Unblocking the pipeline by providing a compelling computing experience in secondary schools: are the teachers ready?

Catherine Lang, Annemieke Craig and Gail Casey
La Trobe University, Melbourne; Deakin University, Geelong
Victoria, Australia.
c.lang@latrobe.edu.au; acraig@deakin.edu.au; g.casey@deakin.edu.au

Abstract

The decline of student interest and participation in computing degrees at university is affecting the stability of computing as a stand-alone discipline in universities. Research indicates that the decline begins in secondary schools. This paper describes an outreach program that was funded by an Australian Council of Deans of ICT grant. The researchers, acknowledging the time-poor nature of teachers' work and that some of them are not trained in the computing discipline, developed curricula and provided resources and student helpers to enable secondary school teachers to deliver a student centred unit of work. This unit focused on a four week program with students developing applications for android phones. The program was delivered in three schools by four teachers and produced mixed evaluation results. In one school the number of students taking ICT the following year increased significantly, this was not reported in the other two schools. Our findings show that even when teachers are provided with resources and artefacts, not all are prepared to deliver a fully student-led classroom experience. We ask "are the teachers ready?" to embrace transformational pedagogies using ICT in the classroom. In this case study we can say some are, but some are not. We also note that the technical issues within school networks hamper the ability of teachers to provide compelling computing experiences to students. Our recommendation for future implementations of the program is to provide teachers with more background on the benefits of a student-centred classroom approach before beginning this four-week unit of work.

Keywords: Broadening participation, Diversity, Computing Education, Outreach, transformational pedagogies.

Introduction

Most computing educators are aware of the downturn of student enrolment numbers in both higher education computing courses and senior secondary school courses in Australia. For example, the Australian Computer Society report that less than 3% of all university students are studying undergraduate computing degrees (ACS 2012). This is not an issue limited to Australia but reported in other westernised nations such as the USA and UK (Durando, Wastiau and Joyce 2009, Mentornet 2012). In the state where the program described in this paper was carried out the decline in the number of students studying the computing discipline in senior secondary school is alarming. The total number of students studying final year units has decreased by 75% since 2001, and the lack of gender diversity within this cohort is equally dramatic, for example in 2001 there were 5879 female students enrolled in the final year computing exams, and in 2011 only 643 (VCAA 2011). The trend of declining enrolments is similar in student selection of university computing courses, a decline of 67% in the same time period (VTAC 2013). These statistics are indicative of a decline in student perception of ICT courses being a valid and relevant component of a future career (Lang 2012).

Research has shown that a positive and engaging experience with ICT in the classroom can spark student interest and desire to pursue this discipline (Lang 2010; Fisher, Lang, Craig and Forgasz 2012) and that engaged and enthused teachers are critical to the running of a successful program (Guzdial and Ericson 2012; Ericson 2011, Thurairasa and Lang 2013). However, an important finding from this research is that teachers in general are keen to participate in professional development to keep their classes relevant and their students engaged, but they are often time poor and lack the relevant physical equipment to deliver innovative curriculum (Thurairasa and Lang 2013). This literature informed the design and delivery of the outreach program that is the focus of this paper. In particular our program was designed to relieve the pressure of time-poor teachers in that the curricula was written to encourage a student-led peer support classroom model of delivery. This allowed teachers to be relieved of the pressure to be the computing expert. Secondly we provided the physical equipment needed, a class set of android phones and a curricula that allowed students to create of a meaningful product – a series of mobile phone games adapted from a popular textbook (Wolber, Abelson, Spertus and Looney 2011). The third unique aspect that contributes to building strong pathways between students in secondary schools and those in universities was to provide each class with a student facilitator, a current undergraduate student, who could assist the teacher in delivery of the program and also act as an informal role model to the school students. Furthermore, we purposely recruited female students to be our facilitators to debunk the myth that computing is a boys only subject area.

In the following sections we provide further information on the background to the study, a description of and justification for our method of delivery and curriculum development. This is followed by the findings from four instantiations of the outreach activity. The paper concludes with a discussion about the impact and future direction of future outreach activities, as well as recommended modifications to our model to ensure that future implementations avoid the same pitfalls.

1 Background

Technology has become more pervasive in every part of life in the 21st century and computing qualifications offer students dynamic career opportunities to work in any sector. Ironically as stated in the introduction student enrolments in these courses are in decline. Prior research into factors that influence student course choices (Lang 2012) and experience with outreach programs to address gender diversity (Lang, Craig, Fisher and Forgasz 2010) emphasised the importance of enthusiastic teachers to student course choices. Australia's secondary education curriculum is currently undergoing a review and it is suggested that this will equip students with relevant computing skills and knowledge, offer a compelling learning experience that will inspire them to pursue ICT both at university and as a career. Enthused and confident ICT teachers will be a necessary component to deliver the new curriculum (Tate 2012).

