

Let's Dance: Teaching Introductory CS Programming Concepts through Dance

Hanan Al Nizami, Jasmine Jackson, Dr. Tiffany Barnes

haalnizami@student.yzu.edu, jasminej@gmail.com, tbarnes2@uncc.edu@mail.uncc.edu



Hanan Al Nizami

Youngstown State University



Jasmine Jackson

Winthrop University

Introduction

The Dance Tool that is designed to teach children of various ages simple concepts of computer science (CS) while having a good time. This web based tool provides an interactive applet in which children can manipulate 3-Dimensional scripting figures to perform dance movements in unique sequences and share them with a community.

The main goals of the project are to provide children with a better understanding of CS concepts while leaving a good impression

Background

It has been noticed that students lose interest in CS in their middle school and high school years, a problem that has reduced the number of enrollments in CS majors. It has also been shown that certain stereotypes are commonly associated with computer scientists, such as “nerds” and “geeks”, a reason for African Americans and some students of minorities in specific to shy away from approaching the field.

The idea of the dance tool developed as

a part of CSDT. CSDT's main focus was to teach students mathematical principles while creating simulations of cultural art. Also, Scratch is another inspiration to our tool. It is a Tile-based visual programming environment and toolkit that allow children to make games, animated stories, interactive art, and share with others on the Net while having a good time.

Research

1. Getting Started

First, we got around getting familiar with Java. We started out by testing Java's ability to run some of the tasks carried out by the Dance Tool to achieve the primary goals. Our tool contained functionalities such as adding 3-Dimensional objects to a canvas (the environment in which the 3-D dancers

and objects will be added into), rendering the objects, adding drag and drop functionalities to buttons that will contain pre-scripted functionality that will affect the 3-D figures. We broke down the tasks to the following:

- Making a simple program that can be loaded to the web since our tool is going to be web based.
- Building an interface that contains a canvas then coding it to have the ability to contain 3-D objects within it.
- Rendering 3-D objects in a canvas.

- Applying drop functionality to JPanels.

In order to achieve that, we had to do the following: 1) Plan little tasks that contain some functionality; as we needed to test the ability to script the functionalities we require for our dance tool in Java. 2) Research and implement code to those assignments and make them work.

Our first task was to test Java's ability to upload to the web. We created a simple calculator in NetBeans then we tried uploading the application to the web. Fortunately, that was an easy task.

Then, we had to work on building an interface that contained a canvas. That canvas should be able to paint 3-D objects within in. We created a 3-D cubical shape in the canvas. Then, we added swing components by which the cube was rendered.

Then, we created a handler to make it possible for the panel to accept drag-enabled buttons. This task took the most time researching possibilities of making it work.

After that we created the primary design of our Design tool. We started to apply the previous code to the new interface. The tool is not fully functional yet but it holds the main functionalities for the future dance tool. (See below).



Conclusions

The Dance Tool is an effective tool to teach CS to younger generations. The more exposure students can experience towards the CS fields, the better the chances are for them to be admitted to the program and stick with it.

Future Work

In the future, the user will have more freedom selecting a dancer of their choice. They will be able to chose a stage or background and insert 3-D objects to it. Buttons in all the tabs of panel 1 will also carry a functionality that will affect the 3-d figure in the canvas directly. A user will be able to use the buttons new, open, save and publish successfully. The timeline will be an interactive one to which all events will appear in different colors. Clicking on one timeline events will highlight the corresponding note in panel 2. A clue featured in a little pop up note will appear by hovering over a button to access a description of the functionality and how it can contribute to the existing script.

ACKNOWLEDGMENTS

Our thanks to the CRA Distributed, Dr. Tiffany Barnes for mentoring the project, and Eve Powell for supervising.