Position Statement: App Inventor Instructional Resources for Creating Tangible Apps

Krishnendu Roy
Department of Mathematics and Computer Science
Valdosta State University
1500 N. Patterson St., Valdosta, GA 31698
Email: kroy@valdosta.edu

***Abstract*—App Inventor is one of the most popular block-based programming environments. Currently, there are limited instructional resources that guide students to create tangible apps using App Inventor. In this positional statement we make a case for the need for more App Inventor instructional resources related to tangible apps. We also present our proposal to address this need.**

I. Introduction

App Inventor is currently one of the most popular block-based programming environments. The main goal of App Inventor is to teach computing and programming to students with limited prior programming knowledge and to democratize app creation by providing an easy-to-learn environment. App Inventor has 3 million registered users with close to $100,000$ unique weekly active users. To date, students have created more than 8 million apps using App Inventor.

App Inventor has experienced broad adoption in diverse venues. Researchers have used App Inventor in summer camps and other outreach activities for K12 students for several years now [1], [2], [3], [4], [5]. App Inventor has also been effectively used for professional development workshops for K12 teachers [6], [7], as well as introductory computing courses at the college level [8], [9], [10]. Extremely popular app design competitions like the Verizon Innovative App Challenge and Technovation Challenge also use App Inventor.

Most apps designed using App Inventor run on an Android phone or tablet. However, it is possible to create apps that can go beyond just the virtual realm and somehow interact with the physical world. Examples of such apps that others have already created using App Inventor are Lego Mindstorms robot controller apps, apps that interface with an Arduino board etc. We define such apps as *tangible apps*.

There are many instructional materials available for learning computing and programming using App Inventor, and they range from books to video tutorials. Video tutorials with written instructions are ideal for fast-paced outreach activities like after-school/weekend workshops and summer camps. However, there are very few App Inventor instructional resources that guide a student to create a tangible app. Most of the information available in this area is scattered over multiple posts in the App Inventor Forum Google group and there in no easily-accessible centralized list of resources.

In this project we attempt to address this void by:

1) Creating a master list of instructional resources related to tangible apps created using App Inventor.
2) Modifying existing instructional resources to make them more suitable for K12 outreach and creating new resources to augment the existing ones.

II. Motivation for Tangible Apps

We have organized computing summer camps at our university using App Inventor for the past five summers. From our anecdotal experience, one of the most popular and enjoyable App Inventor app that the students created was the Lego Mindstorms NXT robot controller app. We believe the "coolness" factor of an app interacting with the physical world cannot be matched by non-tangible apps.

Summer camps organized at Clemson University also reported similar experience in [2]. The most popular App Inventor app in their camp was a treasure-hunt game app that involved tangible interaction using QR codes.

Other recent research projects have also developed block-based programming environments that generate tangible artifacts and interactions. Turbak et al. in [11] reported about a block-based programming environment that allows users to create tangible laser/vinyl cut artifacts. Ardublocks[1] is another project that lets users program Arduino microcontrollers using a block-based interface. Deitrick et al. in [12] reported about BlockyTalky, another block-based programming environment supporting tangible interaction which is also focuses on parallel and distributed programming. Scratch, one of the other most popular block-based programming environments has supported tangible interactions through PicoBoards for a while now. Tickle[2] is another block-based programming environment offering tangible output. Users can use Tickle to program Arduino boards, robots, and even drones and smart home devices. Spherly[3] is a block-based programming environment to program Sphero robots. When first introduced in 2014, Google's *Made with Code* project[4] allowed users to create personalized bracelets using a block-based programming environments. These bracelets were then 3-D printed and shipped to the users for free. Another current block-based tool of that same project allows users to create t-shirt designs with embedded LED lights.

[1] http://blog.ardublock.com
[2] https://tickleapp.com/en-us/
[3] http://outreach.cs.ua.edu/spherly/
[4] https://www.madewithcode.com/projects

978-1-4673-8367-7/15/$31.00 ©2015 IEEE

Based on the success of these projects, we think block-based programming environments generating tangible artifacts and interactions will become increasingly popular in near future. Hence, tangible apps created using App Inventor are an important educational resource and more work needs to be done in streamlining the existing instructional resources and developing new ones.

III. Proposed Work

We plan to achieve two main goals in this project. Our first goal is to create an easily-accessible master list of all the existing instructional resources related to tangible App Inventor apps. To come up with the master list, we mainly plan to look in to the App Inventor website, the App Inventor Forum and the App Inventor Gallery.

Our second goal is to a) make sure that the instructional resources included in the master list are suitable for fast-paced K12 outreach activities like a summer camp, and b) create new instructional resources. Summer camps are usually one week long. In this limited time, there is not much scope for learning details of all aspects of programming. Hence, one of the goals of summer camps is to make the students interested in computing and enhance their positive attitude towards computing and programming so that they continue to explore computing in future. Since many of the students in a summer camp will be learning programming for the first time, confusing instructional resources can negatively impact their attitude towards computing in general. Additionally, many camps are organized with the help of K12 teachers who might be inexperienced in App Inventor themselves. Hence, easy-to-understand instructional resources with clear concise directions are very important for success of an outreach activity like a summer camp.

