RCPM – A Mobile Application for Random Classroom Participation Management

Jorge A. Torres Rodríguez
Master of Engineering in Computer Engineering
Nelliud Torres Batista, DBA
Electrical & Computer Engineering and Computer Science Department
Polytechnic University of Puerto Rico

Abstract — *For many courses in college, class participation is a key element for grading. In environments where a class requires constant feedback from students and their engagements in both dialog and discussions, a need has been identified for a way that would help professors manage students' class sessions. Using Android Studio as the development platform, an application has been devised that can calculate and store student's participation details including awarded points. RCPM, which is an acronym for Random Classroom Participation Management, is a tool designed to help faculty manage such educational dynamics regardless of class size. It is a smart phone based program that addresses the need for reliably annotating student participation details while selecting a student at random for questioning which can help to promote a suitable atmosphere of fairness among classmates through an equitable distribution of question and answer engagements.*

Key Terms — *android application, classroom participation, mobile platform, question distribution, random selection.*

INTRODUCTION

Class participation is beneficial to the learning process of an individual [1]. It helps students to communicate with peers, to listen, exposes them to different points of view and helps them in retaining information [1]. It is very common for many courses at college level to adhere to some type of participation system. Professors tend to incorporate such dynamics in their classrooms and their grading process in some shape or form. More often than not, the scoring methods that accompany this type of engagements are recorded manually using pen and paper. This may suffice for one class with a relatively small number of students but it can be cumbersome for a professor whose academic load and class size may vary considerably from term to term.

Not only the number of students per section has to be taken into consideration but distribution of discussion turns in order to avoid consolidation of responsibility [1], where a handful of students are the main participants within a class, needs to be addressed in order to provide an environment of fairness when choosing who gets to answer and elaborate on the next question. The problem lies in the manual manner in which the participation is recorded and the lack of a reliable way for fairly choosing the next student for a question/answer engagement. The lack of an appropriate tool for tracking these details, record student scores and the fair distribution of participation turns is the motivation behind RCPM.

RCPM, which stands for Random Classroom Participation Management, is a mobile application that has been developed using the Android platform as a way of substituting pen and paper based recording systems.
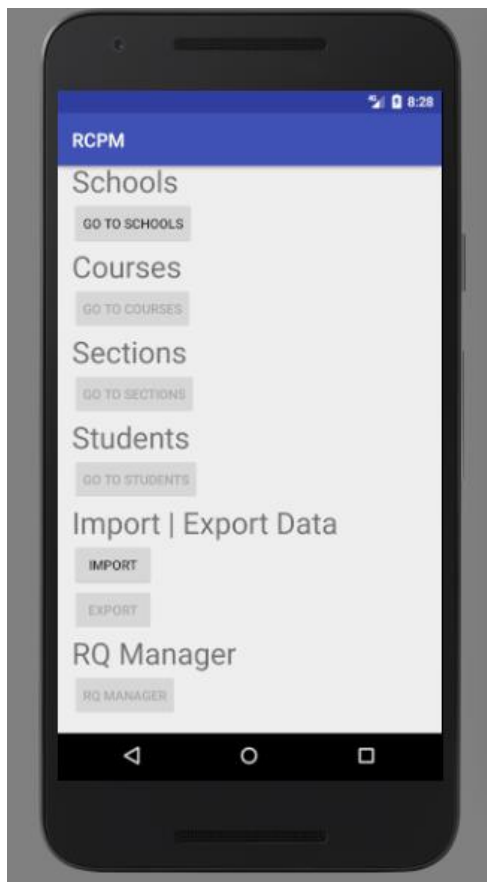
DESIGN AND STRUCTURE

The application addresses three issues that are present when using pen and paper based systems. These issues were identified and defined as follows: time, selection, and digital portability of data. **Time** refers to the time it could take for a professor to choose a student for questioning, record his/her points, the eventual calculation of points per student and the transferring of information to a digital form for reporting purposes at the end of the term for a class that well may be composed of more than 20 students at a time from any assigned section.

