

**Overview.** I used the Dick and Carey Systems Approach model for Designing Instruction to guide my work on this project. I selected this particular model because it is a systems approach and steps are done systematically. This type of instruction is based on completion of step after step. The steps included in the Dick and Carey Systems Approach Model for Designing Instruction are:

1. Identify Instructional Goals
  - a. Members will be able to build basic robotics commands (move forward, stop, move backwards to starting position) that will move the MBot robot wirelessly from a MacBook laptop loaded with the mBot programming software.
2. Conduct Instructional Analysis
  - a. Completed a goal analysis, which helped to guide me in how much I needed to breakdown the programming steps in the instruction.
  - b. Completed a learner analysis by asking questions in one of the Science Club meetings prior to instruction. By show of hands, members answered questions giving me an idea of their background with computers and robotics.
3. Identify Entry Behaviors and Learner Characteristics
  - a. Members will need basic mechanical skills, basic computer skills, knowledge of how to access the mBot robotics application and use of its menus. Awareness of parts of the mBot robots and how they receive instruction from the laptop is also required.

4. Write Performance Objectives
  - a. These are listed below under **Instructional goals and objectives.**
5. Develop Criterion Standards
  - a. Observed the members in a Science Club meeting prior to creating the instruction to gauge what their intellectual levels were as well as their ability to take instructions and apply them to the task at hand.
6. Develop Instructional Strategies
  - a. Due to the students working in groups of two, I developed a screencast showing the members how to program their mBots. This screencast was shown to the members as a group but available to them individually as they worked on their challenges. An empty flowchart worksheet was provided for the members to fill out as a tool to assist them in programming their mBots.
  - b. Instruction given during a regular Wednesday after school meeting of the Science Club in Dr. Batten's 7<sup>th</sup> grade Science classroom where the Science Club normally meets.
7. Develop Instructional Materials
  - a. Screencast showing how to navigate the mBot programming software menus.
  - b. Screencast showing how to program the mBot.

- c. Empty flowchart worksheet for members to fill out to assist them in programming their mBots based on instruction from the screencast.
8. Design and Conduct Formative Evaluation
    - a. Used overt responses after the group watched both screencasts to monitor understanding.
    - b. Circulated around groups and quizzed individuals on what they expected each programming step would do to the robot.
  9. Design and Conduct Summative Evaluation
    - a. Students were evaluated based on their mBot's performance on the movement tasks and subsequent "repeat" programming challenge.

Members will need basic mechanical skills, basic computer skills, and knowledge of how to access the mBot application. Awareness of parts of the mBot robots and how they receive instruction from the laptop is also required. Instruction was given during a regular Wednesday after school meeting of the Science Club in Dr. Batten's 7<sup>th</sup> grade Science classroom where the Science Club normally meets. Most members completed the instruction and challenges within the 90-minute time frame of the Science Club meeting.

**Front-end analyses.** Two analyses were performed: a goal analysis and a learner analysis. The goal analysis was performed in order to help with how much instruction to include in one module. How many programming steps can be included? Will the members have enough of a mechanical skill set to understand

what is being asked of them? Will there be enough time for members to receive the instruction and then put it in practice in order to complete their challenges? The results of the goal analysis showed me that I would not be able to present the amount of instruction that I had originally intended. Due to time constraints because of holidays and other obligations that interfere with the Science Club meeting, only one instructional period was available for this instruction. Therefore, Dr. Batten worked with the Science Club members in other meetings to get the mBots assembled, the mBot programming software loaded, and verify that mBot programming software was able to control the mBots wirelessly.

A learner analysis was performed in order to pair members up based on experience with MacBooks, as most of the younger students do not have MacBook experience. The school migrated from MacBooks to Chromebooks this school year starting with incoming 6<sup>th</sup> graders. The goal of this is to pair a less experience member with a more experienced member in order to allow the instruction and exercises to move at a reasonable pace. Also, a handful of students have some exposure to robots and programming due to attending Dr. Batten's Robotics camp over the previous summer. These students were paired with completely novice members.

**Instructional Goals and Objectives.** Instructional goals and objectives for the initial mBot programming module are:

Instructional goal: Members will be able to build basic commands (move forward, stop, move backwards to starting position) that will move the mBot wirelessly from a MacBook laptop loaded with the mBot programming software.

Objectives:

1. After completing the programming lesson, members will be able to successfully navigate the mBot software menus in order to build programs with 100% mastery.
2. Members will be able to construct a simple program to move the robot a forwards and backwards with 100% accuracy.
3. Hawley Science Club members will be able to successfully complete programming challenges consisting of various parameters, such as speed and time, using the mBot programming software on their school issued MacBooks to program their individual mBot robots with 80% mastery in order to move on to future programming modules.

**Instructional Strategies.** Pre-instructional activities that I intended to use for my project were completed in the previous meeting of the Science Club. Those activities were the installation of the mBot programming software onto the school issued computers and setting up the mBot and laptop to connect to each other wirelessly. A routine strategy that I used was a screencast showing how to create their first command program. There was a poster of all of the parts of the mBot laminated and available for the students to look at any time. Another routine strategy I used is letting the members of the science club practice building command programs before attempting their assessment challenge. Dr. Batten, the Science Club advisor, walked around and quizzed the students to see if they understood what was going on and gave them feedback to their questions and answers and his observations of their activity. The challenges were completed at the end of the instructional period. The challenges must be

successful in order for the students to move on to the next programming lesson at the next meeting of the science club. Students will complete a flowchart activity as an enrichment strategy. This will allow them to understand the ordered set of actions they need to complete in order to successfully command their robot to perform the specified commands and pass their challenge. This also introduces students to how real programmers use flowcharts to map out programs prior to writing code.