In the USA, researchers at the College of Computing at Georgia Tech have created an extensive outreach program that offered particular synergies with our program because it focuses on improving the quality and quantity of secondary school computing teachers. This program reported quantifiable positive results, with the number of students in Georgia taking Advanced Placement tests in computing steadily on the rise since 2008 (Guzdial and Ericson 2012). The Georgia Tech program introduces teachers to "App Inventor" (Ericson and McKlin 2012; Siraj, Kosa, and Olmstead 2012), a block programming interface that "is a visual drag-and-drop tool for building mobile apps on the Android platform" (Wolber, Abelson, Spertus and Looney 2011 p xv). The attractiveness of App Inventor is that it interacts readily with Android smart-phones, allowing students the excitement and pleasure of creating an app that they can use on their own device. In doing so, this provides the tools for a constructivist approach to learning that is both active and tangible as well as aligning the activity with a real world task (Radloff, 2005).

Coupled with the positive finding from the US, we chose to use App Inventor as the basis for our outreach activities because it is based in the cloud (hosted by MIT) therefore eliminating many technical issues. There are extensive online resources, lesson plans and videos to support implementation of the activities. With reference to these existing resources, we could see how different schools have embedded this into their course structure to be able to gain student engagement as well as student interest within the computing field.

Our next step involved adapting an international program with lessons developed for a different learning environment (Wolber et al 2011) to the Australian secondary school curriculum with a particular emphasis on a student-led peer supported pedagogies.

2 Student-Centred Peer Supported Pedagogies

Students learn a lot from their peers (Nuthall 2007) and they come into the classroom with knowledge from their life experiences (Gonzales, Moll and Amanti 2005). These concepts remained at the forefront of our thinking as we designed suitable curricula to integrate android phones into our secondary school lesson plans. We wanted to embrace opportunities for collaborative engagements and reciprocal learning (Gammon and White 2011) that could also open up a new culture of learning where self-organisation and a more social approach to learning could lead to concepts of emergence (Nichols 2012).

"You can't teach it to me, though I can still learn it", these are the words from Thomas and Brown (2011 p.77) as they explain how tacit knowledge grows through personal experience and experimentation. In designing our curricula, we wanted to capture this tacit knowledge and ensure that students had opportunities to experiment and learn through play, sharing and having fun. Rhine and Bailey (2011) discuss the natural tendency of students' attention to wander over time and they argue that "focused distractions" (p. 303) can have a positive effect on learning. The App Inventor along with the android phones brought with them many possibilities for students to be distracted. However, such distractions support our desire for students to experiment and we could perceive that this would add value to the learning experiences for students.

Many types of learning managements systems now exist in educational institutions both in the Cloud and within school Intranets. These can often incorporate a range of interactive elements such as blogs, groups and discussion forums that allow students to interact, publish and share ideas. Such elements can be used to provide a more personalised approach to learning where relationships can be fluid but supported through peer-to-peer interaction and can require students to justify and clarify their understandings (Casey 2013).

This literature informed the development of the student-centred peer supported pedagogy to provide students with a compelling ICT experience to spark their future interest in computing courses and careers.

3 Method

The overarching aims of our outreach program were to:

1. Deliver a compelling computing experience to secondary school students via the creation of a meaningful product and seeing it through to completion (e.g. a game for android phones).
2. Provide time-poor teachers with support materials and necessary artefacts to deliver the program (student-led curriculum, mobile phones, and support in setting up the program).

3. Provide intentional role modelling by placing undergraduate students in the classroom to work with the teacher to deliver the program. We deliberately recruited female students to promote diversity.

In doing so the program addressed the issues of general non-participation of students in computing and lack of diversity of the cohort. The second issue addressed was the school teacher's lack of time to prepare new materials and lack of access to up to date curricula and resources.

Funded by an Australian Council of Deans of ICT Engagement Grant (2012) we were able to finance our research and purchase the necessary class sets of mobile phones. After gaining ethics approval from our institution we invited school teachers to a consultation workshop to determine their acceptance of what we believed was a meaningful activity that would align with the current school curricula. The outcome of this session was the brief to develop new course materials that allowed for the student-led classroom environment, and that was extensible to allow it to fit in as one module of four weeks duration, with an average of 3 classes each week that was suitable for delivery to years 9, 10 or 11 students.