Selection refers to the process of randomly selecting a student for questioning. This process relies on the professor's ability to accurately record

which student has been called so that questions can be spread evenly among classmates.

Digital Portability of data refers to having a viable way of transferring all the data gathered from a class/section to a digital form for point's calculation, metrics and archival purposes.

The first step in addressing these issues is to provide the user with a way of inputting the required data into the application. The user can enter the data manually or import two .csv files with a predefined format/order of key fields and a specified name. Both of these options are available via the application's main window. See Figure 1. The quickest way for getting data into the system is to import the two .csv files which will require some preparation before the import. One file needs to be named courses.csv and the other students.csv. Each file has its own set of fields and order to which the user needs to abide to for the import to work. See Figure 2 and Figure 3.

Files need to be saved in the comma separated values format. The files can be transferred to the smart device by downloading them from any internet storage place where they may be stored or by direct usb connection to a computer. The application provides enough functionality for the data to also be entered manually. A common workflow for manually entering data would be as follows:

- Enter schools
- Enter courses and assign each course to its respective school
- Enter sections and assign each section to its respective course
- Enter students and assign each student to his/her respective school, respective courses and sections.

All basic information, with the exception of points, round and notes, can be edited/updated manually. It is advisable to refrain from making any updates to the assignment/pairing of course-school, section-course, student-school/course/section after the RQ Manager has been ran for the first time for a section as this may lead to corruption and possible deletion of data as a user most likely would have to import or enter a new set of details into the application.
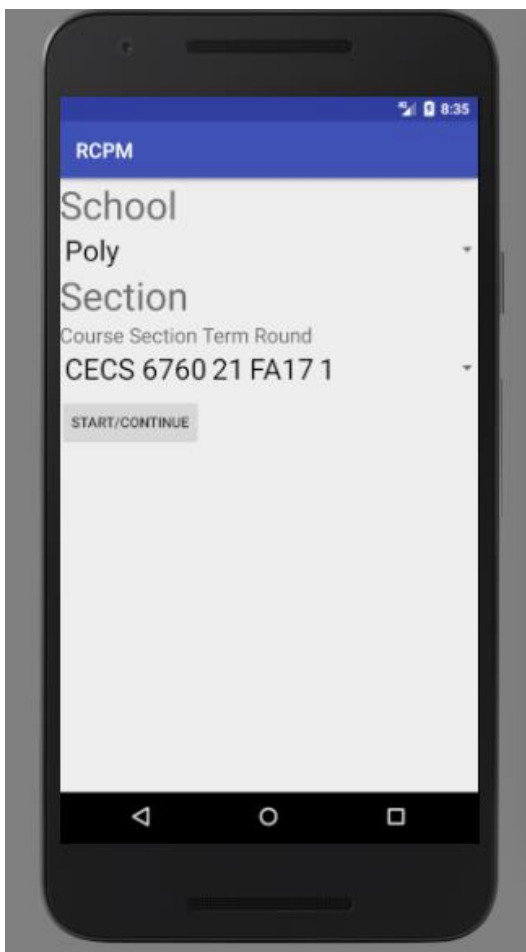

Figure 1
RCPM Main Window
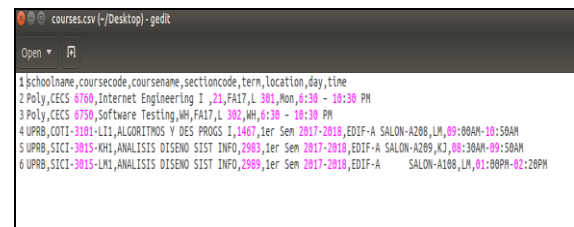
Figure 2
Contents of File Courses.csv
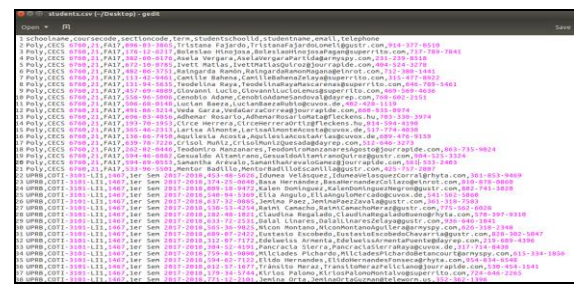
Figure 3
Contents of File Students.csv

