DEVELOPMENT AND IMPLEMENTATION OF A TABLET-BASED EXAM APP FOR ENGINEERING COURSES

Kurt Gramoll
Aerospace and Mechanical Engineering
University of Oklahoma

Abstract

For large enrollment courses, such as Statics, Dynamics, Fluid Mechanics, Solid Mechanics, Thermodynamics, the use of computers (laptops and tablets) in the classroom can save instructor time. However, there are numerous problems with computer-based exams, especially when the students are allowed to access online course materials (eBook, homework, etc). Previously, the author used student-supplied laptops for web-based exams in the classroom, but this has become problematic over the years due to the increased sophistication of communication software (Twitter, Facebook, Skype, SnapChat, etc.). It is no longer reasonable to expect the instructor and TAs to be able to monitor student laptops during exams. To solve this issue, android-based tablets with special apps have been developed and implemented into two fundamental engineering courses, Fluid Mechanics and Solid Mechanics (Mechanics of Materials).

This paper first examines both the selection process for the Android-based tablets and the development process for applications (apps) required for testing. Both Android and iOS tablets were considered. Based on cost and ease of system customization, Android-based tablets were selected. Two different apps were developed for use for tests. The first app, "examApp" permits the student to access the online text and it records the student's choices. It has numerous security features that limit access to the exam only on the tablets provided and only during the test time. All graphics are vector-based so they are crisp on high resolution screens. Included in the examApp was an admin (or instructor) page that provides the instructor and TAs real-time information about the exam while in progress. This includes scores, percent correct for a given problem, students activity, graphs, and direct communications to all students. The second app, "eBookApp", allows the student access to the online eBook used for the class. However, it blocks the student from other online web sites or material. The Android operating system itself was modified to limit access to standard apps like email, app stores, social networks, and even browsers. The apps were developed using Adobe AIR development framework due to its relatively easy development of web and mobile apps when compared to native programming or HTML5 with JavaScript. AIR also avoids programming the same app three times (web browsers, Android and iOS). The successful results (and lessons learned) using tablets for in-class testing for multiple engineering courses are presented.

Introduction and Purpose

While class size for core engineering classes at many universities and colleges may not be as high as the infamous freshman physics or chemistry of 300 or more students, they are generally less than optimal and continue to increase as the demand for engineers' increase. Combining large class sizes with the strong emphasis on research (and the search for funding) results in the need to find ways to maximize the time available for teaching. One time saving method is to use standardized testing methods such as multiple choice exams. These types of exams are not perfect, but it does save time which can be devoted to more student-face time such as office hours or problem solving sessions. Grading papers or exams can take a large amount of time, and may not be the best use of time available for teaching. There are some positives with automatic grading such as fast reporting of test results, and unbiased grading. All instructors have had to spend time dealing with students arguing over more points. Besides, multiple choice questions are common for graduate entrance tests and even professional engineering license exams. It would serve students well if they learn how to answer multiple choice question type exams effectively and efficiently.

Next issue, if multiple choice exams are to be used, how is the best method to implement them? There are multiple ways to administer multiple choice exams, but there are two broad categories; paper-based 'bubble answer sheets' and computer delivered exams. Paper is hard to automate and requires special equipment. On the other hand, computer-based testing can be developed and delivered with student computers or lab computers. It is assumed that system utilizes either a local network server or a general web-server. Web-based method has been used successfully [1,2] for over the last decade by the author by requiring students to have laptops. However, the problem with laptops is they must be monitored by the instructor during the exam for communication software use and unauthorized web-sites. With over a hundred students in a typical large engineering class, this has become a formidable task, even with the help of teaching assistants. There are just too many sophisticated software communication programs and operating system components to truly track all possible ways to access outside help. To alleviate the instructor from monitoring screens and to keep viewing the screens by other students to a minimum, a tablet-based exam system was designed, developed and implemented at the University of Oklahoma. It is important to note, the tablets were supplied by the University for the exam period and was still administered in a normal classroom.

