

Drill Design Suite

Neil Nordhof

Gene Fisher, Advisor

Computer Science Department

College of Engineering

California Polytechnic State University

June 16, 2015

Contents

1	Introduction	1
1.1	The Problem	1
1.2	The Solution	2
1.3	Outline	2
2	Software Use	3
2.1	Adding and Moving Performers	3
2.2	Adding Sets and Playing Drill	4
2.3	The Line Tool	6
3	Software Architecture	10
4	HCI Design and Implementation	11
5	Persistent Show Storage	12
6	User Feedback	14
7	Program Testing Design	14
8	Related Work	15
8.1	Pyware3D (http://www.pyware.com/)	15
8.2	Micro Marching League (http://micromarching.com/)	15
9	Conclusion	16
9.1	Future Work	16

1 Introduction

Drill writing software is used by marching band and drum corps programs to create the visual portion of their shows. The software allows instructors to move performers around to set their dots (or a coordinate on the field) for each set (or formation) in a show. These dots can be printed onto dot sheets, along with information about how many steps to take from set to set, and given to performers so they can learn their portion of the show.

Drill Design Suite implements this system by providing an intuitive approach to moving performers by hand, and supplements that with easy to use but powerful tools to create commonly used shapes and formations. It is implemented using the JavaFX library and its powerful UI components.

1.1 The Problem

Many already commercially available drill writing software are very expensive. One of the most popular programs can run over \$1000, while other less popular programs can still cost music programs hundreds of dollars. With all of the budget cuts that music programs are dealing with, a cost efficient alternative is much needed. There is then the issue that a lot of the cheaper or free programs do not include all of the necessary requirements, or are not created with designing for actual marching use in mind.

There are also many features in the higher priced programs that can add to the cost. Some of them offer 3D performance views of the drill, which will allow the user to watch their show be played out with 3D models. This also includes various poses the the drill writer can put the performers in. Certain software also have controls that are not entirely intuitive, but once learned, are very powerful to use.

A lot of the free applications are not designed for the purpose of being used in an actual marching band program, so they often lack many features that drill writing software have. Some of them do not have the ability to print out the drill for on field use. They also have very loose structure to tempo and count control, or none at all. This software can also be designed with doing virtual performances, where the main venue for showcasing drill is on the Internet.

1.2 The Solution

Drill Design Suite aims to strike a balance between offering many of the essential features of the high-end software, but maintain the accessibility of the cheap or free software. This requires omitting the fancy but bloated feature of 3D simulation of the shows. The process of making, texturing, rigging, and animating models is very time consuming, and is ultimately a lot of work that isn't strictly necessary for creating good drill efficiently. There is also the issue of accessibility, requiring controls that are easy to use and master. Creating a tool set that does what the user wants with each tool is important; they valuable shortcuts that allow the user to create the forms they need quickly.

1.3 Outline

The rest of this report will go into detail on the features that have been implemented in Drill Design Suite. Most of the main features will be demonstrated through a series of in-software examples. Some of these features will be further explained and detailed in later sections. There will also be sections on testing and feedback, and a look into where the project stands now and its future.

2 Software Use

When the user first loads up the application, they are greeted with a blank field to work on, as seen in Figure 1.

