

Fairy Houses: A Creative Engineering Experience

Andres Gonzalez, Robert Fowler, Harrison Froeschke, Sabra Leong,
and Bruce Gooch^(✉)

Texas A&M, College Station, USA
gooch@cse.tamu.edu

Abstract. A common misconception is that neither Engineers nor their projects are creative. We report on a laboratory program for creating educational displays like those in Museums. The program enables students to (1) design and build an edutainment product, (2) gather data to test their product, and (3) interpret their data. The objective of the project is to create an educational experience for users that shows the potential for creativity within the field of Computer Science. The undergraduate participants created an interactive display that teaches two Computer Science concepts. They created an interactive experience for via a Bluetooth connection to an Arduino powered “Fairy House.” Users unravel a story that involves solving riddles via interpreting Morse code. The display has two learning goals. First, players gain an understanding of how to turn code into text. The second is players discover how binary trees speed search tasks. We also had two design goals, any number of people can play, and players can join or leave without interrupting others. Through in-house testing and trial runs, the students determined that this method of interactive learning is useful for teaching new concepts.

Keywords: Fairy house · Chat-bot

1 Introduction

Most people know about Morse code. Yet, few understand it. The task of decoding streaming Morse code using a table is involved. One must search the entire table for every dot or dash. Decoding streaming Morse code using a binary tree is simple. One traverses the tree by following the pattern of dots and dashes to find a letter. So, learning Morse code fits both of our educational goals. We now needed a compelling story to engage users.

We decided that a light display is a good way for users to learn Morse code. During World War 1, the Allies used Begbie Lamps to relay messages on the battlefield. After World War 1, photos of the Cottingley Fairies arose. The photos show two girls with what appears to be fairies. We came up with a story of English Fairies who pledged to defend the Empire. King George assigned them to the Royal Signal Corps. They learned Morse code to send messages to humans. Following the war, our fairies left the violent world of men for 100 years and have now just returned.

The story gave us the idea of creating a fairy village with the houses made from signal lanterns. The fairy houses flash at one another as if talking. When a person approaches, the community falls silent as if to see whom the new person is. One fairy house has a sign explaining how to communicate with fairies. The houses send simple Morse code questions via LED lights. The Fairy Village display provides two charts to aid users in deciphering Morse code, a Morse code table and a Morse code tree. The user can send text messages to the fairies using a Bluetooth-enabled mobile device. This simple communication enables people to learn about Morse code and binary trees.

2 Related Work

There have been other projects that have tried to implement Computer Science to engage users in a mix of the virtual and physical environment. Chit Chat Club [3] attempts to this by creating human-sized avatars that attempt to engage users, both present and remote, using facial expression. Our project however takes this ideology a step further as there is communication via the messages sent by the house and the light flashes to give the user feedback on different levels.

Another similar idea to our project is Architales, an interactive story table that creates an interactive environment for users to give users an enjoyable engagement with applications. This project is very similar to what we aimed to do, as the table creates an interactive environment that allows users to be immersed and learn about the story and see computer science in action.

In 2015, The Harmonic Walk was developed at the University of Padova in the Department of Information Engineering to create an “interactive physical environment designed for learning and practicing the accompaniment of a tonal melody.” This project, in a similar way to the Fairy House, presents a method of interaction for the user to learn about something new to them. In the case of music for the researchers at Padova, users learned about tonal melody as well as chord progressions, melody accompaniment, and improvisation. An important distinction made in this project that is also made in the Fairy House Project is the distinction of skill level by the users of the interactive environment.

3 Methods

3.1 Hardware

Each fairy house contains an Arduino Uno and Raspberry Pi to provide computation and I/O control. Their small form factor allows us to hide the hardware inside the houses. Also, in order to update the output future programmers need no knowledge of Arduino. We used Arduino HC-05 Bluetooth modules to send messages to participant’s mobile phones. An RGB LED serves as the Morse code output (Fig. 1).

The original idea for our fairy houses was to retrofit solar powered lanterns with proximity and light sensors. Users would interact with the lanterns by moving near them and shining or flashing lights at them. Multiple lanterns would be connected via a