While there are commercial systems available, such as Canvas, for web-based and tablet-based testing, they are generally for large implementation and can be costly. There are also limitations on what type of problems and graphics can be integrated into their systems. This can be problematic for engineering where detailed diagrams are needed. Since the author already had a working system for web-based testing, including thousands of problems in electronic format, it was determined that it would be easier (and cheaper) to develop his own testing app. This is not a solution for everyone, but did work well for the author. This solution also permits future use of the system as a MOOC (Mass Open Online Course) platform since there will be no cost to users.

This paper presents the development of a tablet-based testing system for core engineering classes that involve problem solving, such as Statics, Dynamics, Solid Mechanics, Fluids Mechanics, Heat Transfer and Thermodynamics. The goal was to deliver a multiple choice exam to the students using tablets. The tablets would only be given for during the exam time, and used for general class activities. This required special application software (apps) to be developed and tablet OS modified. Both these tasks are outlined in this paper. The server system with a database of problems already existed from the previous laptop-based testing system.

Tablet OS and Model Choice

Before actual tablet app development could start, the actual tablet model and operating systems needed to be chosen. With the large number of tablets on the market, this was not a simple task. It was determined that the main constraints included 1) cost, 2) ease of programming, 3) limiting communication type apps, 4) and capability to install custom apps. When this project was started, Fall of 2013, there were two main types of tablets, Apple iOS and Google Android.

The capability to install custom apps and limit communication apps were critical issues in deciding what OS to use. Because Apple requires all apps to be distributed through the iTunes App Store web site, side-loading (installing apps directly from a local computer through a USB connection) of apps to the tablet is prohibited except for development. Also, Apple does not allow rooting (access to system files and apps) of the iPad tablet which prevents the removal of communication apps like email, chat messenger, Facebook, twitter, and iTunes (cannot allow student to download apps during exam). On the other hand, the Android operating system is considered to be an open system which allows the user to side-load any program (it does however, encourage users to install from Android marketplace website, like Google Play). Since the apps for testing would be unique for the author's students, it was determined that side-loading would be required. Also, most Android tablets can be easily rooted (user control of all system files) which allowed the removal of any apps or system feature that could allow students to gain outside assistance (i.e. email, chat, Bluetooth, NFC (near field communications), Facebook, etc.) For these reasons, Android system was chosen over the iOS. The programming ability of both systems was about equal. As a secondary issue, the higher cost of Apple iPad models was also a concern but not the deciding factor.

Within the Android system there are still hundreds of tablet models and dozens of manufactures and they each have their own unique specifications and cost. Initially, a short list of minimum level of specifications was established which are listed in Table 1. The table also gives a short reason why that specification is important. After reviewing over 10 different tablets in the Fall of 2013, two tablets, Hisense Sero Pro 7 and Dell Venue 7, meet or exceeded the required specifications. There were better tablets on the market, but their price generally ranged from $200 to $400 for a 7 or 8 in tablet. Since cost was important (education budgets are always limited), it was determined that a solid, mid-range tablet would be suitable and cost effective. It is expected (hoped) that these tablets will be usable for the next 4 to 5 years, or when their batteries no longer hold a charge. Removable batteries are rare for small tablets.

Both the Hisense and Dell tablet where nearly identical, and both were acceptable for the project. Dell's camera resolution was lower and the price higher. On the other hand Hisense had lower system RAM. The performance benchmarks of the CPU were comparable for both the Hisense CPU Tegra 3 quad core and the Dell Atom dual-core. In the end, the deciding factor was price, especially when refurbished Hisense units came available in late 2013 for less than $90. The refurbished units performed well on all tests and were substantially lower in price. After a semester of full use, all refurbished tablets are still working as expected.

While most specifications are self-evident, a few need qualifications. The external SD port is helpful in rapidly loading new systems into memory. After two or three uses, all systems are wiped and reloaded using seven micro-SD cards. This takes about two hours for 77 units. It would take over eight hours if done by side-loading. The Hisense also has a large following of hackers that develop special MODs (customize systems) that have proven helpful. GPS is currently not used, but for future distance learning, that can be a useful tool to identify location of the user.

