Panel: Using App Inventor in Introductory CS Courses

Meimei Gao  
Mercer County Community College

Julie Johnson  
Vanderbilt University

Dale Reed  
University of Illinois at Chicago

Cate Sheller  
Kirkwood Community College

Lyn Turbak  
(Moderator)  
Wellesley College
Panel Overview

• App Inventor and the NSF-funded *Computational Thinking through Mobile Computing* project

• Panelists: how they used App Inventor in their introductory courses

• Questions/discussion with audience
MIT App Inventor

A visual blocks-based, cloud-based programming environment that democratizes the creation of apps for Android Mobile devices.

Example: here are all the blocks for a raffle app that we’ll run at tomorrow’s App Inventor breakfast (7:00-8:15am in 2502B)
Acknowledgment: This work was supported by the National Science Foundation under Grants 1225680, 1225719, 1225745, 1225976, and 1226216.
Teaching Computational Thinking

• algorithms as computational recipes
• divide/conquer/glue problem solving
• control and data structures
• abstraction & modularity
• algorithmic complexity
• digital representations
• …

1. Leverages features that situate apps in the world.
2. Hinges on event-oriented behavior.
3. Emphasizes useful programs embedded in a social context.
4. Takes advantage of the larger informational ecosystem.
5. Involves design, engineering, and entrepreneurship.
1. High-level abstractions for mobile device features facilitate creating fully functional situated apps

2. Simple approach to event handling makes it easy to specify app behavior.

3. Visual blocks language, cloud-based environment, and live programming with connected device lower barriers to programming.
Mobile CT Project Resources

nsfmobilect.wordpress.com

• Online curricular materials that use App Inventor to teach computational thinking in a mobile context: whole courses, course modules, tutorials, videos, concept & maker cards, live coding quizzes

• Techniques for assessing computational thinking: pre/post course surveys, rubrics for student projects.

• Dissemination and community building: App Inventor educators site, workshops, and publications.
Workshops

3-hour workshops: SIGCSE 2013 & 2014, CCSCNE 2014

3-day workshops: summers 2013 and 2014 (UMass Lowell)
APP INVENTOR  BREAKFAST

CURIOUS ABOUT TEACHING WITH APP INVENTOR?
WANT TO SHARE YOUR EXPERIENCES ABOUT TEACHING WITH APP INVENTOR?

3/7 SAT. 7:00-8:15AM

ROOM 2502B

ALL ARE WELCOME TO THIS FREE EVENT

THIS EVENT IS SPONSORED BY THE NSF COMPUTATIONAL THINKING THROUGH MOBILE COMPUTING TUES GRANTS (DUE 1225680, 1225719, 1225745, 1225976, AND 1226216)
Using App Inventor in Introductory CS Courses

Meimei Gao
Mercer County Community College, NJ
About the Course

• Get start from Fall 2013

• A new course

• Name: Introduction to Programming with Mobile App Development

• Can be a substitute for CS 0 (Alice)
<table>
<thead>
<tr>
<th>Major</th>
<th>Degree</th>
<th># of Students</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Information Systems</td>
<td>AS</td>
<td>14</td>
</tr>
<tr>
<td>Computer Information Technology</td>
<td>AAS</td>
<td>13</td>
</tr>
<tr>
<td>Personal Development</td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>Computer Science</td>
<td>AS</td>
<td>11</td>
</tr>
<tr>
<td>Game Programming</td>
<td>AAS</td>
<td>3</td>
</tr>
<tr>
<td>Digital Media Art</td>
<td>AAS</td>
<td>2</td>
</tr>
<tr>
<td>Liberal Arts</td>
<td>A.A.</td>
<td>2</td>
</tr>
<tr>
<td>Web Design</td>
<td>AAS/Cert</td>
<td>2</td>
</tr>
<tr>
<td>Architecture</td>
<td>AS</td>
<td>1</td>
</tr>
<tr>
<td>Aviation</td>
<td>AAS</td>
<td>1</td>
</tr>
<tr>
<td>Business Administration</td>
<td>AS</td>
<td>1</td>
</tr>
<tr>
<td>Computer Graphics</td>
<td>AAS</td>
<td>1</td>
</tr>
<tr>
<td>Database Administration</td>
<td>Cert</td>
<td>1</td>
</tr>
<tr>
<td>Electronic Engineering Technology</td>
<td>AAS</td>
<td>1</td>
</tr>
<tr>
<td>Networking Engineering Technology</td>
<td>Cert</td>
<td>1</td>
</tr>
<tr>
<td>non-decided major</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>67</td>
</tr>
<tr>
<td>Course Schedule</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Topics</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Overview of computer concepts, programming languages, operating systems and development environments</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Get started, Event-driven programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Properties of components, Variables</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Creating animation applications, Procedures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Conditionals, Procedures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Test#1 &amp; Lists</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Lists, Activities, Web pages/applications</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Repetitions; Persistent Data – TinyDB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9. Texting &amp; Location-aware Apps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10. Midterm Project and Term Project Discussion</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11. Persistent Data – TinyWebDB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12. Web APIs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13. Test#2 &amp; Term Project</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14. Introduction to Android app development using Java</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15. Term Project Presentation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Format: Hybrid