This resulted in a module of work that could be delivered over 4 weeks depending on the timetabled classes (provided in Appendix 1). We knew that the schools had an average of 3 lessons a week for computing electives, with each lesson being typically 40 to 50 minutes duration. The module of work was focused on fostering creativity using the web based application hosted by MIT. It provided teachers with preparation instructions such as to install and pre-test the software as recommended on the MIT website, create an online sharing space for the project or use emails and set up the necessary Google mail accounts. The module also suggested assessment areas linked to the required learning areas of Visualising Thinking, Creating for Communicating and Design, Creativity and Technology. Lessons were structured around discussion, student-centred sharing for tips and hints, written tasks for peer and self-assessment, and reflection. Worksheets were provided, a list of online resources, helpful websites and a reminded that the activities were written to be student centred and that the teacher was not expected to have a complete understanding of all the tutorials and skills, acknowledging their time-poor situation. Week 1 activities were built around the theme of “Exploring”, Week 2 was “Learning by sharing”, Week 3 was “Creating” and the final week focused on showcasing student work. Assessment tasks were constructed to reflect the student-centred approach and the scaffolded learning. Tasks 1 and 2 were focused on the students' ability to help others, Task 3 on Knowledge building and Task 4 was collaborated group work and sharing with a lower grade level (we suggested Year 7 students).

The program was delivered in the second half of 2012 in three different schools, one of which repeated the program in semester 1 2013, so the evaluation is from four instantiations. There were also four teachers involved with the delivery because the school that delivered it twice had a different teacher for the class in 2013.

The timeline of the project was from June 2012 through to May 2013. Tools were also developed for collection of data and evaluation. Student helpers were employed and instructed to write a weekly blog at the end of each lesson they attended. We developed a teacher pre-survey and conducted post program interviews with the teachers. We were particularly interested in determining how they implemented the program and what issues they encountered. Given the short time-frame of the research grant it was not feasible to get parental permission to conduct student surveys. Teachers reflections and their reporting on student acceptance of the program was deemed to be sufficient in this instance, coupled with the observations of the student helpers, we believed we could obtain a satisfactory assessment of the effectiveness of the outreach activity.

4 Findings

The teacher interviews and student helper blogs provide a good insight into the effectiveness of this program. We provide a summary of these in the following sections.

4.1 Teacher Feedback

School	Characterisation
A	Male teacher. Independent School. High level Year 10 Multimedia Elective (teacher reported). Six apps used. 18 students (6 females)
B	Male teacher. Government School. Low level Year 11 class (teacher reported behavioural and struggling ESL students). Three apps used. 20 students (4 females)
B2	Male2 teacher. Teacher was a replacement for female teacher who started the program and had delivered 2 apps. Year 10 elective. 18 students (3 females)
C	Male teacher. Government School. Year 10 Game Design elective. 25 students (3 females)

Table 1: Summary of each class

In each of the follow up interviews it was obvious that the success of the program was tied very closely to whether the teacher followed the modules as they were designed. However, in each case the teacher said that they would use the unit in the following year. The two teachers who used the program as designed (B and C) were the most positive in their feedback. Teacher A did not use the mobile phones provided, admittedly there was a glitch in delivery and they did not arrive until mid-way through week 1 of the course. He also did not feel empowered to direct the university helpers to conduct any specific help or activity, despite having met them before the start of the unit:

Uni students walked around the class once or twice per week. They were mainly viewers. Did help some girls but were not used to being in the classroom. Probably needed more structure as to who and how to target. (Teacher A)

Teacher B2 also did not use the phones initially because he was afraid that students would break them. Teachers

A, B and B2 reported technical difficulties, and slow internet connections:

School system was a bit slow on the Internet - google problem. This was frustrating. Needs to be on the school servers definitely. (Teacher A)

One obstacle - internet in the room failing frequently. Internet based software was an issue. For bits of periods and sometimes it was slow. (Teacher B2)

Teacher B saw it is an advantage that students had to work through problem solving their own laptops:

The students can actually produce an App that they can put on their phone. A real link between what you are doing and what happens in the real world. (Teacher B)

This teacher was the most enthusiastic about the program and reported that he delivered a professional development overview to his colleagues in an after school meeting.

Fantastic course outline. I could pick it up with four weeks planned… All I had to worry about was getting familiar with the programming.

Teacher B commented that he did not usually do group work and found this a useful challenge. He structured it by allocating two minutes for each student to talk while he did the timing explaining that he needed to do this because many of his class were reluctant to talk due to lack of confidence and familiarity with computers. He believed that this worked well with the start and stop guidance. The teacher linked this to Quality Assurance in big companies saying to the students that in the real world they need to communicate their work – this gave them purpose for the group work.