Table 1: List of Tablet Specification (2013).

Feature or Spec	Reason	Hisense Sero Pro	Dell Venue
1 Meg Front Camera	Monitor student during exam	1 gig	0.3 gig
3 Meg Back Camera	Monitor room during exam	5 gig	3.1 gig
USB port	Standard power connector	yes	yes
External SD Slot	Load system, Load apps	yes	yes
4+ hour battery	Last 2 or more classes	7 hrs	8 hrs
Can be Rooted	Remove system apps	yes (easy)	yes (hard)
1280x800 resolution	Sharp, readable image	1280x800	1280x800
7 inch	portable, not viewable by others	7 in	7 in
Cost	< $150	$129 ($90 refurb)	$145
1 gig RAM	run multiple apps	1 gig	1 gig
8 gig storage	store basic apps	8 gig	16 gig
Dual core CPU	CPU time for simulations	Tegra3 Quad	Intel Atom Dual
GPS	locate student	yes	yes
HDMI	External Projector (not critical)	yes	no

Tablet App Development

Two separate apps were developed. First, the examApp program was the main app that generated the exam and provided the mechanism to submit their answers. The second app, eBookApp, was secondary, and provided a secure method for students to reference the course text. The two apps were independent, and were not connected.

The examApp was designed to access multiple-choice problems from an online database located on the class web site server (eCourses.ou.edu). The app does not set up the exam, just assembles the problems for the students based on the pre-set test questions chosen by the instructor through the course web site. In addition to presenting the problems, the app also checks date and time, verifies the user (student) name and password, and checks if tablet is authorized to access the exam. The exam is similar to previous web-based exam system used for the last 10 years for Statics, Dynamics, Fluid Dynamics and Solid Mechanics. [1,2] The exam screen, shown in Figure 1, allows the student to view all problems by scrolling up and down. The choices can be submitted multiple times, but only the last submission is graded.

The examApp was developed by the author using Adobe AIR programming framework [3]. This framework is an extension of Adobe Flash and Adobe Flex tools, and can be used without cost. There are also several free 3rd party editors like FlashDevelop [4] that simplifies the development process. The main advantage of AIR is its cross platform delivery capability. It can be compiled for Mac OS, iOS devices, Windows, Android devices, and Flash enabled web browsers. Thus, with limited resources, a single app can be delivered on all major platforms without any modifications. The app is installed on each tablet, but the problems are not stored on the local tablet. The app pulls the side scripts written in Microsoft C# language. The server OS is Windows 2008 R2 using MySQL database. The database also contains student information, such as user ID and passwords. Only authorized students of the class can log into and view the exam. As an added security feature, all tablets are registered with the database using their unique MAC address (serial number of the network card). Thus, when the examApp communicates with the server, it provides the server with its MAC id, and the server then checks if that MAC is registered. This prevents outside tablets from accessing the exam even if they have access to a student account.

Another security feature of the app is its ability to check if a student has already logged into the exam. The app uses a secure socket with the server for a persistent connection (uses RTMP communication protocol). Unlike normal HTTP connections, RTMP stays open and can monitor if multiple users are accessing the database with the same account. The system will stop any user from logging in an account more than once simultaneously. This has been a problem