The issue of selection is handled by the RQ Manager interface where the user chooses the school and the course-section pair (class) to work with. See Figure 4. As the process is started or continued, the application will randomly select a student for questioning from the pool of students belonging to the selected course-section pair. This screen depicts the current round, elapsed time of the engagement, name of the student, score input area and notes input area. See Figure 5. This benefits the professor in providing a quick and simple way to record the engagement.

Figure 4
RQ Manager Window

Once the professor saves the information for the current student, the application randomly picks the next student for questioning. The combination of randomly selecting a student and then ta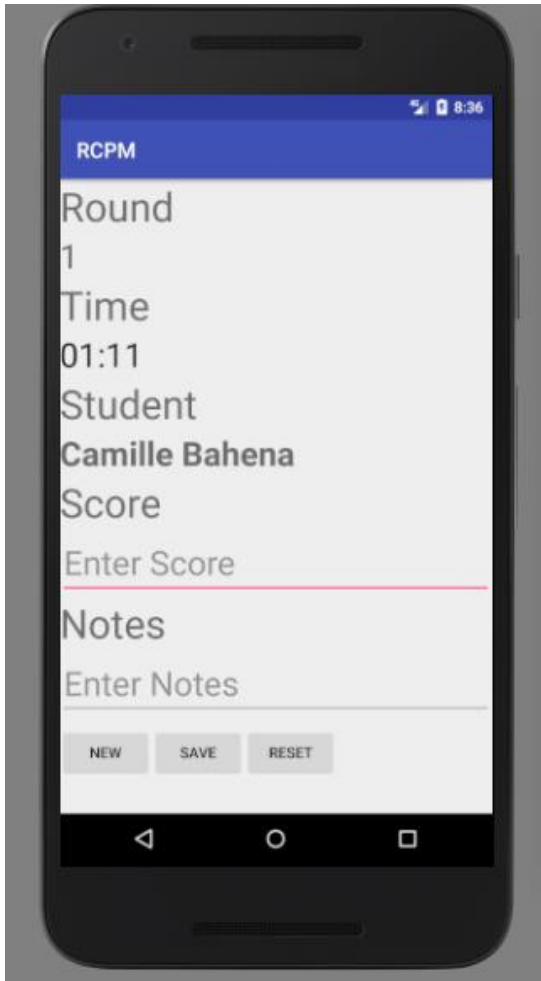king the student out from the pool of individuals until the current round is completed helps to foster an atmosphere of fairness and equal opportunity within the class. Note that a round is considered complete when all students have been called to participate at which point a new round is started.

The third issue to address is the digital portability of the data. In the same way the application provides a mechanism for importing data into its database, it also gives the user the option of exporting the current state of the participation data to a .csv file. This file can later be manipulated with any standard spreadsheet program available to the user. See Figure 6.