I am not very good at getting students that are reluctant to share coherently their experience when they do something new. I think I can do this better next time. I would like to coax them into this more to develop these skills.

Students kept the same groups throughout the program with the girls working together. Next time the teacher said he would do this a bit differently, intentionally mixing the students more. Teacher B was very positive about the extra help provided in the classroom by the university students.

One student wanted to extend 'Hello Purr' as a slideshow and they (student helpers) didn't stop him and worked with him on the conditional logic. They provided good 1-1 support as well as being able to present up the front. It was good that the University girls were only slightly older, this helped the students relate to them. [Teacher B)

The intentional role-modelling was commented on by one other teacher, he stated:

[The female students] in particular were more hesitant in grabbing a phone. One had really low self-esteem – M [the Uni student] was great with her. I asked her if it was better for M to help or me...she replied M because she is a girl. The first girl coming into the group ended up doing more than some of the boys. She was hesitant initially. None of the girls wanted to be in the computer class and didn't want to be with a room full of boys - "they are noisy and smell bad" [Teacher C]

While it is difficult to measure success in a small trial, one teacher reported that word of mouth seemed to occur in the playground and the recent course selections for 2013 indicated that student numbers for Year 10 IT next year would clearly increase, possibly three classes.

This is a very pleasing increase because there has been a spiral downward in numbers over the years".

4.2 Student Helper Feedback

The student helpers kept weekly blogs on how each class went. Each of them was a volunteer who was studying an ICT degree program at university. They were given a copy of the text book and also the student curriculum as well as an android phone to practice on before going in to the classroom. They were rewarded at the end of the program with a book voucher and reference letter to thank them for their participation. In each case the student reflection provided another level of feedback to us on how well the students engaged with the curriculum.

The perception of Teacher A that the students were mainly viewers was not consistent with blog reflections:

Week 1: "I was helping students add sounds to their media pallet…I had to explain that all the coding happens within the blocks editor"

Week 2: I spent most of my time helping one of the girls in the class"

Week 3: The questions became more complex which required E and I to search for the answer on the internet… There were some questions that were more about how to develop a function; for example adding a score function to their game.

The weekly reflections show the student's learning, and also highlight the intentional role-modelling of normalising that girls can and do understand programming.

The student who worked with teacher B2 added insight as to why the implementation was less than successful:

Class 1: [the teacher] hadn't brought any of the phones to the class for the students to use so they were using the emulator. This couldn't test most of the extension work as a few features are unavailable as it's just an emulator.

Class 2 [the teacher] still hadn't figured out which task sheet the students were up to. [the teacher] had brought the phones to the class but had not unpacked the new ones.

Class 3 [the teacher] hadn't photocopied the next task sheet for the students as most of them had finished the week 3 tasks.

The teacher and student helper feedback indicate that when the curricula was fully embraced by an enthusiastic teacher, the desired outcome of students having a compelling ICT experience was achieved. However not all factors were within the power of the researchers or even the student facilitators.

5 Concluding Remarks

A student led peer support model curriculum was created, but we now ask "are teachers ready?". It would appear that in this case two embraced the opportunity to allow their students to explore the program and share with each other, a third was quite entrenched in his own way of program delivery and on a tight time-frame so limited the implementation of the curricula and the fourth appeared to have other issues to contend with, such as classroom support and preparation time. It should be noted that the teacher B2 was a late replacement to the program and had not attended the initial teacher briefing workshop. We have no control over changes to staffing in schools, and while the initial teacher (B) was still involved in IT education in that school, it appears that teacher B2 had limited internal support.

The model for student led learning using artefacts (mobile phone) and drag and drop programming interfaces was generally positively embraced with students and our aim to build teacher technical efficacy and promote student – teacher learning partnerships was achieved in two of the four instance. We know that all the schools are continuing to run the program as part of their IT curriculum, and that one school has purchased its own class set of Android phones.

Our classroom facilitators were provided to encourage student to student interaction and take the pressure off the teacher somewhat. As can be seen by their comments they embraced this role. While one teacher observed that they were not utilised, the students reported that they were indeed helping individual students.

The use of the AppInventor tool to spark interest in programming appears to have achieved a positive outcome. This grant and outreach program has acted as a springboard to ongoing research opportunities. We have developed a workshop program for school teachers that is being delivered in an intensive mode to twenty teachers. The opportunity to share the findings from this first run of the program will alert them to some of the pitfalls experienced, e.g. the technical set up within their own school. The importance of teachers to take up the program and allow students to explore together in the classroom is integral to a successful outcome. Teachers need to embrace different pedagogies to allow students to explore as they learn.