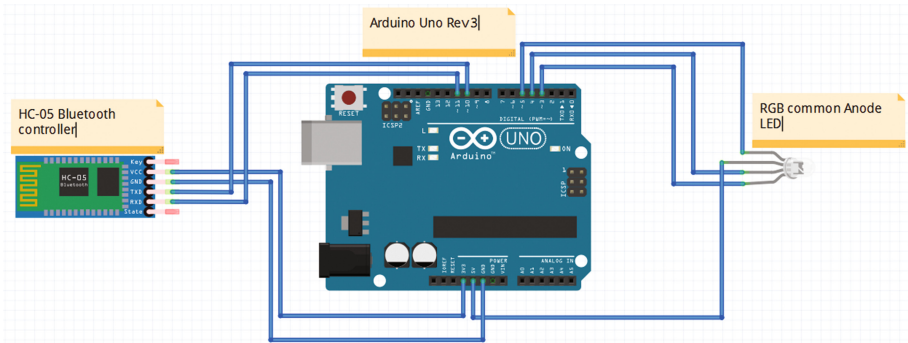


Fig. 1. Diagram of the Arduino hardware.

network and would communicate and interact with each other based on the actions of the users. The program for this networking would run on Raspberry Pi's installed in the bottom of every lantern.

The first major change to the project was the use of Arduino's instead of Raspberry Pi's. The reason for this change was the design simplicity Arduino possessed compared to the Raspberry Pi. Arduino's capability to run small machines such as sensors or servos using a small amount of RAM was much more appropriate than the more advance Raspberry Pi. Since we intend to create a large number of these lanterns, it would also be cheaper to use the Arduino's. However, as we began to work with Arduino's in the houses, we found that the networking capabilities were a challenge for us to effectively set up. We decided to drop this feature, as it would also allow more people to play the game separately in an area containing many lanterns instead of one person or group using the entire network (Fig. 2).

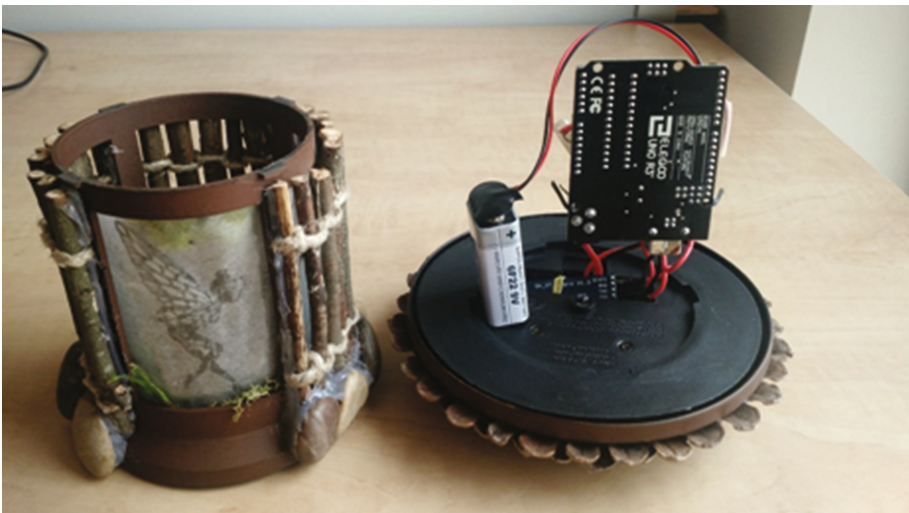


Fig. 2. The internal hardware of a fairy house.

3.2 Software

Our goal was to create an interactive art installation playable by multiple people at a time. The primary purpose of this project is to demonstrate that computer science and engineering are creative fields. We wanted to create something entirely new that demonstrated that there are always new forms of art and expression.

The entire project is built on the Arduino platform. We have found that its ease of use for people who may not have experience with microcontrollers, coupled with its versatility, lead to a system that everyone could contribute to while not becoming bloated with extra parts. In this specific case, we decided on the UNO Rev3 for its form factor, low power consumption, and its ability to handle all the I/O that was required of it.

Originally we intended to use proximity sensors and light sensors as affordances for interaction with the lanterns. However testing demonstrated that users found only Morse code interaction to be too slow and cumbersome. Users became rapidly bored.

These results lead the team to consider text communication. We installed a HC-05 Bluetooth module into the lantern. The UNO is currently using a HC-05 Bluetooth module as its primary means of input which also doubles as an output source. This way, a user can connect to the lantern via Bluetooth and send it messages just by texting it. The lantern would now more effectively communicate with the user and use Morse code for certain messages. Since we knew how the lanterns could communicate, we needed to figure out what specifically they would say to the users (Fig. 3).

Participants can use any Bluetooth terminal app to connect to a fairy house. This allowed us to have an input source that is in the participant's hands and makes them all that much more engaged in the exercise, while also allowing us to have a standardized way of giving instructions, hints, etc., via printing out to the participants terminal screen.

The hardware level code is straightforward. A nested for loop converts words to characters. Characters are mapped to a corresponding series of dots and dashes using a switch statement. If no input from the participant is detected the message will simply repeat until it detects new information, then depending on the input either a new message will start to be broadcast or the encoded message will repeat but the Bluetooth terminal will display a reason why the message has not changed.