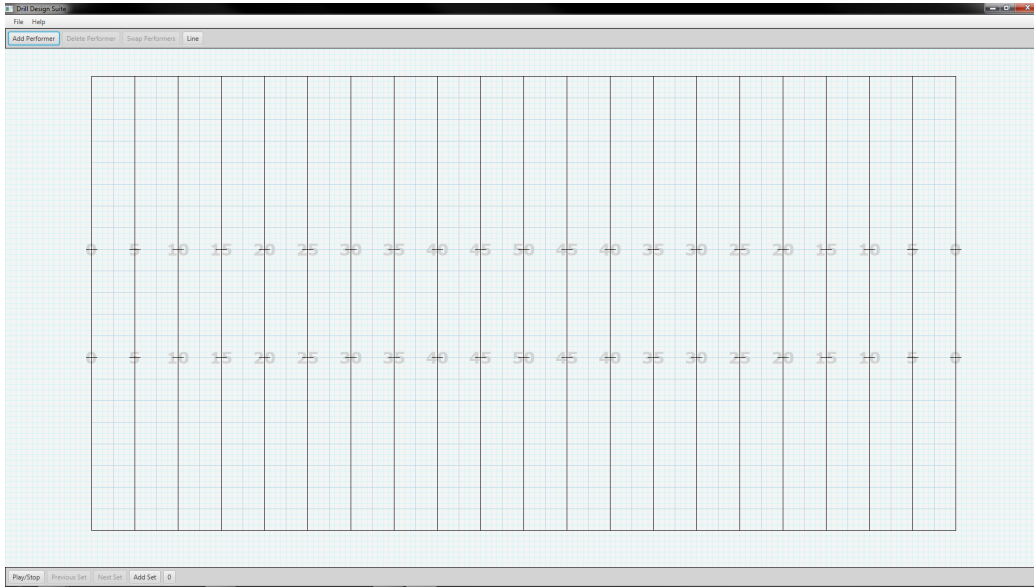


Figure 1: Application when first loaded

2.1 Adding and Moving Performers

The user can use the 'Add Performer' button at the top left of the screen to add a performer to the field, as seen in Figure 2. This performer can be clicked and dragged to change their location on the field, with their position being locked to the grid on the field.

Multiple performers can be placed on the field, and can be selected with standard methods. The user can drag-select multiple performers, and the order of this is determined by when the selection box contains the performer (this order is important for Section 2.3). Multiple performers can also be selected by shift and control clicking performers; shift clicking will only add performers, while control can both add and remove performers to the selection. The user can drag a whole selection of performers together for faster positioning.

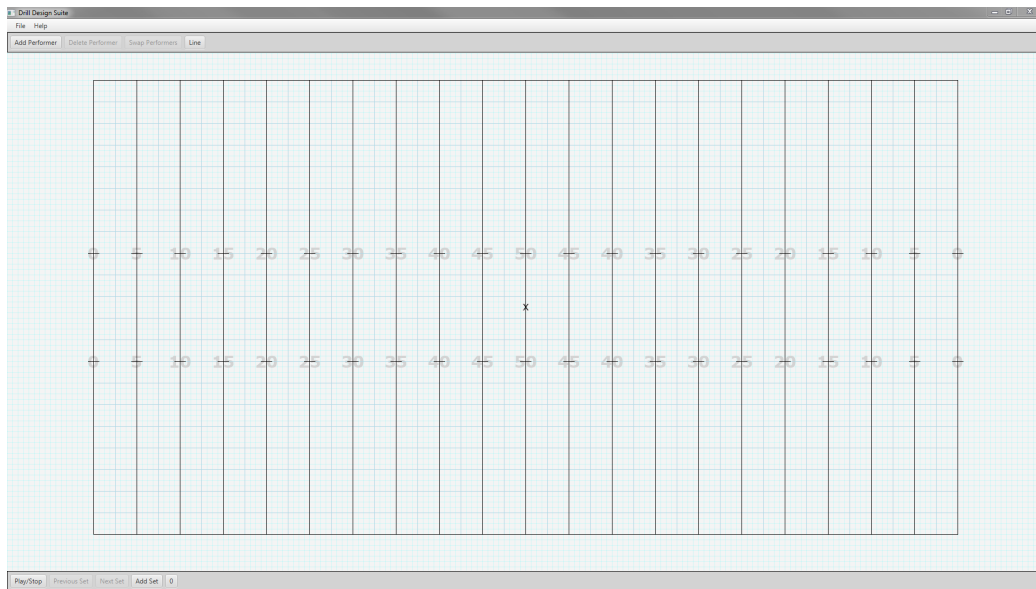


Figure 2: A performer that has just been added to the field.