A limitation of this paper is that it is based on only four implementations of the program. However it delivers a model, curriculum and structure that can provide greater school university interaction to promote the creativity and knowledge building of programming to middle school students via enthusiastic and competent teachers.

6 References

ACS (2012): 2012 Australian ICT Statistical Compendium. Australian Computing Society. CIIER. Melbourne, Australia: 25-26.

Casey, G. (2013): From Telling the World to Showing the World in an Era of Social Media. In . Jan Herrington et al. (Eds.), *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2013* (pp. 1853-1862). Chesapeake, VA: AACE. Retrieved October 10, 2013 from http://www.editlib.org/p/112222.

Durando, M., P. Wastiau and A Joyce. (2009): Women in IT: The European situation and the role of public private partnerships in promoting greater participation of young women in technology. European Schoolnet. Brussels, Belgium. http://www.eun.org/. Accessed 26 November 2012.

Ericson, B. (2011): Attracting K-12 students to computing, Webinar delivered via NCWIT and ISTE. http://www.ode.state.or.us/teachlearn/pte/cte_updates_webinar-ncwit.doc. Accessed 15 November 2011.

Ericson, B. and T. McKlin (2012): Sustainable and Effective Computing Summer Camps. *SIGCSE'12 Proceedings of the 43rd ACM Technical Symposium on Computer Science Education.* R. McCauley, L. Smith, D. Musicant, T. Camp and P. Tyman. Raleigh, North Carolina, ACM.

Fisher, J., C. Lang, H. Forgasz and A. Craig (2012): *Digital Divas: Designing approaches to enthuse girls' interest in ICT studies and ICT careers*. http://digitaldivasclub.org: 1-30. Accessed 5th July 2013

Gammon, M. and J. White (2011): (Social) Media Literacy: Challenges and Opportunites for Higher Education. In: WANKEL, C. (ed.) *Educating Educators with Social Media*. Bingley, UK: Emerald Group Pub.

Gonzalez, N., L. C. Moll and C. Amanti, (2005): *Funds of knowledge : theorizing practices in households, communities, and classrooms*, Mahwah, N.J., L. Erlbaum Associates.

Guzdial, M. and B. Ericson (2012): Georgia Computes! An Alliance to Broaden Participation across the State of Georgia. *Inroads.* NY, USA, ACM. **3**: 86-89

Lang, C. (2010). Happenstance and Compromise: a gendered analysis of students computing degree course selection. *Computer Science Education* 20(**4**): 317-345.

Lang, C., A. Craig, J. Fisher and H. Forgasz (2010): Creating Digital Divas – Scaffolding Perception Change Through Secondary School and University Alliances *The 15th Annual Conference on Innovation and Technology in Computer Science Education*, ITiCSE 2010. Bilkent, Ankara Turkey. ACM:38-42.

Lang, C. (2012): Sequential Attrition of Secondary School Student Interest in IT Courses and Careers. *Information Technology and People* 25**(3**): 281-299.

Nichols, A. (2012): Blogging across the curriculum: An action research project. *Journal of Educational Multimedia and Hypermedia*, **21**, 165-174.

NuthallL, G. (2007): *The hidden lives of learners*, Wellington, N.Z, New Zealand Council for Educational Research.

Mentornet (2012): Number of Women Awarded Computer Science Degrees Has Declined Since 1984 http://www.mentornet.com Accessed August 2012.

Radloff, A. (2005): Supporting student learning in the 21st Century university: What's the job and whose job is it? *ATLAANZ 2005 Conference*, Dunedin, New Zealand.

Siraj, A.M. Kosa and S. Olmstead (2012): Weaving a tapestry: creating a satellite workshop to support HS CS teachers in attracting and engaging students. *Proc 43rd ACM SIGCSE Technical Symposium on Computer Science Education*. Raleigh, NC, USA, ACM Press.: 493-497.

Rhine, S. and M. Bailey (2011): Enhancing in-class participation in a Web 2.0 world. In: WANKEL, C. (ed.) *Educating educators with social media*. Bingley, UK: Emerald Group Pub.

Tate, N. (2012): Let's learn from the British ICT overhaul. *The Australian*. Sydney News.com. January 31, 2012.

Thurairasa, S. and C. Lang (2013): The Ripple Effect: An Outreach Project for High School Teachers Using a Cell Phone Programming Tool. *Society for Information Technology & Teacher Education International Conference*. R. M. M. Searson. New Orleans, Louisiana, United States, AACE.