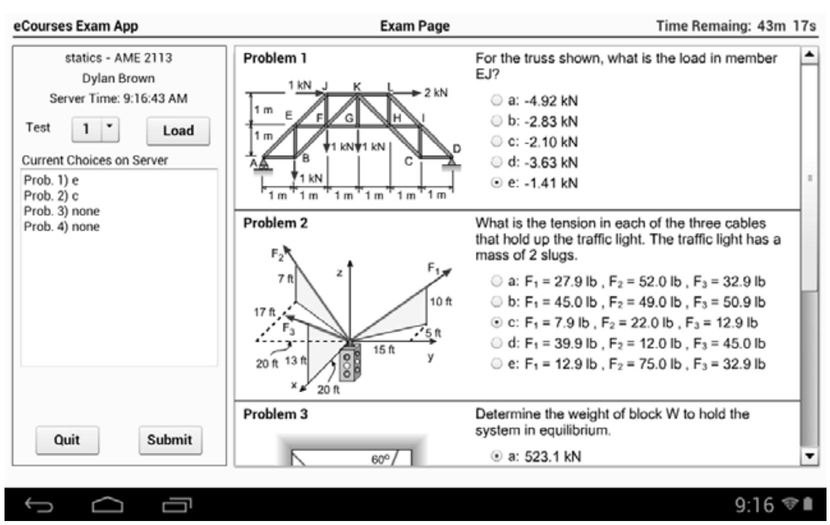

Figure 1: Student Exam Screen in examApp.

previously with web-based testing system where students share accounts and view each other's answers during an exam.

In addition to the main exam screen for students, there is an administrative or instructor screen for monitoring the exam in real time (Figure 2). This screen allows instructors and teaching assistants to view how each student is doing during and after the exam. The overall test distribution is given in graph form. A listing of all students in the class is also listed with their current score, the device number (tablet) and IP address. If a student name is clicked, then more information about the student is provided below the student listing. While currently not used, the app also can activate the student camera as they are taking the exam. Remember, these tablets are not students own and are strictly used only for the exam. Thus, there are no privacy expectations and the camera can be used to monitor the student during the exam to confirm the student is taking the exam. Future plans include using visual identification for distant or remote students. The instructor page also reports the percentage correct for each problem. This has proven helpful in identifying potential issues with a particular problem. If the percentage correct is low, then the problem can be quickly reviewed to make sure the problem is correct and there is not typo.

All text, graphics, and diagrams in the examApp are vector-based. This includes the problem diagram. This provides sharp and crisp images for viewing, especially in high resolution devices. The current device is 1280x800 resolution, which is higher than the actual designed resolution of 1050x595. The lower resolution was chosen to allow smaller devices (i.e. phones in the future) to view the exam clearly. Since vector graphics can scale to any resolution without loss, setting a lower design resolution has no negative effects. If new devices are used with higher resolution, the screens will not need to be redesigned since the vector graphics will scale to any resolution without distortion.

The second app that was developed, eBookApp, was to provide access to the online eBook [5] during the exam time. The web-based eBook was the primary learning material for the course and the author allows students to use all learning material during exams, i.e. they are open book - open notes exams. Initially, it was planned to use an open source browser such as Firefox to give students access to the eBook. After extensive research, it was determined that it was not possible to completely limit access to unauthorized web sites. There are many techniques to restrict web sites, but all have limitations, and in the end, clever users can

