Smartphone Software Development Course Design Based on Android

Wei Hu, Tianzhou Chen, Qingsong Shi and Xueqing Lou
College of Computer Science
Zhejiang University
Hangzhou, Zhejiang, P.R.China
ehu@zju.edu.cn

Abstract—Mobile computing is popular when wireless network has been deployed almost everywhere. Smartphones have been the important tools in our society for the abundant functions including communication, entertainment and online office etc as the pivotal devices of mobile computing. Smartphone software development has also become more important than before. Android is one of the emerging leading operating systems for smartphones as an open source system platform. Many smartphones have adopted this platform and more smartphones will do so in the future. It is also an emerging problem on how to develop software for smartphones based on Android and those platforms like it. We propose smartphone software development course design based on Android in this paper. What this course focuses is how to teach the development technology to Students. The course design has two parts including the syllabus design and hands-on lab design. At the same time, the innovations are also described in detail and these innovations play a key role in the teaching.

Keywords-smartphone; software development; course design; Android

I. INTRODUCTION

With the rapid progress of embedded system and wireless network, mobile computing has been popular according to its flexibility and portability. Mobile computing has brought tremendous changes in our lives. As wireless network has been deployed almost everywhere, smartphones are used more and more widely. Smartphone is not only a cell phone but also the portable computing platform [1]. These smart devices have been the important tools in our society for their abundant functions in communication, entertainment, online office and many other fields [2 -4]. In the future, smartphones will be more powerful and have new functionalities. They have the potential to replace desktop and laptops [5].

Smartphones are different devices from the desktop computers and they are embedded systems. Such systems are resource limited compared to common computers. They are battery-driven with small memory and slow embedded processors [6]. Software including operating system runs on such devices should be energy efficient with fewer memory footprints and more optimizations. The embedded operating systems designed for smartphones play a key role as the heart of software system [7]. S.P. Hall and E. Anderson [8] has described the operating systems for mobile computing. The features of the different embedded operating systems for smartphones are presented in their paper including RIM (Research In Motion) of Blackberry, Windows Mobile, iPhone, Symbian and Android. These operating systems are also the common operating systems in the market. According to their views, software development relies on these operating systems and Android also has the potential to tackle the problems that the users have on their phones.

When the operating systems are introduced for smartphones, how to teach Students for the programming is also emerging as an issue which should be considered in the universities. In fact, smartphones have been introduced into education in some universities. In [9], smartphones were used as the instrument in the course to involve Students to solve real-world problems. [10] described smartphone based prototype application for e-learning. Such techniques will help Students retrieve the useful information. And [11] also presented the design for distance education through smartphone. Smartphones are important platforms in teaching.

In this paper, we describe our design of smartphone software development course design based on Android. In this course, what is smartphone and how to program for smartphone applications are the main two focuses. Students will learn what is the software platform of smartphone as well as how to solve programming problems on smartphone. We also present some innovative methods to help students to study for the importance of the practice.

This paper is organized as follows. Section 2 describes the motivation why the course is designed. Section 3 describes course design including the syllabus design and hands-on lab design. And Section 4 presents the innovations in the course design and implementation. At last, Section 5 offers the conclusions and future work.

II. MOTIVATION

With the popularity of smartphones, it needs more programmers who have the capabilities on software design for smartphones. So this is also the task of the universities. In current course design, mobile computing and embedded system are both proposed as elective contents in curricula of operating system and net centric computing respectively in CSC 2008 [12]. Typically, this task is arranged in the course
of embedded system or mobile computing. While in such
courses, operating system and programming for smartphones
are only small parts compared to the whole courses. Thus
students have to study by themselves if they want to learn
more knowledge. But they will have many difficulties
without help. They should be offered an opportunity for
further study guided by teachers.

Based on our experiences in the course design of
embedded system [13, 14] and multicore programming [15],
we begin our design on the course of smartphone software
development (hereinafter this course will be abbreviated as
“SSDCA”, which means “smartphone software development
course based on Android”). Our concern is how to combine
the theory and practice in this course. Students need to learn
the theory, but also to spend a lot of time to programming
practice. At the same time, Android is selected as the basic
software platform. There are several different embedded
operating systems for smartphones in market. Android is
based on Linux, which is designed and published by Google
in 2007 [16] and it is used by many smartphone
manufacturers [17]. Now the Open Handset Alliance (OHA)
[18], which is organized by Google, is working to make
Android become top operating system for smartphone in the
world [8]. And Android is free for all users based on open
source model. It is helpful to students to learn the underlying
design of this platform and we can obtain enough copies of
Android easily. That is why Android is selected.