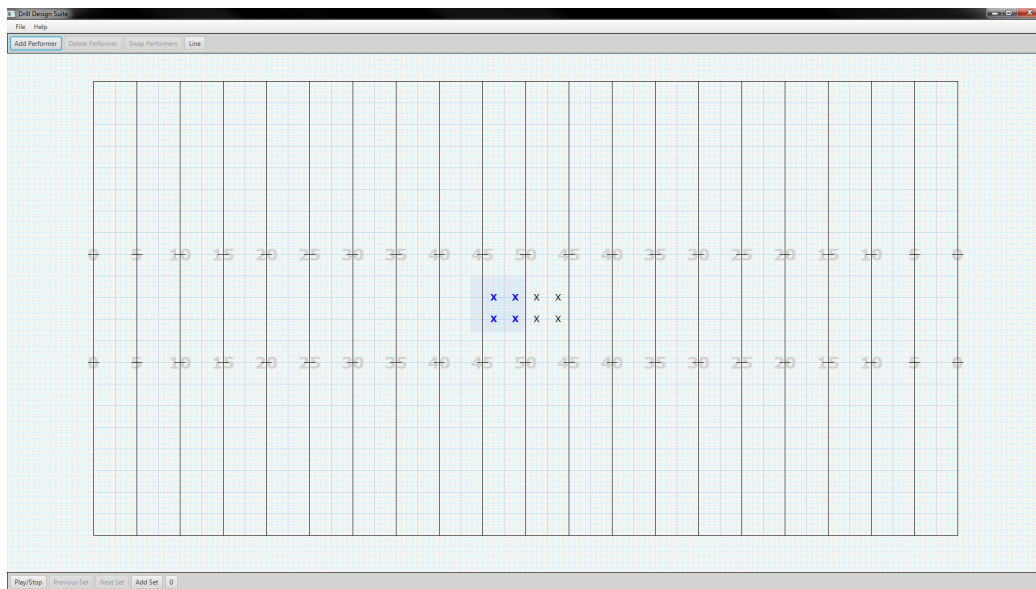


Figure 3: Performers being selected with drag select

2.2 Adding Sets and Playing Drill

Once the user is satisfied with their first set of the show, they can add another set using the "Add Set" button in the bottom left corner of the screen. For

each set added, a set button will appear on the bottom bar, which will allow the user to quickly jump to that set. They can also navigate through sets by using the "Previous Set" and "Next Set" buttons.

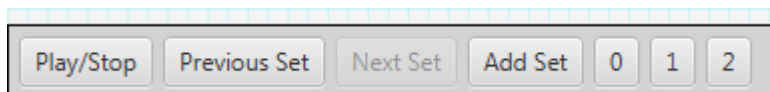


Figure 4: The bottom bar

When the current set is not the first set, performers will have a trail on them that connects them to their previous dot (see Figure 5). This allows the user to see the general path of performers, and to make sure that each performer is going to the right position. It can be a quick way to also look for potential collisions, but it is hard to judge if a collision will happen since performers' paths may cross at different counts within a move.