VCAA (2011): Section 2: Satisfactory Completion of VCE units: http://www.vcaa.vic.edu.au/vcaa/vce/statistics/2010/statssect2.html. Accessed April 8, 2011.

VTAC, (2013): Analysis of Applications, Offers, Enrolments and Deferments by School Type 2002-2003. SectionD_10_11.pdf. http://www.vtac.edu.au/pdf/stats/2010-2011/SectionD_10_11.pdf. Accessed 5th July 2013

Wolber, D., H. Abelson, E. Spertus and L. Looney (2011): *App Inventor: Create your own Android Apps.* Sebastopol, California,, O'Reilly Media Inc.

Appendix 1:

'Create Your Own Apps' – Teacher Notes and Unit plan - App Inventor, 4 week lesson plan (Written by Gail Casey, version 5th Sept 2013 – gcasey@deakin.edu.au)

Unit Overview – This is a flexible unit of work which takes a student centred approach to learning.

Teachers should use this document as a guide only. The Wolber App Inventor text is online at http://www.appinventor.org/projects

Unit Title: 'Create your own Apps'
Year Level: **Year 7 to 11 - approximately 4 x 50 min**

Unit Summary:

This four week unit of work will help students understand the way in which mobile phones operate through the use of Apps. Students will research the online programming software 'App Inventor' before writing their own Apps using an Android mobile phone (or simulator). App Inventor fosters creativity through technology and is programmed through an Internet browser where students can design their own mobile phone apps. The program uses a series of blocks, like pieces of a puzzle, where students build a series of behaviours that when put together can appear live on an android phone.