50% face-to-face meeting and 50% online activities

• Online activities (before class):
  • reading, watching tutorials, building apps

• Face-to-face Meeting:
  • students’ demos, discussing problems, going through the key concepts, reviewing self-review questions, building apps

• Online activities (after class):
  • reviewing, projects, working on self-review questions
Emulator or Devices

• We use both
  • Devices are not required for the course.

• Devices
  – 6 tablets (about $1200) in classroom/lab
  – 5 tablets (about $1000) in library
Using App Inventor in a short-course

Get 'em hooked

Julie Johnson
Vanderbilt University
Students

• Freshmen Engineering Majors
• CS/non-CS/undecided mix
• 14 class meetings (1 hour)
• Introduction to what CS is
CS is not...

- just CS1
- programming alone
- recreating a program as described by your instructor

Although these are critical learning milestones in CS1, this is not a good snapshot of what the CS career field is all about.
Just like practicing piano scales is necessary...

...that’s not what being a pianist is all about

But how can we get to the creative, fun, sexy side of CS without them?
Class schedule

• Reading Code
• Interacting with user (input/output)
• Variables, lists
• Animation (sprites)
• If statements
• Loops
• Database storage
• Web access
• Multiple screens
Project

- In groups of 3
- Begins in Lecture 5
- Focus is on creativity
- Introduce Software Engineering process

- Identify user group
- Create Problem Statement
- Brainstorm
- Prototype (not in AI)
- Build
Creativity is the focus

- Computational Thinking
- Constructs, flow
- Teamwork
- Usability
- Marketability
Using App Inventor in

CS 100: Discovering CS

Dale Reed
University of Illinois Chicago (UIC)
UIC:

- Public Research University with ~26,000 students
- 9th most diverse campus in the U.S.
- CS Dept. has 600 students

CS 100:

- Breadth-first introduction of CS to non-majors
- AP CSP equivalent course
- ~50 students each semester since Spring 2013
- ~15% of students are nursing students
- bit.ly/cs100
App Inventor in CS 100

- About 25% of course is App Inventor:
  - 9 of 15 labs are in AppInventor
  - 3 of 10 assignments are in AppInventor

- Topics: Intro. to programming:
  Variables, loops, decision statements; handling text, TinyDB, TinyWebDB

- Programs: Games or Utilities

- Logistics:
  - We use the emulator in lab.
  - About 30% of students have Android
  - We have a “library” of 10 Android devices for loan from Chicago CSTA

Apps: Gabriel Palomino (F'14), Filip Radzikowski (F'13), Anthony Ochoa (F'14)
Results

• “When I go to parties and people talk about technology, I can join in the conversation.”

• Though not the goal, a few students (1-2 each semester) do end up pursuing CS

• Students understand they can be producers and not just consumers of technology.
  • App to count animal observation for biology field work in Australia
  • Games

dalereed@gmail.com
bit.ly/cs100
Wellesley **CS117 Inventing Mobile Apps**

- Full semester **CS0** course based entirely on App Inventor, similar to Dave Wolber’s USF course
- Focus on app design and implementation
- Learn computational thinking concepts in context
- Five projects:
  1. App using randomly chosen (1) user interface (2) media, and (3) sensor components.
  2. Game app
  3. Sound recorder app
  4. Location-based app
  5. Web database app
Sample 2014 CS117 Apps
Web Mashups Course (Eni Mustafaraj)

Guess the Wellesley Places!

Here are the points you just played. Click on the markers to review the Wellesley Places you've visited today!

Clear Map and Return
Questions/Discussion
APP INVENTOR

CURIOUS ABOUT TEACHING WITH APP INVENTOR?
WANT TO SHARE YOUR EXPERIENCES ABOUT TEACHING WITH APP INVENTOR?

3/7 SAT.
7:00-8:15AM

ALL ARE WELCOME TO THIS FREE EVENT

THIS EVENT IS SPONSORED BY THE NSF COMPUTATIONAL THINKING THROUGH MOBILE COMPUTING TUES GRANTS (DUE 1225680, 1225719, 1225745, 1225976, AND 1226216)

BREAKFAST

ROOM 2502B

3/7 SAT.
7:00-8:15AM