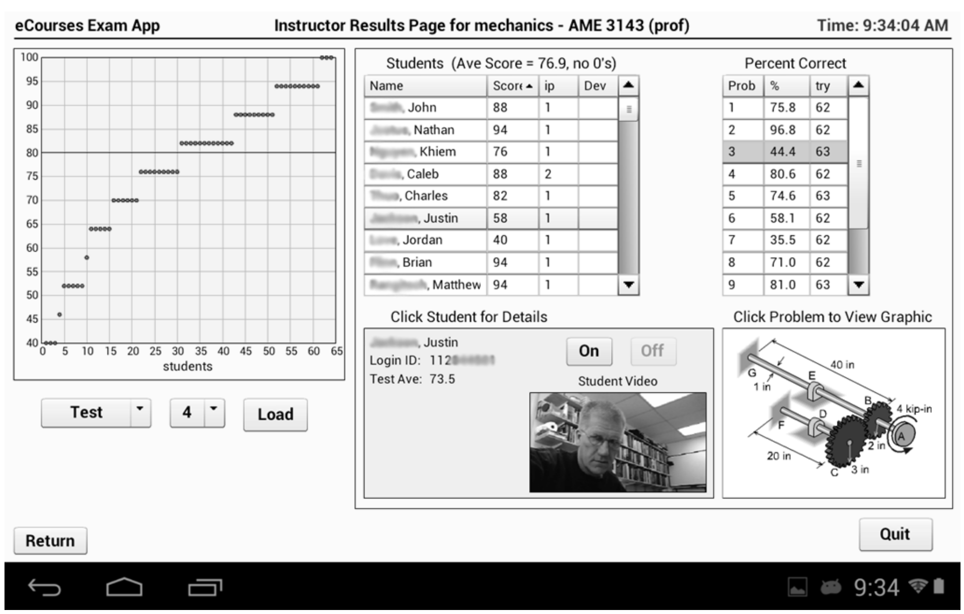

Figure 2: Instructor Screen in examApp.

usually defeat the restrictions. Thus, all browsers were removed from the tablet (see the next section) and another app, eBookApp, was developed to allow controlled access to course eBook.

The eBookApp, similar to examApp, was developed using Adobe AIR so that it could be used for other mobile devices in the future, not just Android (Figure 3). The app itself is simple since it just calls a web site, eCourses.ou.edu, and presents the web pages associated with the online eBook. Adobe AIR programming framework has a basic web browser that can be used in any app. This made it easy to display the web site as the only component in the app. There are no buttons or URL text fields to allow navigation. All eBook navigation is done within the web page itself. The lack of a back or forward button is not critical since the eBook has multiple menus to allow easy navigation to any page. Basically, eBookApp is a browser window that only calls one web site, eCourses.ou.edu. It was simple, but provides students with full access to the web site, and no specialization of the eBook for the tablet was needed. The only problem with this solution was that all links to other web sites from within the web site had to be removed. Students had access to all eBooks, not just the course eBook. This is helpful for referencing previous courses like Statics or Math.

Tablet System Modification

Exam security was a reason for switching from laptop-based exams to tablets, as mentioned previously. It has become increasingly more difficult to monitor student's laptop screens for rogue communication programs and unauthorized web sites. Furthermore, large laptop screens are easy to view, even 2-3 rows apart. University provided tablets helped solve these issues, but presented new security challenges due to the pre-installed system features and apps. For example, there are email and messaging apps that are standard on all stock tablets. Plus there are system level communication features like Bluetooth and NFC (near field communication). Both apps and system features need to be removed or disabled for exams so that students cannot communicate during the exam. This required that the tablets be rooted so that all system and app files could be accessed, modified or removed.

The Hisense Sero Pro 7 tablet was chosen in part for the ease in which it could be rooted [6]. Generally, manufactures of mobile devices discourage (and even try to prevent) rooting so that users will not accidently remove critical system files and 'brick' the device. Rooting the Hisense provided access to important communication applications that could not be removed otherwise. Disabling is not sufficient