To achieve this second goal, we plan to modify some of the existing resources. Some of the existing resources were designed using App Inventor Classic. We plan to re-implement them using the latest version of App Inventor. Some other resources lacks detailed description. In those cases we plan to create detailed descriptions.

Additionally, we plan to create some new tangible apps and associated instructional resources. We have some preliminary ideas about what tangible apps we want to create:

- Lego Mindstorms robots are one of the most popular robotics kits and are widely used in K12 classrooms. App Inventor had blocks to interface with the NXT (previous) version of the Lego robots. But, App Inventor does not support the latest version, Lego EV3 robots. We want to see if we can build that support in App Inventor or not.
- NFC tags have become relatively inexpensive now. So we plan to create some apps leveraging App Inventor's NFC blocks.
- Finally, we plan to create some more apps that can control an Arduino. For example, one of the app that we plan to develop will enable an user to open and close window blinds using a smartphone. A second app will use an Arduino to light up an LED with different colors to denote how hot/cold a day's temperature will be.

We plan to create written as well as video tutorials for all these apps. We also plan to create a list of possible variations for each app so that students can further explore their creations.

IV. Conclusion

In this position statement we make a case for increasing popularity of tangible apps and the need to develop more instructional resources that demonstrates how to create tangible apps using App Inventor. We believe tangible apps are going to be more common in future. Hence, instructional resources related to tangible apps will be an exciting augmentation to the current set of App Inventor instructional resources. Successful completion of this project will open the door to new opportunities like organizing tangible-apps related summer camps and organizing teachers' workshops similar to Google's CS4HS workshops, with the main focus being tangible apps.

References

[1] S. AlHumoud, H. S. Al-Khalifa, M. Al-Razgan, and A. Alfaries, "Using app inventor and lego mindstorm nxt in a summer camp to attract high school girls to computing fields," in *Global Engineering Education Conference (EDUCON), 2014 IEEE*. IEEE, 2014, pp. 173–177.

[2] M. H. Dabney, B. C. Dean, and T. Rogers, "No sensor left behind: enriching computing education with mobile devices," in *Proceeding of the 44th ACM technical symposium on Computer science education*. ACM, 2013, pp. 627–632.

[3] B. Ericson and T. McKlin, "Effective and sustainable computing summer camps," in *Proceedings of the 43rd ACM technical symposium on Computer Science Education*. ACM, 2012, pp. 289–294.

[4] K. Roy, "App inventor for android: report from a summer camp," in *Proceedings of the 43rd ACM technical symposium on Computer Science Education*. ACM, 2012, pp. 283–288.

[5] A. Wagner, J. Gray, J. Corley, and D. Wolber, "Using app inventor in a k-12 summer camp," in *Proceeding of the 44th ACM technical symposium on Computer science education*. ACM, 2013, pp. 621–626.

[6] J. Gray, H. Abelson, D. Wolber, and M. Friend, "Teaching cs principles with app inventor," in *Proceedings of the 50th Annual Southeast Regional Conference*. ACM, 2012, pp. 405–406.

[7] J. Liu, C.-H. Lin, P. Potter, E. P. Hasson, Z. D. Barnett, and M. Singleton, "Going mobile with app inventor for android: a one-week computing workshop for k-12 teachers," in *Proceeding of the 44th ACM technical symposium on Computer science education*. ACM, 2013, pp. 433–438.

[8] H. Abelson, R. Morelli, S. Kakavouli, E. Mustafaraj, and F. Turbak, "Teaching cs0 with mobile apps using app inventor for android," *Journal of Computing Sciences in Colleges*, vol. 27, no. 6, pp. 16–18, 2012.

[9] E. Spertus, M. L. Chang, P. Gestwicki, and D. Wolber, "Novel approaches to cs 0 with app inventor for android," in *Proceedings of the 41st ACM technical symposium on Computer science education*. ACM, 2010, pp. 325–326.

[10] D. Wolber, "App inventor and real-world motivation," in *Proceedings of the 42nd ACM technical symposium on Computer science education*. ACM, 2011, pp. 601–606.

[11] F. Turbak, S. Sandu, O. Kotsopoulos, E. Erdman, E. Davis, and K. Chadha, "Blocks languages for creating tangible artifacts," in *Visual Languages and Human-Centric Computing (VL/HCC), 2012 IEEE Symposium on*. IEEE, 2012, pp. 137–144.

[12] E. Deitrick, J. Sanford, and R. B. Shapiro, "Blockytalky: A low-cost, extensible, open source, programmable, networked toolkit for tangible creation," in *Proceedings of Conference on Interaction Design for Children, Aarhus, Denmark*, 2014.