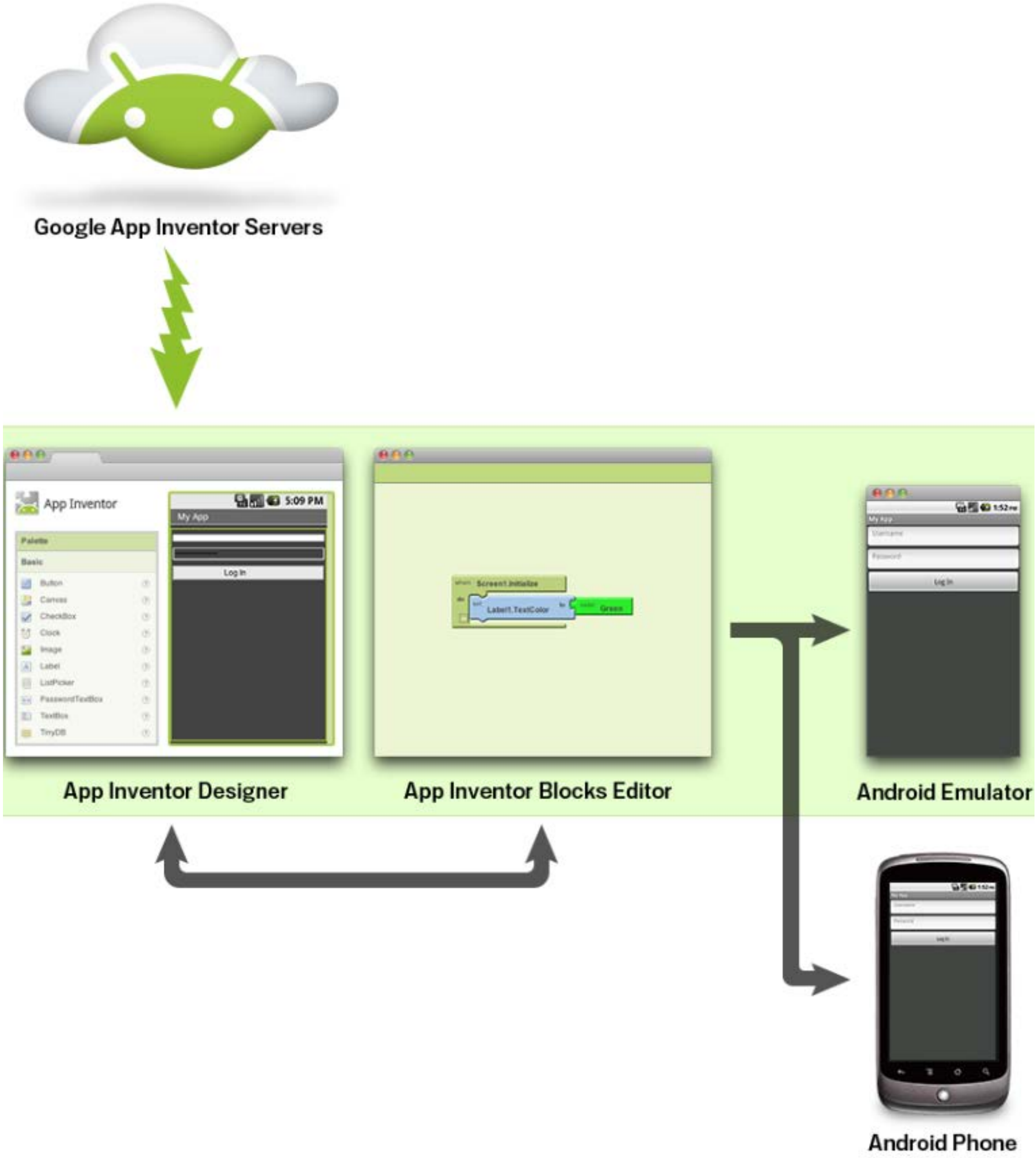

Figure 1: What is App Inventor | Explore MIT App Inventor

http://explore.appinventor.mit.edu/content/what-app-inventor
Screen clipping taken: 15/07/2012, 8:06 AM

Things to note:

1. You need to be on the Internet to run App Inventor. It is a Web based application and runs by browsing to the App Inventor website, http://appinventor.mit.edu.
2. More resources for each of the tutorials can be found at http://www.appinventor.org/projects
3. The App Inventor Setup Installer software is something you need to install beforehand so that your computer can talk with the android phone or in the android emulator.
4. **Many video tutorials have been created by David Wolber and links to his YouTube channel is given at http://www.appinventor.org/projects after clicking on the appropriate chapter, but for High school students, video tutorials made by Chris Groff should also be considered. Chris Groff's YouTube channel can be found at http://www.youtube.com/user/cgroff17**
5. Some tasks involve students taking **screen clips** of websites to share with their peers. These could be done using the **'Prt Sc'** key on the keyboard or using a screen capture software such as 'Jing' (See, http://www.techsmith.com/jing.html)
6. It is assumed that **good Internet** access is available.
7. **Don't** use **spaces** when naming files

Possible areas for assessment could include:

1. Skills - ability of students to learn, design and create using App Inventor
2. Communication – ability to share one's knowledge & communicate through the software.
3. Research & Investigation projects.

Please ensure that:

- Teacher and Students all have a Google account
- Java and the App Inventor software is loaded and runs on the computers
- You have backup activities for the classroom if the school does not have good internet access

Some suggestions in this unit plan:

- This four week unit is student-centred and works well if the teacher knows very little about App Inventor (or appears as such).
- Start lessons with a student-centred activity such as a student showcase, pair and share or a tips and hints session.
- Peer-to-peer learning is valued; hence, access to a shared space where program files and resources can be shared is helpful and supportive for collaboration. This could include an online drive, dropbox, wiki, blog or even school network drive.

Helpful Websites, other than http://ictplus.ning.com/:

- The main App Inventor site is at http://www.appinventor.mit.edu/ - click on 'Invent' to start programming.
- There are many helpful videos available on YouTube (See, http://www.youtube.com/watch?feature=player_embedded&v=VTbyqDCK3A0). Many of these can be downloaded prior to class for students to access from their school Intranet. Some are also available from Vimeo (See, https://vimeo.com/search?q=App+Inventor)
- Useful resource
  - Getting Started with Android App Inventor is http://www.i-programmer.info/programming/mobile/1789-getting-started-with-android-app-inventor.html, but be careful because it does refer to the old App Inventor site.
  - **Many video tutorials have been created by David Wolber and links to his YouTube channel is given at http://www.appinventor.org/projects after clicking on the appropriate chapter, but for High school students, video tutorials made by Chris Groff should also be considered. Chris Groff's YouTube channel can be found at http://www.youtube.com/user/cgroff17**
  - http://en.wikipedia.org/wiki/Google_App_Inventor, http://www.appinventorblocks.com/ , http://www.appinventor.org/
  - Other video tutorials also available at http://www.youtube.com/playlist?list=PLCF2969C390CE87F4
  - Other resources, http://www.appinventor.org/course-in-a-box

Create Your Own App: Programming with App Inventor Lesson Plan (4 Weeks)

Week 1 – what is App Inventor & 'Hello Purr' (Task 1 & 2)

Introduce the topic - the teacher instigates a class discussion by asking questions about mobile phones and mobile phone apps. This is to tease out what students know. The teacher should not have the answers and should not tell them what App Inventor is or where to find resources.

