android Development and Distriubution. PA USA: Apress.
- Android: https://developer.android.com/index.html
- iOS: http://developer.apple.com/iphone
- BlackBerry: http://na.blackberry.com/eng/developers/
- Windows Mobile: https://dev.windows.com/en-us/getstarted



Lab ”3”
Android Studio

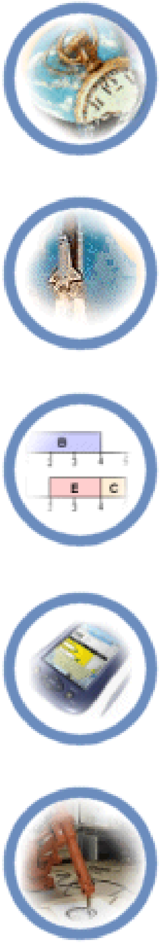
<lecturer, date>

Lab ”3”

- Lab3 is about installation of Android IDE and getting familiar with Android and Java programming through a basic Android application development (*HelloWorld*).

  ✓ Download and install Android Studio
  https://developer.android.com/sdk/index.html

  ✓ Download and install the latest SDK tools and platforms using SDK Manager
  https://developer.android.com/tools/help/sdk-manager.html

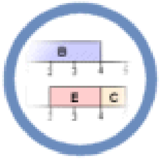


Seminar ”3”
Mobile App Development
(Android, iOS, BlackBerry, Windows Mobile)

<lecturer, date>

Seminar ”3”

- Make a report of Android Studio features and work flow
- Android Studio: https://developer.android.com/tools/studio/index.html



Mini-Project ”3”
Mobile App Development
(Android, iOS, BlackBerry, Windows Mobile)

<lecturer, date>

Mini-Project ”3”

- Prepare a short report of Xcode, its features and interface builder
- Resources
  - Overview & Features
    - https://developer.apple.com/library/mac/documentation/ToolsLanguages/Conceptual/Xcode_Overview/Xcode_Overview.pdf
    - https://developer.apple.com/xcode/
  - Download Xcode: https://developer.apple.com/xcode/downloads/