III. COURSE DESIGN

A. Design Principles

Though many students want to learn how to
programming for smartphones, such a course will not be
obligatory for all students. Not all the students will be
interesting in this topic. So SSDCA is defined as an elective
course for students. During the design, we follow the
following principles:

• **Design Principle 1 (DS1)**: this course should be
able to help students to establish an overall
framework of smartphone software development.
Thus students will have the basic ideas on how to
programming for smartphones and it is helpful to
further learning.

• **Design Principle 2 (DS2)**: Teaching content should
be more emphasis on practice rather than theory.
Though theory and practice will be combined in this
course, practice is more important than theory for
programming especially when these students are
undergraduates.

• **Design Principle 3 (DS3)**: what students learn from
our course should be able to meet the needs from
industries. When students enrolled in this course,
what they will learn should be conducive to their
employment.

According to these design principles, when the course is
designed, we first determine the syllabus. Because the
advantages of Android as shown in Section 2, this platform
is selected as the core teaching content in this course. And
then we design the hands-on labs based on Android. We also
propose some teaching methods with accordance with the
characteristics of the course. These innovations in teaching
methods will help us to improve the teaching effect.

B. Syllabus

According to our DS1, we first find out what should be
arranged as the teaching contents in SSDCA and then these
contents will be organized as the syllabus. The establishment
of teaching contents is based on the understanding of the
needs of the industries and our teaching/research experiences.
The basic contents of theory teaching are divided into seven
parts on this basis as shown in Fig. 1.

![Figure 1. Basic Teaching Contents](image)

In Part I, there are four topics including the introduction
of smartphone, hardware architecture of smartphone,
software platform of smartphone and application
development of smartphone. In this part, we will give an
overview of smartphone including hardware and software to
students. In Part II, different mainstream operating systems
for smartphone are introduced to students including Android,
Linux, Symbian, Windows Mobile and Palm OS etc. From
this part, students will understand the basic situation of these
operating systems.

From Part III to Part VII, Android is the main content.
First, the architecture of Android is taught to students to help
them have a rough understanding of the overall system.
Students will know the basic layers of Android including
Android has four layers including application layer,
application framework layer, library layer and Linux kernel
layer. And then the kernel of Android will be presented to
students including process management, memory
management, module mechanism, network protocol and the
security. This part introduces the fundamental design
concepts of Android. In Part V, the library and run-time
environment are illustrated. Library is used to provide service for the developers through application framework layer. And the run-time environment consists of core library, which provides the functions of JAVA language, and JVM, which provides virtual machine mechanism in Android. Part VI focuses on the component mechanism and the system service in application framework layer. At last, the development environment, SDK and API are introduced to students, in which how to develop Widgets on Android is shown as the development examples.

Figure 2. Three types of hands-on labs

As we can see from Figure 1, the teaching contents cover what smartphone is, the main operating system for smartphone and the detail of Android. The contents from Part I to Part VII are theory teaching contents. We also design hands-on labs for practice which will be describe in the next sub-section.

C. Hands-on Lab Design

Practice is very important for programming. In SDDCA, students will master how to programming for smartphone based on Android through practice. We divide the hands-on labs into three types as shown in Figure 2.

The three types are basic hands-on lab (Bas), development hands-on lab (Dev) and comprehensive hands-on lab (Com). The first two types are unit hands-on labs and have close relationship with the theory teaching. Each hands-on lab will be associated with a special content theory teaching, while the last type of hands-on lab is only a single one. Students need to use what they have learned in an integrated manner to complete this hands-on lab. The details of these hands-on labs are shown in Table 1.

The hands-on labs will be arranged with 64 credit hours including 32 credit hours for basic hands-on labs and development hands-on labs and 32 credit hours for comprehensive hands-on labs. Our hands-on lab design is consistent with DS2.

