Problem Statement

Soo-Min Yoo - CS461 Fall 2017

Our project is about the creation of a learn-to-code kiosk app that uses block-based logic and simple user interface design that makes it easy for kids to learn basic coding and STEAM (Science, Technology, Engineering, Art and Design, Math) skills. The goal is to get kids interested and excited about computer programming through game development. Our challenge is to make the learning process fun to motivate kids' imagination and desire to learn more programming. By doing so, kids will have an opportunity to try out coding in a creative and entertaining way, while learning skills that are becoming increasingly valuable in this fast-growing digital age. This challenge can be overcome by creating a kiosk app with a user interface similar to Scratch or Stencyl that teaches basic coding through simple game development. This will give kids an enjoyable experience in learning computer programming, and give them the motivation to further advance their acquired skills and create awesome games while they're at it. Also part of our project is programming natural interactions such as drag-and-drop, right-click, and double-click; creating an in-built library of background images, sprites, and sound effects; and the ability to save creations to the cloud so kids could take their creations with them. These features and functionalities will be displayed on the interface of the kiosk app.

The problem we are trying to solve is how to make computer science feel less scary or confusing to learn for beginners or those with no experience, and turn it into an easy and fun experience that will get kids excited and fired up to learn how to code. Computer science can feel intimidating and difficult to learn or try out for the first time, and many kids often get discouraged or feel afraid of pursuing a hobby or career in the field. In today's fast-advancing digital age, it's becoming more and more important for future generations to understand how technology works underneath the surface, and it's essential to get kids interested in developing coding skills, even if it's just the basics.

We could solve this problem by building an app that makes learning how to code a memorable and fun experience for kids, by using something many kids love - games! Game development involves not just programming but many other skills including story writing, music, and art. This means game development encourages involvement of kids from a wide range of interests and skillsets, and is also a great way to give kids who were originally interested in non-programming skills a fun introduction and opportunity to try out computer programming for the first time. By using block-based logic, this learn-to-code kiosk app will provide a simple and easy way to ease into the concepts of basic computer programming.

We will know when we have completed the project once we have a working prototype kiosk app, with a fully designed user interface that is similar to Scratch or Stencyl, with no system menus and no ability to exit the app or modify the computer. The app will get the user started in either creating a game (walking them through an optional tutorial if they choose), or opening an existing game. Once the user clicks an option to create a new game, they will be able to customize some options by giving their game a name, determining its screen size, and what platforms they will publish the game to.

Once the game is created, the user will have access to a home workspace with a sidebar listing all the game's resources, such as sprites, graphics, sounds, and game logic. The user will then be able to make new resources, open existing ones, or import external ones. The app will have its own in-built default library of background images, sprites, and sound effects. It will also have resource editors that will allow the user to customize sprites' appearance, behavior, and physical properties. In addition to the game resources, the user will have access to various game logic and player interaction that makes their game really come to life. The user can design their own game logic in a design mode that uses special "code blocks" that they can drag and drop into their desired order and function. Sprites can have specific behavior attributed to them that the user can configure to their liking. These sprites can then be placed into scenes (a.k.a. game levels) that they will interact in. Scenes can be designed in a scene designer, which looks like and functions similarly to a 2D art program. After the user finishes creating their game, they will be able to test their game, as well as save their creation to the cloud so they could take their creation with them. If we can, we could also have the app provide more tools that can let multiple kids collaborate on a game at the same time.