Background & Related Work
CSC 590

James Reed
December 4, 2009

1 Mobile Application Development

Mobile Application Development is the creation of software applications targeted specifically for mobile devices that may or may not rely on a cellular infrastructure[2]. Developers of mobile applications tend to target niche markets, in particular, those devices which tend to be most popular. Currently, the most popular of the mobile devices are cell phones. While it is true that there are many similarities between developing a mobile application and developing any other application, there are just as many differences, if not more. In the section that follows I intend to highlight those differences in a way that helps describe what exactly developing a mobile application means.

1.1 Technical Challenges and Dependencies of Mobile Application Development

End users of cell phones the world over have their choice of a large number of cellular network operators to choose from. In each network, there are dozens of different cell phone models being released each year. The market is flooded with different models and manufacturers of cell phones. Additionally, each of these phones has different hardware capabilities, which requires at least different versions of operating systems, if not different operating systems entirely. Each of these differing version or systems, on which mobile applications must run, in turn specifies their own Application Programming Interfaces (APIs) which provide an application with functional access to a devices hardware. On top of this, it has become common place to supply a Development Platform, which in turn will have its own set of versions and APIs. Intended to alleviate some of the dependencies surrounding device specific API, the varying versions of development platforms end up causing just as many problems as they solve[2].

With all of the different possible combinations of networks, devices, operating systems, and development platforms its no wonder that mobile application development is generally done for a niche market. Typically, a mobile application
is usually restricted to a particular combination of development platforms, operating system, device, and network[4]. Furthermore the functionality that an application can offer depends on the functionality that each component provides. For instance, a GPS application can only be implemented on a development platform that offers GPS APIs for a GPS enabled device that runs on a network that offers GPS. So with the lack of a common standard, its no wonder why development of Mobile Applications is generally aimed at niche markets.

Successful Mobile Application Development depends on identifying a niche market to develop, which then shapes the technologies for which development is performed. In turn, the technology a developer chooses results in the use of a mix of different APIs at different technology levels. The APIs used to develop a mobile application then affects the devices and networks an application will run on. Ultimately, the devices and networks for which an application is developed affects how an application can be distributed. No matter how good the application is, it cannot be successful if its developed for a device or network that no one uses.

Adding to these development dependencies is the certification requirement that most device manufacturers, network operators, or platform developers impose on mobile applications before allowing them to be distributed to the end users. The certification process gives the certifying party the power to ensure the security of applications that will be distributed to their customers as well as establish a level of quality and interoperability. The certification processes vary depending on the approving party and add another variable to the development process.

Lastly, adding to the list of dependencies mobile developers must be aware of is the market for devices themselves. Most of the popular devices on the market tend to be at least a year old. This is due to contractual obligations imposed by network operators. Therefore it generally tends to take about a year before the latest devices with the newest technologies find their way into majority of end users[2]. While the number of users who pre-pay for their network service and are free to switch networks, and thus technologies, is on the rise, the older mobile devices are cheaper as well. It therefore tends to be more profitable to develop for older devices and that have a larger market share and cheaper end user costs. This is not to say that developers should be complacent with current technologies. Rather, developers should use the year that it takes a technology to penetrate the market to familiarize themselves with it and develop a strategy, while at the same time continue development with their core technologies.

2 Android Mobile Development Platform

Google describes the Android Mobile Development Platform as, The first truly open and comprehensive platform for mobile devices, all of the software run a mobile phone but without the proprietary obstacles that have hindered mobile
innovation[7]. The Android platform, often referred to as just Android, is what some might think of as the first step towards a standard specification for mobile phones. Android provides[3]:

- A loose description of what the hardware (or phone) should be capable of in order to support the software stack.
- A Linux kernel providing the hardware interface, memory management, process control, etc.
- Open source libraries for application development.
- A run time to host and execute Android applications, which is the Dalvik Virtual Machine, like a JVM.
- An application framework for exposing system services to applications.
- A User Interface.
- Some pre-installed applications (phone, messaging, maps, etc...).
- A Software Development Kit.

In the year following its release by Google Inc., Android has quickly acquired wide-spread market adoption from major United States network operators like TMobile and Verizon, as well as device manufacturers such as HTC, LG, Samsung, Sony Ericsson, and Motorola. Each device manufacturer is responsible for compiling creating the Linux kernel which their device will run. The kernel then makes use of the standard run-time host. Developers then write applications in Java using the Open source libraries and application framework for getting system services. However, Android does not make use of the Java Virtual Machine (JVM) to run these applications. Instead, Android uses an optimized version of the JVM called the Dalvik Virtual Machine (DVM) that uses its own byte code. The Java then gets compiled down into Dalvik bytecode[3]. In the end what you get is a standard set of APIs that a developer can use to write mobile applications, which will run on any mobile device and network that supports the Android platform.

3 Mobile Application Development and Computer Science Education

In much the same way development of computer games has been used as a tool to teach Computer Science students while at the same time improve recruitment and retention[1], Mobile Application development can be used as an affective medium for education. In one such instance, a course was developed for new Computer Science students that focused on game development for mobile devices[5]. The choice to develop games for mobile devices was based on the idea that such games are inherently simpler, though none the less enthralling,
since the devices they run on tend to be more limited in terms of user interfaces. The games could be developed in relatively short time frame with a reasonable set of Java programming skills. More importantly, the course was able to use the topic of mobile game development, one in which the students had an interest in, to exposing the students to more advanced topics. Such topics included Graphics, Human Computer Interaction Research, Data Structures, Artificial Intelligence, Compute Networking, and Software Engineering practices. In another example, a course specifically designed to educate about Mobile Development was created[6]. This Mobile Development course, which received positive feedback, focused on more advanced technical topics such as Web Services, Operating Systems, and Information and Network Security.

College aged computer science students see mobile devices as more than a means for communication. For students, mobile devices are becoming an ever increasing part of their day to day lives. New and inventive applications are coming out all the time that transform the mobile phone into something else, like a remote control for your computer, a movie recommender, or a grocery store scanner for comparing prices. With the right application a cell phone can become anything. This implies that with the right application we can at the very least spark interest in, if not relate to their personal lives, arbitrarily complex computer science topics to students through the development of the application.

References


[6] Qusay H. Mahmoud, Thanh Ngo, Razieh Niazi, Pawel Popowicz, Robert Sydoryshyn, Matthew Wilks, and Dave Dietz. An academic kit for integrat-