<table>
<thead>
<tr>
<th>Type</th>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bas</td>
<td>Development Platform Build of Android SDK</td>
<td>Installation of Android SDK and Eclipse ADT plug-ins to be familiar with Android SDK and the platform</td>
</tr>
<tr>
<td>Bas</td>
<td>Use of Dalvik JVM</td>
<td>To be familiar with Dalvik JVM, the structures of DEX file and Apk file, compile, decompile and package of Apk</td>
</tr>
<tr>
<td>Bas</td>
<td>Preparatory Widget Development</td>
<td>Installation of Eclipse WDT plug-ins to be familiar with the development follow of Widget on Android</td>
</tr>
<tr>
<td>Dev</td>
<td>Basic Development of Android UI</td>
<td>To be familiar with the Android Activity Class and programming on Android UI</td>
</tr>
<tr>
<td>Dev</td>
<td>Advanced Development of Android UI</td>
<td>To master the multi-resolution support and style customization of Android UI</td>
</tr>
<tr>
<td>Dev</td>
<td>Database Development</td>
<td>To master the basic operations of SQLite database of Android including create, management, basic instructions and transaction process</td>
</tr>
<tr>
<td>Dev</td>
<td>Location-related API</td>
<td>To be familiar with location-related API of Android and master the programming of location-based applications</td>
</tr>
<tr>
<td>Dev</td>
<td>Basic Network-based Development</td>
<td>To be familiar with the common APIs of network-based development on Android including standard JAVA API, Apache interface, Android interface and standard C/C++ API</td>
</tr>
<tr>
<td>Dev</td>
<td>Web Server Development</td>
<td>To master programming on mobile web server</td>
</tr>
<tr>
<td>Dev</td>
<td>Online Application Development</td>
<td>To master the thread management of online application and interface update</td>
</tr>
<tr>
<td>Dev</td>
<td>Music Player</td>
<td>To understand Android multimedia framework and basic techniques of music player development. To be familiar with the concepts of Application, Activity, Service, Intent, Broadcast Receiver, etc. through the development of a simple music player. And how to analyze Android applications through MAT tools</td>
</tr>
<tr>
<td>Dev</td>
<td>Advanced Widget Development</td>
<td>To master the techniques of Widget development for AppWidget and WebWidget</td>
</tr>
<tr>
<td>Dev</td>
<td>Animation Development</td>
<td>To master the basic development methods of Android animation</td>
</tr>
<tr>
<td>Dev</td>
<td>MediaRecorder API Programming</td>
<td>To be familiar with MediaRecorder API and master MediaRecorder programming</td>
</tr>
<tr>
<td>Dev</td>
<td>MediaPlayer API Programming</td>
<td>To be familiar with MediaPlayer API and master MediaPlayer programming</td>
</tr>
<tr>
<td>Dev</td>
<td>VBA Porting</td>
<td>To understand Android UI further through porting VBA to Android</td>
</tr>
<tr>
<td>Com</td>
<td>Comprehensive Hands-on Lab</td>
<td>To complete a novel application by using Android APIs</td>
</tr>
</tbody>
</table>
IV. INNOVATIONS IN TEACHING METHOD

As mentioned in both Section 2 and DS3 in Section 3, SDDCA provides abundant practice design and can help students for their employments. This is great challenge for us to achieve this goal. So we design some innovative teaching methods for SDDCA according to our experience in teaching as shown in Figure 3.

These innovations are used with the traditional teaching methods in class. Though traditional teaching methods are important in teaching, these innovations are also necessary for SDDCA to improve the teaching effect. In this section, these innovations in teaching methods will be described in detail.

A. Collaboration with Industry

Programming for smartphone will meet the needs from industry. Though we have experienced teachers in teaching and research in related fields, the experience and demand from businesses are also very important for SDDCA. When SDDCA is designed, we have established partnerships with Google to improve the teaching effect. Android is from Google and the engineers of this company will provide enough support for our SDDCA.