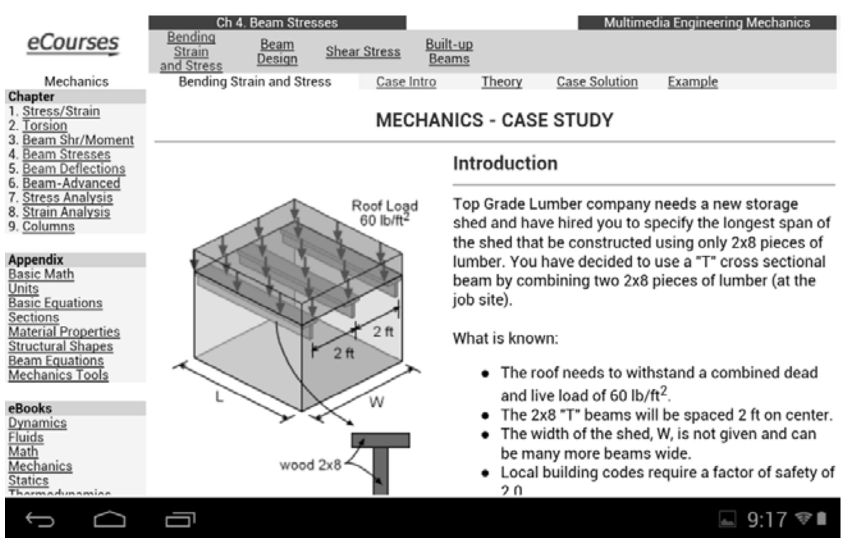

Figure 3: Instructor Screen in examApp.

since students could re-activate them during the exam. The key system apps that were removed are Google Gmail, Google Hangouts, Google+, Google Drive, Chrome Browser, Email, Search, and Play Store. Other minor pre-installed 3rd party apps were also removed.

In addition to apps, Android system has various communication methods that are a standard part of the system. Two critical ones are Bluetooth and NFC. These are not stand alone apps, but integrated into the system. However, they must be disabled at the system level or students could theoretically communicate peer-to-peer between tablets. The only way to permanently disable them is to use root access and remove critical files associated with those features. Not being an Android system expert, this process was done by trial and error. If a file was removed that was needed for other system functions, the tablet would exhibit strange problems. A previous working version of the full system would be re-installed and the process started again.

To minimize students from accessing system settings, a new system launcher was installed. A launcher is a 3rd party shell program that allows users to modify the system graphics, layout, and access to various system functions. While not perfect, launchers can help control what users can access. Figure 4 shows the modified system screen that removed easy access to the system settings and app drawer. Both can still be accessed, but there is not a screen settings button like on normal Android systems. Also, the top application bar was removed, providing more vertical space for the exam app. Only those apps that can be used during the exam were shown on the main screen. These included, examApp, eBookApp, flowHPC (2D flow application for drag), torsion HPC (torsional shear stress application), realCalc (commercial scientific calculator), and Wolfram Alpha (commercial numerical solver).

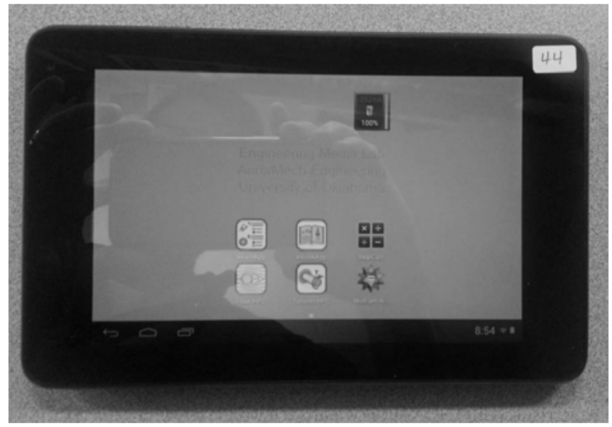

Figure 4: System Screen for Tablet.

This project did investigate the use of "kiosk" programs that allow the complete lockdown of mobile devices for use in public locations. Those programs could be used in place of rooting the device and removing files, but there is a limit to the customization. Also, they are relatively expensive at $30-50 license fee per device. Considering the cost of the tablet themselves was only $90, it was not realistic to spend up to 50% more for locking down the tablet.

Tablet Storage, Transportation, and Charging

An often overlooked issue with tablets is the mundane problems of long term storage, transportation, and charging. Initially, commercial charging stations and rack storage cabinets were investigated, but due to the high cost, they were not viable. In most cases, charging stations and rack systems were more than the total cost of the tablets. The alternative solution was to design and construct a custom system that addressed storage, transportation and charging issues.

To allow easy transportation, the storage system was designed to be fit on a standard lab utility cart (actually, the cart used was a 15 year old unused cart in the lab). As illustrated in Figure 5.1, individual wood racks are assembled on both the top and bottom shelf of the cart. The rack itself was built from standard 3/4 in shelf planks with 1/2 in square wood dowels screwed onto the plank (Figure 5.2). Each dowel was wrapped with duct tape to minimize damage to the tablet screen when placed in the rack. Velcro stripes were glued to the ends to secure the USB power supply that was used to charge the tablets.

