



LEARNING MODULES

GK-12 DISSECT at New Mexico State University

Title: AppInventor in a Forensics classroom

Author: Ashley Michalenko

Discipline or Area: App creation

Teacher: Beth Rewalt

School: Centennial High School

Subject of class: Forensic Science

Grade: 11/12

COVERAGE OF COMPUTATIONAL TOPICS

This module covers algorithms, iteration, branching, clarity, and correctness. This module incorporates all topics taught thus far in the classroom.

OBJECTIVES

With this module students will learn how to create a simple Android app using AppInventor. They are expected to use the forensic science knowledge obtained in the classroom thus far to create an app that would be useful to a forensic scientist. They will also learn how to use the online program AppInventor and through that build their computational thinking skills.

EQUIPMENT AND MATERIALS

What equipment and materials are needed for this module?

Computers that have a compatible browser for AppInventor. AppInventor is compatible with Chrome, Safari, and Firefox.

BACKGROUND AND REFERENCES

The purpose of this module is to teach students how to use the application AppInventor. While learning how to use AppInventor students will learn and practice iteration and branching. As they create their apps they will learn how to make their apps clear and correct/minimize bugs in code. In general they will be practicing algorithm development as they build their app.

Students will chose a topic that they have learned in class such as fingerprint identification or blood scattering patterns and design an app that incorporates information on the topic.

AppInventor is an online programming tool that can be accessed at <http://appinventor.mit.edu/explore/>. Students will need to create a Gmail account prior to use.

PROCEDURE

The first class served as an introduction to AppInventor. I briefly went over the application on the teacher's computer and as a class we watched tutorials for a few beginner apps. These tutorials are available on the AppInventor website.

In the second class I had students create a Gmail account if they did not already have one and signed in to use the program. I walked them through one of the tutorials we had watched the previous week, the Ball Bounce app, and by the end of class everyone had the app created.

The next several classes I let the students explore with the program and design their apps. I was available for any help they needed. However, I wanted them to explore AppInventor freely so that they could better learn the program. When I noticed many students asking a similar question I would make announcements to the class with information pertaining to the topic.

During one of the classes I also had students draw out their app. At the time our computers were running slow so not much progress was being made and I wanted to keep the students engaged. Each student had to draw pictures of what each screen in their app looked like. Then if the screen had buttons I had them write down what actions needed to happen after the button was clicked. This was to help them more when coding actions in AppInventor. Because we had issues with our computers this module was not completed and students did not have complete apps by the end of the semester. If they had we would have spent a class doing demonstrations of the apps.

The main learning goal of this module was to teach algorithm development. Through the algorithm development they would have been introduced to all CT concepts. They used branching and iteration while coding and correctness and clarity in debugging.

CT was introduced through the module. I tried to point out terms when they were being used by the students and half way through the semester did a vocabulary review with examples pertaining to AppInventor. I showed screen shots of when branching and iteration were used and talked about being clear, correct, and efficient in coding.

NOTES AND OBSERVATIONS

Students liked the idea of creating apps. The only obstacle was our computers. In a classroom with fast and reliable computers they modules would have been extremely successful. When students had a good computer and were able to make progress they really enjoyed the work. However, they became very discouraged when they were slow.

The best day was when I had students draw out a plan for their apps. All students ended the day with a plan and were very engaged. While talking to students throughout the semester I could also see them picking up on CT concepts and recognizing the terms being applied to the app.