The contents of the file are based on a sql query that outputs the information related to students and their corresponding engagement details at a particular date. An instance of a line is composed of the school name, course code, section code, term, student name, school assigned id number, round, participation date, points and notes. Most android devices today do not provide an actual external storage such as an sdcard slot; instead they offer a partition of the internal memory for use as the external storage [2] which most of the times is labeled as: sdcard. It is at this location where the exported file, named participation.csv, gets saved to. The file could then be uploaded to an online storage service for archival purposes or downloaded to the user's computer for report prepping of courses and their respective sections at the end of each term.
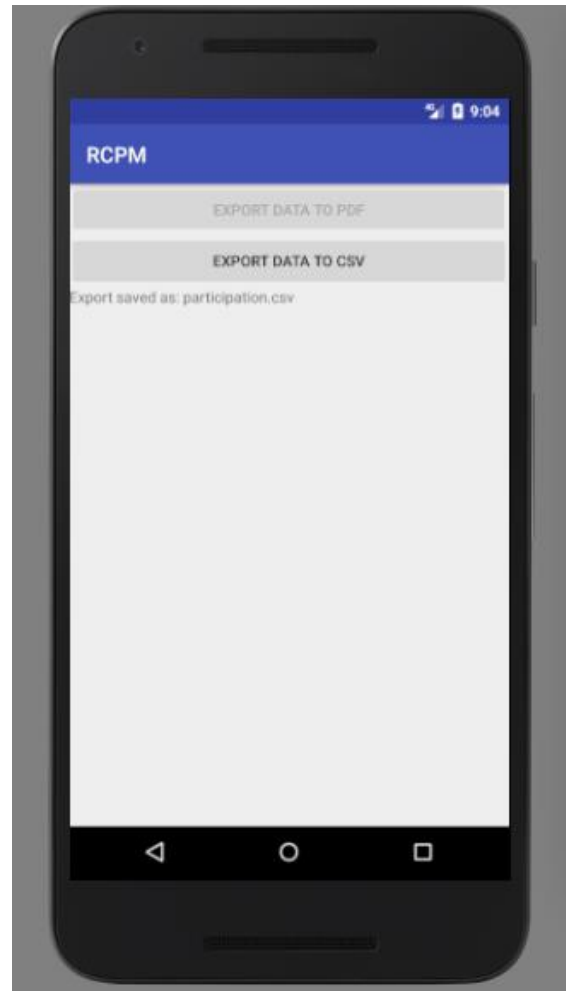
Software Architecture

A definition for Software Architecture can be stated as being an abstraction of a system/application that lets us focus, at a very high level, in its workings, organization and how it is tied together to bring about a desired outcome. An Android application is no different when it comes to its own representation of this concept. The major components of an Android app are: Activities, Services, Content Providers and Broadcast Receivers [3].

Figure 5
RQ Manager Round Window

to a central repository of data that can be shared with multiple applications. **Broadcast Receivers** are system or application processes that pass messages to any other component that is subscribed to receive them.

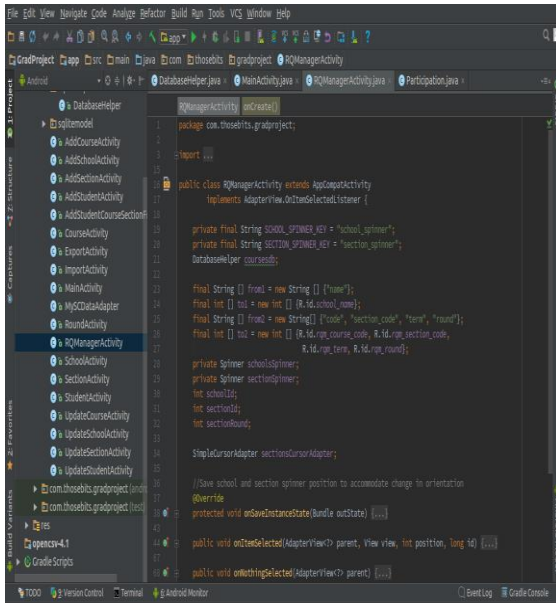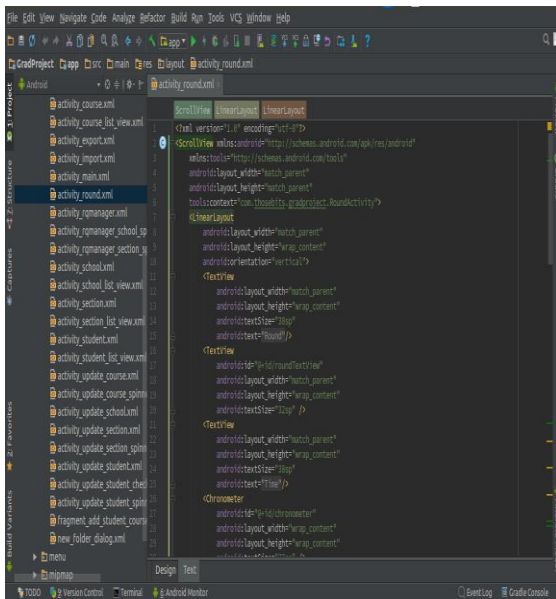
Figure 6
Export Data to CSV Window