Give students the Week 1 handout - this is their instruction sheet and provides students with a check list for the first two tasks.

Task 1 – What is App Inventor and why use it?

This task encourages students to explore and find out more about App Inventor while sharing the resources they find with their peers. This task tries to personalise the experience by prompting students to think about what App Inventor can do for them.

The self and peer assessment is attempting to put the responsibility for learning onto the students. For example, if the work is not done then it is for the students, in their group meeting, to provide helpful feedback and direction to each other.

The teacher is responsible for identifying the ideal time for groups to meet each lesson. This involves students in groups of 3 around one computer and presenting their work to their group. If they don't have access to a space where they can post their work, they could email the task to each group member. This allows them to easily click on the websites that each have found.

Note:

- Descriptions of 'High', 'Medium' and 'Low' for the peer and self assessment can be found at the end of the Week 1 Student handout.
- Peer assessment is an important part of Task 1 and sets the scene for the following tasks.
- It is advisable to collect the handout sheets at the end of each period so that they is not lost.
- Students have until the end of each week to use the advice from their peers as well as their own understandings to improve their work.

Task 2 – Creating 'Hello Purr'

- Ensure that the pdf instructions and the 'Hello Purr' video tutorials are copied onto the school network so that students have access.
- Using the overhead projector show the students how to open both the App Inventor software and the video tutorial.
- Play the initial instructions from the first 'Hello Purr' video tutorial. Then minimise this and demonstrate the action using the App Inventor software. Continue to run the video and pause after each step to demonstrate within the actual software – **you are modelling this method of learning**. Encourage students to use this method on their own computers using headphones. Discourage them from watching the entire video and expecting themselves to remember all the steps.

By listing the skills gained after each tutorial the students are able to build the language of the software. For example, in the 'Hello Purr' skills include: renaming components, button properties, label properties, overlaying a button with a picture, adding sound, using blocks editor, connecting to Android phone.

Task 2 – Sharing, Feedback and Assessment

**Talented students should be encouraged to move on to Paint Pot while weaker students may strengthen their understanding by spending more time on modifying Hello Purr.

Possible Assessment for Week 1:

1. Research
2. Task 1 & 2
3. Ability to help others.

Week 2 – Paint Pot Tutorial (Task 3)

Distribute the week 2 student handout. The next recommended tutorial is 'PaintPot'. Where possible, the teacher should use a student to model the first part of one of the PaintPot tutorial (have both the video tutorial and the App Inventor software open and pause the video tutorial when working on the software).

Open up discussion for tips, hints and common issues from previous lessons. The PaintPot instructions from the author of the App Inventor book are available from http://www.appinventor.org/paintpot.

Students continue with the tutorials and are supported by a group of peers.

Note:

1. Peer feedback & support – each period, get students together in their groups to discuss what they have done. They are expected to show their group the work that they have done and discuss any problems they have had.

2. Discussion of the PDF tutorial - the PDF versions of each tutorial can be very helpful. The groups should also be encouraged to look through the appropriate PaintPot pdf tutorial available. Using a combination of video and print media supports different learning styles. Also, after using the video tutorials, it is useful to view the pdf as they provide more detailed information.

Assessment:

1. Task 3   2. Ability to help others   3. Language of the software

Week 3 – Programming Jargon, Mole Mash Game and Review (Task 4 & 5)

During week 3 and 4, further theory from chapter 14 (Understanding an App's Architecture) and 15 (Engineering and Debugging an App) may be useful to support and extend the programming concepts being used.

Distribute the week 3 student handout. Task 4 requires students to discuss a range of terms used – this should be done in their groups although each student should submit their own answers.

The recommended week 3 tutorial is 'MoleMash'. There are five parts to the video tutorial for this game at http://www.appinventor.org/molemash. Note that at this site you can download the Chapter, download the source and download the APK file (package for the phone).

Where possible, the teacher should use a student to model part of a tutorial during each class – this could occur after students have been working on their programming for 15 min, when a talented student can be identified or when a number of students are having trouble.

Demonstrations should be optional for students to watch if they are at different stages.

During each class there should be some class discussion of tips and tricks as well as common problems

Students should meet in their group each lesson to provide peer feedback and constructive advice.

Assessment:

1. Task 4 & 5   2. Ability to help others.

Week 4 – Create your own App

This week students create an App of their choice or a, perhaps, a their own video help tutorial or handout with the aim of helping Year 7 students learn about App Inventor.

Distribute the Week 4 handout.

Students should meet in their group each lesson to provide feedback, support and peer assessment.

Assessment:

1. Task 6   2. Ability to help others (peers and Year 7).