We will invite the senior engineers from industry to join our teaching team. These engineers are not the formal teachers of this course, but they are a great help in teaching. The teachers of SDDCA can discuss on different topics in smartphone related fields with these engineers. The engineers from industry can give us the real needs from industry directly. This will help us to improve our course design. And at the same time, these engineers will also give lectures to students periodically. They can tell students the progress of Android in detail and how to improve the efficiency of their programming. These lectures will be useful and important complement to SDDCA. Companies will also provide the latest materials on smartphone, Android and programming to SDDCA. The technology for smartphone is still in rapid development. The teaching contents should be updated according to the new advances. The support from industry will help us to complete such work.

B. Community Integrated Education

As the number of teachers and resource constraints, SDDCA can only cover part of the overall contents of smartphone software programming. Android is an open system and has been released to the users. Many programmers who have experienced in programming on Android have organized as special community for Android. Such communities are public and can be accessed online. Different programmers from all over the world will provide their programming experiences on these websites including the difficulties that they have met and the corresponding solutions. Such experiences come from the real world and will help students to study better. SDDCA will use such online communities as an important method for teaching. We will first find out the reputable communities and then select some candidates which are suitable for students.

There are also some online forums in our campus. Such forums are used as the internal information platform. Special sections are created and opened to students for the discussion in campus. Students can ask questions in such sections and these questions will be answered by some other students. Different students can exchange their ideas and experiences through the discussion and Q&A. The teachers of SDDCA will also be the users of such sections. They will guide the discussion, provide the problem-solving ideas or initiate some discussion on special issues. This is our campus community.

Students are also organized through special student technology club to share their experiences face to face. In such club, students can discuss in groups on different topics. The teachers or Ph.D candidates will be the tutors in some groups. They can solve some problems by themselves in these groups or through the help from the tutors. The students in club can also be organized as development teams by themselves. Such teams can design and implement the applications for smartphone on Android or some other operating systems. This is our physical community.

The mentioned communities are organized as a multilayer community integrated education as shown in Figure 4. The different communities can complement each other. Students can also be the contributors of Android community through their work and activities in campus community and physical community. The information from Android community can also help the construction of the other two communities.

C. Attractive Supplementary Contents

In addition to the above methods, we also provide abundant practice for students. Students require a lot of practice to learn programming. Though SDDCA can provide
64 credit hours to students, it is still not enough. At the same time, students will need more practice from real world. We provide the following three means.

**Special contest.** We will design special contests based on SDDCA. Students are encouraged to organize their own teams. We will provide some topics for these teams or they can put forward their own ideas. These teams must complete the design and implementation of the software, which will be very small perhaps. Novel ideas will be evaluated as the better ones than those without innovations. Students can obtain the experience from such contests like in real world. And students will also understand what innovation is and they can come up with better and novel ideas.

**Special training.** We provide special training to students. Senior engineers will be invited for such training. Different from the lectures described in Sub-Section A of Section 3, special training will only focus on the detail of the programming. Each training will have only one topic and this topic will be explained in depth. Such training can cover the details which can not be introduced in SDDCA for limited resources.

**Special project.** We provide the special projects to students. Such projects are funded by our university through Student Research and Training Program (SRTP). A SRTP project should be completed in one year. Students can apply for SRTP projects freely. If students apply for projects based on SDDCA, they can obtain effective guidance. But we will never give them the solutions directly. They should complete the projects independently. What we provide is the problem-solving ideas and technology support with limitations.

V. CONCLUSIONS AND FUTURE WORK

With the popularity of smartphones, programming on such devices is also needed by industry. It is a goal of the universities on how to teach the smartphone software development. In this paper, we share our experience in design SDDCA, which is smartphone software development course based on Android. We first propose three design principles and then design SDDCA including the syllabus and hands-on labs according to these principles. We also present our innovations in teaching method, which comply with the design principles and characteristics of programming on smartphone. Such innovations can help us to improve the teaching effects and help students master the programming on smartphone.

We still have a lot of work to do in the future. What we must complete is in the follows: 1) A online website for SDDCA should be constructed as the portal; 2) More resources should be provided to students for their study; 3) the videos of SDDCA should be completed and published to students online.

ACKNOWLEDGMENT

This work was supported by the Special Funds for Key Program of the China No. 2009ZX01039-002-001-04 and the Special Funds for Key Program of the China No.2009ZX03001-016.

REFERENCES