**Instruction.** The idea behind the creation of the mBot robots is for every child to have one of their own. They are inexpensive, easily programmable robots that were originally offered as a Kickstarter project. The G.C. Hawley Science Club was able to acquire funding to purchase ten of the mBots. Since there are seventeen students in the Science Club, the students must be paired up. Prior to this instruction, the Science Club members assembled the mBots and installed the programming software in pairs of two. They were paired up based on the answers to the learner analysis questions. The members also used the included remote to operate the mBots, just to be able to control the mBots via a method they were familiar with.

My instruction module introduced them to the mBot programming software interface. The interface uses Scratch programming as well as Arduino programming. For the purpose of my instruction, we used the graphical user interface of Scratch as Arduino is pretty advanced for the target audience. Even I, the instruction creator, had a hard time wrapping my head around the Arduino code. When first opening the programming software, a lot of what is

shown is not relevant for the purpose of this instruction, such as the penguin sprite and a few of the menu options.



Two screencasts were produced for this instruction. One gave an [overview of the menu items](#) and showed members how to set up their graphical interface in order to be ready to program the mBots. This instruction could have been done live, but for future programming modules, students will have to set up their graphical interface each time they open the software. There is not an option to modify or customize the opening layout of the software. The screencast link has been put on their Haiku page for the members to access at any time. Haiku is the learning management system that Granville County Schools is currently using. A course site has been set up for the Science Club and all active members have been enrolled in that site. Before moving on to the programming instruction, Dr. Batten and I walked around to make sure each pair had their graphical interface set up correctly.

After customizing the graphical interface of the software and checking to make sure everyone was on the same page, I showed the members my [second screencast](#). The second screencast shows members how to drag and drop commands to program their robots. There are many options, but when you start with the mBot program command, all but three of the programming menu options are made unavailable because they do not work with the robot. In the screencast, I explained what each of the drag and drop features that I used were for. I showed members how to build the program by dragging and dropping the individual commands and connecting them to one another. I did not test the program in the screencast as I chose to do that during the live instruction. After

the screencast is over, I showed the students how to start the program. My mBot then executed the program how I built it. I feel that the descriptions that I gave during the screencast were a good enough explanation that the students could imagine what it should do. I also chose not to show each individual command executed by the robot as I was building the program because I feared I would lose the attention of my audience. While viewing the screencast, the members were tasked with filling out a flow chart to assist them with building their own program with their mBot partner. The members had to correctly fill out the flow chart before they could interact with their mBots and the programming software. A highlight of future programming modules was presented in order to motivate the members to complete the current challenge so that they could move on to future challenges. These additional programming modules were also presented to keep the attention and excitement of the members. The last forty-five minutes of the Science Club meeting were used for the members to familiarize themselves with creating programs and using the drag and drop commands. A challenge was given to the members to complete before the end of the meeting. The members have to complete this challenge before they can move on to more involved and difficult challenges with new drag and drop commands. The challenge given to the members was to make the mBots move forward for a certain amount of time, stop, and move backwards for a certain amount of time. They had to use three different speeds and three different time intervals. Stop was one of the speeds they had to use. Dr. Batten and I both walked around the room and quizzed the members on the different commands that were discussed in the instruction.



**Evaluation.** The instruction was implemented in the media center at G.C. Hawley Middle School. This is also where the evaluations took place. This location was chosen instead of Dr. Batten’s classroom where the Science Club normally meets to allow as much space as possible for the mBots to move. There were 14 members of the Science Club present that participated in the activity. Observations and interviews were conducted throughout the instruction to monitor and record the evaluation. The members were asked what they liked about the instruction and screencasts. A plus delta chart was created. The answers are:

	
Good pace, instruction did not move too quickly for listening.	The pace of the screencasts is too fast for some members to take notes.
Instruction only focused on basic information needed for the day’s task.	Remind members that they can watch the screencasts again and pause and rewind if needed.
Members liked how the key menus were repeated at the end of the menu overview screencast.	

Dr. Batten and I walked around the media center and observed the member pairs building their programs and completing their challenges. The successful completion of the challenge is the instrument we used to collect the evaluation data. All 7 pairs of members completed the challenge and instructional goals with 100% success.

In reflecting upon these results, I realize that there are improvements that need to be made. The entire Science Club and robotics experiments are all new for this school year. The programming language of the mBots is new to me as well. Many of the members have experience with the Lego Mindstorm robots,

which use a different programming language that many felt is more user friendly. The reason I chose to create screencasts is so that the students could go back and view them if they needed to. I wanted the members to watch the screencast initially all the way through without pausing or rewinding it. Then, if needed, the screencasts were made available to them. I was actually surprised that so many of the members were able to figure out the "repeat" command so easily.

### **Recommendations**

1. Ask more detailed questions in the learner analysis. Even though some members had robotics experience, that question was not detailed enough. I should ask what interface/programming language experience they have in robotics as Scratch/Arduino that the mBots use is completely different than most other programming languages the members have been exposed to. It created a learning curve for all.
2. Do not let members have access to the mBots until after the instruction is completed. Access to the mBots provided a huge distraction from the instruction.
3. Have members close their computers during both screencasts so that all of their attention is on the screencast and not the programming software.
4. Give a brief overview of flowcharts and their symbols. The members were not familiar with what symbols to use for what action or process.
5. Another recommendation is to create a troubleshooting chart in regards to connection problems and movement problems. The mBots seemed to steer off to the left or right and not go in a straight line. Part of the

- problem was due to the wheels not being on securely. Occasionally the mBots did not respond wirelessly, and the wireless dongle would need to be re-seated and the software re-connected.
6. A final recommendation is to explore ways to move the mBots at different angles. Many of the members felt that my additional “repeat” command in the challenge was too easy.