Activities are considered one of the core building blocks of an android application [4]. See Figure 7. They are the first components the user will interact with. An app may be composed of one or more activities each of which is a screen providing a graphical means for user interaction. The user interface code is contained in what are called Layout files which hold the structure of the visual representation of an activity. These files are written using a set of xml tags that are Android specific. See Figure 8. This xml code serves to make a separation between the façade of the application and the code that manages its actions. **Services** are processes that run in the background without user interaction they can be started by apps and may persist even when the app is no longer in use. **Content Providers** manage and provide access

RCPM is mainly composed of activities which retrieve and store information in a SQLite database that acts as the application's content provider. Student, school, course, section and round data is saved, deleted and updated using the database as its main data repository. The database follows a relational paradigm. It contains six tables with 1: M relationships. RCPM does not use broadcast receivers.

Figure 7
RQ Manager Activity in Android Studio IDE

Figure 8
XML Tags in Layout file for Round Activity

Android Studio

RCPM was designed, built and programmed using the Android Studio IDE and its software development kit tools. This platform is very appealing for its ease of use, verbosity in providing information to the developer, code completion and its various installation options allowing you to build an app in any of the three major operating systems currently available for the consumer market: Mac OS X, Windows and Linux [5]. One of its key features is its capability to emulate different basic android devices through its emulator interface. Granted, the emulator is not capable of emulating certain functions found in smart phones such as Wi-Fi (API level equal or greater than 25 does support it), Bluetooth, sdcard insert/eject, device attached headphones, near field communication technologies and usb. Still the emulator provides enough smart phone functionalities to test and debug an application such as RCPM. The emulator allows for testing "on virtually any Android device configuration" [6] making this feature of the Android Studio very practical as having access to multiple devices for testing would be near to impossible due to the gamut of different hardware present in the market [7] [8]. Code versioning control was employed throughout the whole process using Git, a distributed revision control system, as Android Studio includes integration capabilities for such system and github.com for code repository storage, retrieval and management. Android Studio and Git are freely available.

CONCLUSION AND FUTURE WORK

We have briefly discussed some of the issues present in recording class participation using pen and paper based methods such as: data not being digitally portable, time constraints when manually annotating participation details and the proper selection of students for a question answer engagement. RCPM suitably addresses these issues by reliably managing student participation details and engagement distribution.

Future work includes providing direct access to students round data for notes and points updates, choosing specific course-section pairs for export, creation and exporting of a report in pdf format depicting summary information per student per section, integration with messaging systems for broadcasting professor to students announcements as well as one to one communication, and porting the application to IOS devices.

REFERENCES

[1] Rocca, K.A., "Student Participation in the College Classroom: An Extended Multidisciplinary Literature Review," in *Communication Education*. vol. 59, No. 2, April 2010, pp. 185-213.

[2] Android Studio. (n. d.). *Using the External Storage* [Online]. Available: https://developer.android.com/guide/topics/data/data-storage.html#filesExternal.

[3] H. Bagheri et al, "Software Architectural Principles in Contemporary Mobile Software: from Conception to Practice", in *Journal of Systems and Software*, vol. 119, Issue C, 2016, pp. 31-44.

[4] Android Studio. (n. d.). *Activities* [Online]. Available: https://developer.android.com/guide/components/activities/index.html

[5] Android Studio. (n. d.). *Download Options* [Online]. Available: https://developer.android.com/studio/index.html.

[6] Android Studio. (n. d.). *Features* [Online]. Available: https://developer.android.com/studio/index.html#features.

[7] International Data Corporation. (2017). *Smartphone OS Market Share, 2017 Q1* [Online]. Available: https://www.idc.com/promo/smartphone-market-share/os.

[8] Android Studio. (n. d.). *Platform Versions, Dashboards* [Online]. Available: https://developer.android.com/about/dashboards/index.html.