Tips and Tricks for Developing Successful Technology Projects for Introduction to Programming and Outreach

Anca Doloc-Mihu
Georgia Gwinnett College
Lawrenceville, GA, U.S.A.
adolocmihu@ggc.edu

Cindy Robertson
Georgia Gwinnett College
Lawrenceville, GA, U.S.A.
crobertson2@ggc.edu

ABSTRACT

We are proposing a workshop that instructs the audience on how to develop fun and engaging projects involving different technologies. These types of projects can be used by faculty to introduce programming skills in the classroom, but they are also a fantastic resource for outreach activities. Students can create these projects as a way to develop and promote their technical skills. Presenters will also share best practices and allow for discussion in creating and then conducting these projects at their institution.

CCS CONCEPTS

• **Applied computing → Interactive learning environments**.

KEYWORDS

project-based learning, teaching, block coding, critical thinking, cs, education, it, outreach, programming

1 DETAILED DESCRIPTION

There is a great need for a diverse and capable workforce to maintain our nation's excellence in science, technology, engineering, and mathematics (STEM). In particular, women and minorities are severely underrepresented in the information technology (IT) fields. In addition to a standard curriculum, outreach activities have been successfully used by many institutions to attract more students to IT. Our Technology Ambassadors Program (TAP) at Georgia Gwinnett College (GGC) has been a successful IT outreach program for eleven years, training a diverse population of college-level students to develop and conduct engaging and fun technology outreach projects to audiences ranging from K-12 students to general education college classes at GGC.

More recently, we have found that these projects [1], originally designed as outreach activities, are also very good resources for introducing basic programming concepts in our classrooms. Our

previous work has shown that when our students use block coding as a means to slowly and easily introduce coding concepts, they are more prepared for text-based programming skills. Using block coding to program a game or activity to learn algorithmic thinking is not only more fun, but more importantly it is a better way to develop the step-by-step process of writing coding instructions and is more enticing to students. This flow allows them to understand and be able to apply basic computing concepts and enjoy the learning process.

In this workshop, we will introduce the audience to different types of engaging projects that faculty and students perform at our institution and to our TAP repository website where we host freely available projects. We aim to spark lively discussions about how to best introduce programming skills to engage audiences of any age and establish new collaborations with audience members.

1.1 Target Audience and Topics

Our target audience is anyone interested in introducing programming concepts by using fun and engaging projects involving various technologies - this includes both faculty and students. Faculty can use these projects for introducing basic programming concepts in the classroom as well as STEM outreach to increase and broaden participation in computing at their institution or in the community. These types of projects work well for K-12 students as well as college-age students. Students can create these projects to not only showcase their technical skills but to acquire new skills to add to their resumes.

1.2 Outline

(10 min) Part 1. Introduction to our Projects: This session will briefly introduce the audience to our TAP program, where the idea for these programming projects originated from. The audience will then be introduced to our TAP website which showcases examples of projects developed by students during the program and implemented in our classrooms and at our many outreach events.

(45 min) Part 2. Programming Projects: This part will include two sessions that will demonstrate two different programming projects that our students have created and successfully implemented in our classrooms. Each session is a hands-on activity where the audience will be requested to participate in and interact with the respective technology.

- *(15 min) Demos.* This is a short session during which the audience will interact with various technologies (Cozmo, Sphero, drones, VR, AR, JS, etc). For example, they see how we use block programming to teach students how to maneuver a robot through a maze.

- *(30 min) Workshop.* During this session, the audience will get to experience one of our programming workshops. First, they will learn how to interact with a technology and then they will be engaged in a hands-on step-by-step activity teaching them how to program the technology. The audience will also be challenged to build a simple code, for example, to maneuver a robot.

(20-45 min) Part 3. Project Development: This is a hands-on session during which groups of 2-3 participants will brainstorm and develop their own project using the technologies learned. Our TAP faculty and students will act as mentors during this session.

(25 min) Part 4. Discussions: This session will have two parts as follows:

- *(15 min) Panel Discussions.* During this session our TAP panelists, consisting of both faculty and students, will engage the audience in lively discussions related to best practices for introducing programming concepts and running a successful outreach program.
- *(10 min) Q & A.* The audience will have the opportunity to ask questions about the website content and workshop projects.

1.3 Execution Plan

Each session/part of the workshop will be conducted by 1-2 designated TAP faculty. The Part 2 session will also include projects conducted by TAP students. Students will demonstrate several different types of technologies to showcase the various projects on our website. This is a hands-on session where the audience will interact directly with technologies.

During the Part 3 session, the audience will develop their own project using the learned technologies. The duration of this session can be adjusted to be anywhere between 20 min to 45 min depending on the participants' interests and the time allotted for the workshop.

The discussion session will include both TAP faculty and students and aims to engage the audience in discussions of best practices for introducing programming and algorithmic thinking into the classroom. If the audience is interested, we will also discuss our outreach program. A faculty member will moderate this session.

2 KEY TAKEAWAYS

During our workshop, we will share our experience in implementing programming projects in the classroom in an effort to foster a diverse student population to become future technology professionals. The proposed workshop will introduce attendees to our program, allowing them to experience both short demonstrations and a longer workshop that focuses on a single technology. The attendees will interact with a panel of both faculty and students to discuss approaches and learn about our newly constructed project repository from which they can adopt our freely available materials, or if interested, how to set up an outreach program at their institution.

The workshop will showcase TAP, which is a service-learning outreach program at GGC, which has been using programming workshops to recruit minority students to the field of IT since its inception. This program was the catalyst for our new approaches

to introducing programming in our classrooms. We will discuss best practices for using these projects effectively.

ACKNOWLEDGMENTS

REFERENCES

[1] Cindy Robertson and Anca Doloc-Mihu. 2023. Understanding College Level Student Learning of Basic Programming at an Open Access Institution. In *Proceedings of the 2023 ACM Southeast Conference.* Association for Computing Machinery, New York, NY, USA, 26–32. https://doi.org/10.1145/3564746.3587007