Initially, it was planned to use the individual USB power supply that came with each tablet to charge the tablets, and mount them on the cart. But it was soon realized that it would take too much room, and would require too many AC power strips. Thus, a smaller USB power strip was used and attached at the side of each rack with Velcro. Each rack was designed for seven tablets. The power supply for each USB strip was plugged into a standard AC power strip at the back end of the cart. These were combined together and finally one power cord was used to connect to the wall AC outlet. One problem with most inexpensive USB power strips is their low amps output. They are limited to 3-4 amps which means each tablet will have only about 0.5 amps to charge. This is acceptable if the tablet is not being used while charging. But if the tablet must be used (screen on) while charging, then the tablet should have 1-2 amps to charge. The cart itself is designed for industrial use which makes it ideal for transporting the tablets from building to building. Both test classes that utilized the tablets in Fall 2014 where in a building about a half mile from the engineering labs.

Actual Use and Feedback

The examApp, eBookApp and modified tablets were used in two junior level basic engineering courses, Fluid Mechanics and Solid Mechanics, at the University of Oklahoma for the 2014 Fall semester. Each class had approximately 60 students. Both classes had three midterm exams, 75 min each, and a final exam, 120 min. The tablets were not used in the class except for exams. To help students become familiar with using the tablets for exams, a simple non-graded exam with two trivial questions was given one class period before the first exam. This also served as a beta test for the modified tablets and apps.

Generally, the tablets were distributed with the help of a teaching assistant 5-7 min before the start of the exam. They could log into the app, but the test would be not available until the start time. The time was based on the server time, not the local tablet time. When the students logged on, a persistent RMTP connection was established with the server. This allowed the server to monitor the student progress in real time and keep the local tablet set with the uniform time.

One of the reasons to use tablets is to help reduce desktop clutter with large laptops. As can be seen in Figure 6, the tablet did not take much room. The students still used their own calculators, but a full engineering calculator was also available on the tablet. For actual exams, paper copies were also available in case the network or tablet had technical problems. Of the eight exams, only one had any technical difficulties, and that was due to

Figure 5.1: Storage System Cart.

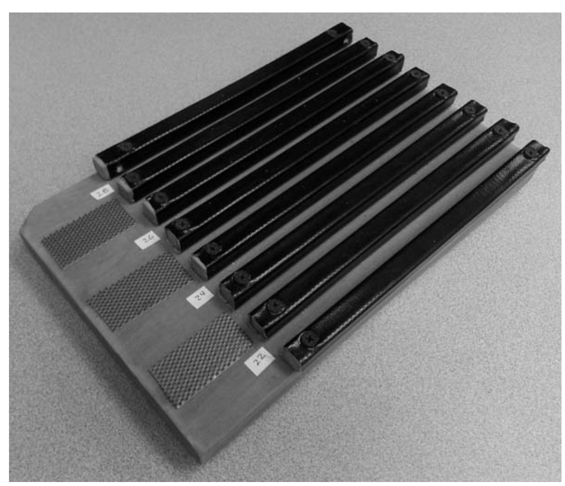

Figure 5.2: Single Storage Rack.

Android system losing access to its keyboard on some of the tablets. Later, this problem was tracked back to removing a critical system file. Interestingly, the problem took 3 weeks to become active and then only on a few tablets.

Figure 6: Students Using examApp During Exam.

Feedback from the students was neutral to positive. Interesting to note, many of the positive comments were related to strict exam control that prevented early viewing of the exam which is common when paper copies are distributed (first ones to get the exam see it 1-2 min before some students). Also, students like that everyone had to stop at the same time. It seems, many students dislike other students continuing to work even though the instructor states the exam is over. With the tablets, the submission time is controlled by the server, and all exams stop at the same time. There is a 1 minute grace period when the exam ends and before the system shuts out additional submissions. This solved the issue if there was a last minute network delay or problem.