Our first idea was to install a chat-bot into the lantern for the user to converse with. However this presented a problem because if the user wanted to communicate any complicated sentences, then the user would have to have a complete mastery of Morse code. As a result, we decided upon another idea that reduced the amount of Morse code needing to be interpreted. This idea would have the user begin the game by connecting to the fairy house and texting it "Hello". The fairy in a house would ask the user riddles. By solving the riddles users gain trust of the fairy and unlock further dialogue into the fairies' backstory.

As the user is reading the riddle, the lantern that they are connected to would flash the answer in Morse code. We found that most users were not familiar with interpreting Morse code. We made it easier to distinguish between dashes and dots by making dashes a blue light and dot a red light. We found that coloring the lights significantly lowered the cognitive load of Morse code translation for novices. They were able to solve the fairy's riddle, and send a text with the answer.



Fig. 3. A completed fairy house.

A portion of the project was designing the look and feel of the lantern to give the appearance of a Fairy house. We spray painted the lantern brown, covered it in twigs, rocks, and left a window in the side to show an image of a fairy. Inside the lantern, along with all of the electronic components, is a circular piece of paper with a scene of magical fairies drawn on it. When the lantern flashes, the scene is illuminated and can be easily seen in the dark.

To tell the fairy’s story we developed a text-based adventure. We used the ADRIFT software package. It allowed us to map the story with response paths based on user/fairy questions and answers. The text-based adventure allows players have a different set of interactions with the fairies. While the users may traverse different paths, the final story is always complete. Users are able to “find their spot” once they return to a Fairy House for a new session at a new time and even new location by giving key words to the fairies that trigger a response.

4 Evaluation

4.1 Interactive Display

We evaluated the fairy village using a practice-oriented model developed by Nacke. The Nacke standard calls for separate testing of the technology, gameplay, and social quality. We used Arduino TestSuite to unit test the hardware for correct input and output. We tested the text-based adventure software using the HP software evaluation tools. We tested that; software is installable on the RaspberryPi, runs in real-time, and responses are correct for given input. Gameplay analysis proceeded in three phases. First by expert evaluation, second a usability study. Third, we measured learning outcomes in players with a survey.

The testing that has been completed about the interactivity and ability to learn from the communication with the house shows that this medium is optimal for teaching and showing the creativity that can be found in Computer Science. High School students and college students alike are able to interact and learn about the creativity that engineering has to offer in a world where technology is so prevalent and only becoming increasing so. With Fairy Houses set up in various locations, users will be able to stumble upon a world hidden in plain sight, akin to the world of programming and engineering. With this information, we hope that readers will use the findings in this paper to implement teaching strategies to educate while creating a fun environment and experience for users.

4.2 Student Team

Learning is a multi-faceted activity, and almost everyone has a learning style preference. Most of us use components from many styles when we try to learn something. Since there are so many ways to learn, it follows that there should be many ways to evaluate how effectively learning, and the corresponding teaching, are taking place. Therefore, we used a multi-dimensional evaluation strategy to determine if the learning goals of the project were met.

We conducted evaluation along three axes: the student, a neutral observer, and the instructor. Results from each evaluation axis were analyzed individually, and then triangulated against the results of the others.

Student Provided Data. Research method: Qualitative design journal, Quantitative questionnaire. Each student created a design blog. The design blog gave the instructor

an opportunity to assess students' knowledge of HCI concepts as well as enable students to develop critical thinking skills applicable to software and hardware design. We evaluated student learning via lab reports, project documentation, and a project presentation. We also used a novel method of documenting student learning. The undergraduate students developed a second "making of" text-based adventure. It tells the story of a creative team of engineers bringing a fairy village to life.

The second method for collecting student data was a questionnaire to establish baseline information about the learners. The following list of questions is examples; How do they rate themselves as programmers? How experienced are they with Arduino? How "interested" are they game design? How experienced are they with a research and reporting. The questionnaire allowed us to answer the following questions:

Did the students with more experience do better than those without? In the case of hardware the answer was yes team members who were previously Arduino hobbyists were able to answer questions and suggest solutions that other team members were not. In the case of software all students were evenly matched and all were able to make contributions. In the case of game design two students had far more experience, however all team members were able to make significant contributions to the final product.

How long does it take for learners to consider themselves "experienced"? By the end of a one-semester course all of the students considered themselves experienced in system design and reporting. All of the students started the course believe that they were experienced in software design and programming. All of them admitted at the end that their initial assessment was in error as they had learned a great deal about software engineering and programming. Students universally remarked on the post course questionnaire that they learned a great deal about programming that is, "not in the books."