Summary

A new tablet-based testing program was designed, developed and implemented for use in large enrollment engineering courses, such as Statics, Dynamics, Fluids, and Solid Mechanics. The main reason for developing the program was ongoing difficulty in monitoring student laptops with the previous web-based testing system. With the increase of social media, and availability of sophisticated (and hidden) communication programs, the security of the exam could no longer be reasonably assured. The new tablet-based system presented in this paper allows a number of added security features, such as a single login per account, real-time monitoring of student during exam, restricting which tablets can be used through MAC address registration, control of test app distribution, IP address access control, and even forced removal of a student from the test if needed. While these features do not guarantee the exam integrity, it does drastically reduce the opportunity for students to cheat electronically. The exam itself has added features to minimize cheating like random problem order and random answer order. It should be noted, this app was not currently designed for distance or remote testing. While that is a future possibility, it was assumed that the tablet would be used in a physical classroom with visual monitoring by the instructor and teaching assistants.

The development of the examApp involved a number of technical issues. First, the tablet itself had to be rootable so system level communication apps could be removed or disabled. This limited the operating system to Android devices. Next, the app itself had to be programmed to open a persistent connection with the server, pull the exam questions from the class server database, control device access, monitor student submissions, and provide the instructor an administration screen for monitoring students. Developing this app was the main effort in this project, and took approximately 6 months' time by the author. Adobe AIR programming framework was used for developing the app due to its flexibility to be compiled on multiple operating systems.

The examApp was successfully implemented in two courses by the author, Fluid Mechanics and Solid Mechanics for a total of eight exams (including two final exams). With the exception of one exam, there were no technical issues and was successfully used. The one exception was a problem with the tablet system losing access to the keyboard system app, thus limiting text input.

This was due to removing too many system files during the testing phase, and the problem did not surface until 3 weeks of constant use. Paper versions of the exam were used in the few cases where this error surfaced.

The students had no negative feedback on the use of tablets for exams. Of course, they do not like exams in general, but seemed to accept the use of tablets to deliver exams without concern. They did like the ability to use other course related apps on the tablet, such as a scientific calculator and the WolframAlpha app. They also had full access to the course eBook. As more courses use eBooks, the use of tablet access to an eBook may become an important feature. It is felt that the use of examApp was successful, and further development will be done. It is hoped that in the future, through the use of built-in cameras, the app can also be deployed for remote test taking. However, this will involve additional security issues that will need to be addressed.

Bibliography

1. Thukral, M. and Gramoll, K. (2002, June) "Web Portal for Basic Engineering Courses with a Problem Database and Integrated Communication," ASEE Annual Conf, Montreal, Quebec, Canada, 16-19 Jun 2002

2. Gramoll, K., Hines, W. and Kocak, M. (2005, June) "Delivery and Assessment of Teaching Statics over the Internet to Community College Students," ASEE Annual Conf. Proc., Portland, OR, 12-15 Jun 2005.

3. Adobe AIR, www.adobe.com/products/air.html, Feb. 1, 2015.

4 FlashDevelop, www.flashdevelop.org/, Feb. 1, 2015.

5. Gramoll, K.(2007, June) "A Web-based Electronic Book (eBook) for Solid Mechanics," ASEE Annual Conf. Proc., Honolulu, HI, June 25-27, 2007.

6. Hisense Sero Pro 7 Rooting, http://forum.xda-developers.com/showthread.php? t=2297246, Feb. 1, 2015.

Biographical Information

Kurt Gramoll is currently the Hughes Centennial Professor of Engineering at the University of Oklahoma. He has previously taught at the University of Memphis and Georgia Tech. He graduated from Virginia Tech with a Ph.D. in Engineering Science and Mechanics in 1988. His research includes development and implementation of educational technologies for engineering education and training that utilize simulations.