System Provided Data. Research method: Quantitative data collection from within the software development system. We tracked GitHub commits and changes as well as user communication statics during project development. Initially extroverted students contributed more to discussions. After three weeks of meeting and talking in a round table fashion all students were comfortable enough to contribute to online and face-to-face. GitHub commits showed no significant change over the course of the project. This may be due to the tight scheduling and oversight that the students enforced upon them.

Observational Data. Research method: Qualitative observation. Looking for things that learners enjoy and find frustrating can quickly lead instructors to enhance certain areas of the project while fixing others. Trained observers can quickly see patterns and suggest modifications. Moreover, the observers can quickly see if their suggestions were effective. Learners consistently identify responsiveness as one of the most important dimensions of instructor "service" to students. Close observation of potential problems will let us respond in a very timely manner. We employed a neutral observer of the project who observed team meetings and reviewed the team's research blogs. The observations were shared with the instructor and team during the project. The observer documented their observations on the following: How do the students interact

with one another? How do the students interact with the instructor? Where does most of the “enjoyment” seem to be coming from? Where is most of the frustration coming from?

Student interaction with the instructor evolved over the course of the project. Initially students assumed that they would be given an assignment. By the end of the project they were able to interact with the instructor as a research peer. The enjoyment came from interaction in the laboratory and from exceeding the expectations that the students had of them. Student frustration in the beginning of the project was due to the open-ended nature of the project. During the reporting portion of the project some of the students were dismayed by the fact that their writing was not like that of the research papers they were reading.

Instructor Data. Research method: Qualitative structured anecdotal recording. Reflective practice is a way in which educators can “develop a greater level of self-awareness about the nature and impact of their performance”. This practice involves looking critically at our own work as instructors. Questions that we pose to ourselves can start this process such as: What “worked” and why?

5 Conclusion

Computer Science is typically viewed as a field that lacks creativity due to the prevalent stigma of the field being boring. As a result, many creative minds are off put by the field that they believe to be inhospitable to their creativity. Changing this stigma is vital for the future of the field, so that a variety of thoughts and ideals are able to help shape the future and drive innovation. This is an important issue to address because as we move towards increasingly difficult challenges, varied ways of thinking will be crucial for finding different approaches to the problems of tomorrow. Many typical approaches to this issue fail miserably as looking at the creative side of computer science and exemplifying this side to students is forgotten. Some previous approaches have tried to show this creativity through programs, having a tangible object to interact with and receive visual feedback from engages people and sparks interests. Our project shows the creativity that is found in Computer Science by generating critical thinking, engagement, learning a tangible skill, and having a visually stimulating object to interact with.

6 Future Work

In the future, we would like to look into other methods of creating entertaining environments for users to learn about not only an interesting topic, but also the creativity of computer science. One way that this could be done is evolving the Fairy House system into an experience that includes motion detection and light detection to make the houses even more interactive. The motion detection will have the lights projected by the fairy house intensify as users draw near to the houses, and have the lights fade when no motion is detected. The light detection will allow the users to use a

light source to communicate back to the fairies in Morse code, making the act of learning Morse code even more interactive. In this way, a more immersive experience for the user can encourage repeated interactions as well as spread of the popularity and bring more users to the Project to learn about Computer Science and Engineering.

In addition to advancements with the Fairy Houses, we will also begin development on a teaching platform based on the experience we gained from working with the users of the Fairy House Project. The new project will implement an interactive achievement system for students so that they will have constant positive feedback and response to make an otherwise menial task more engaging.

References

1. https://cfacaa.human.cornell.edu/dea.arl/linked%20docs/Green_DIS2014%20.pdf
2. <http://mf.media.mit.edu/pubs/conference/TangibleNavigation.pdf>
3. Karahalios, K.G., Dobson, K.: Chat club: bridging virtual and physical space for social interaction. In: Proceedings of 2005 Extended Abstracts on Human Factors in Computing Systems (2005)
4. Karahalios, K., Bergstrom, T., Yapchaian, M.: The ISEA chit chat club. Ginger - an installation. ISEA c4f3 2006, San Jose, CA (2006)
5. <http://social.cs.uiuc.edu/projects/Ginger/index.html>
6. <http://social.cs.uiuc.edu/people/karriekarahalios.html>
7. Mazalek, A., Winegarden, C., Al-Haddad, T., Robinson, S.J., Wu, C.-S.: Architales: physical/digital co-design of an interactive story table. In: Proceedings of the 3rd International Conference on Tangible and Embedded Interaction 2009, Cambridge, UK (2009)
8. http://synlab.gatech.edu/data/papers/mazalek_tei2009_architales.pdf