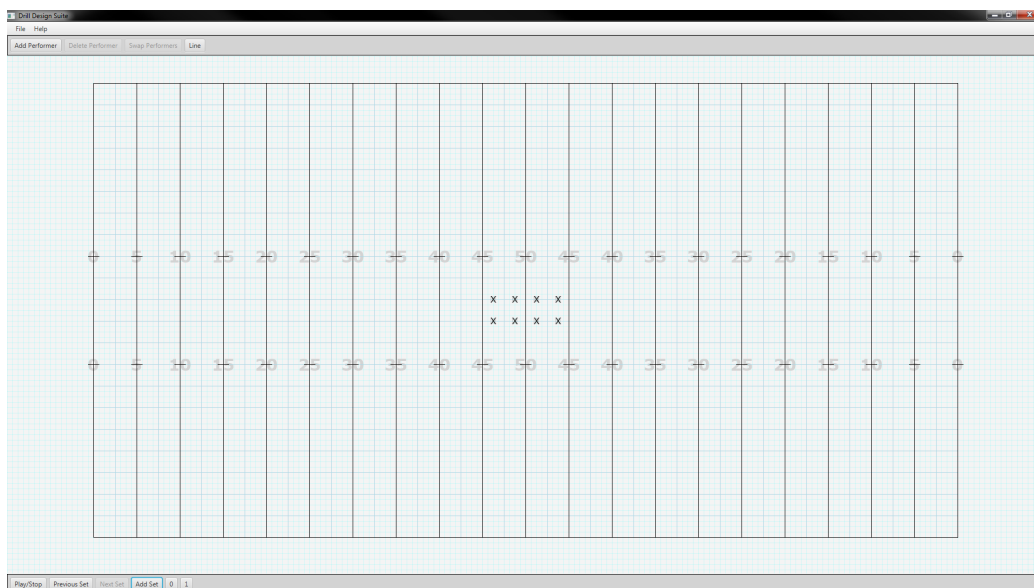


Figure 5: Performers gain 'tails' when the current set is not the first set

An easier way to find collisions is to use the "Play/Stop" button on the bottom bar. This will animate all of the performers as they move from set to set, starting on whatever set is currently displayed. This makes it easier to spot collisions since the user can see where each performer is throughout a move, and also allows the user to see how each move will flow.

2.3 The Line Tool

The first of the tools implemented in Drill Design Suite is the Line Tool. This tool allows the user to select a group of performers, and make a line out of them. This tool has two modes for use, auto and manual mode, with each mode having their uses in different situations

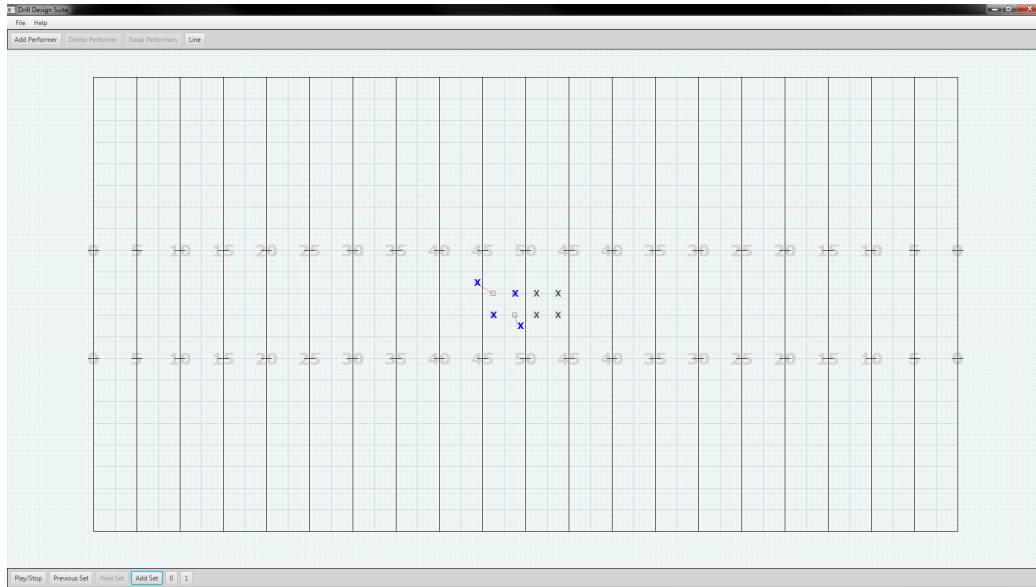


Figure 6: The selected performers positioned for use in the line tool

The auto mode for the line tool works by using the coordinates of the first and last performers in the selected group as the end points of the line. The rest of the performers in the selection are then evenly spaced in between the endpoints, based off of the order they were selected in. As seen in Figure 6, the four selected performers are going to be used to make a line. The performers were selected from left to right order, so the top left performer is the first selected, with the bottom right corner being the last selected. With more than two performers selected, the user can use the line tool by pressing the line button on the toolbar, which will open the prompt seen in Figure 7.

From this prompt, the user can change the mode by checking the 'Auto line' box at the top of the prompt. The Spacing and Angle fields can be left blank for auto mode, and we will go into more detail on those later. The user can then press select, and the performers will be made into a line, as seen in Figure 8.

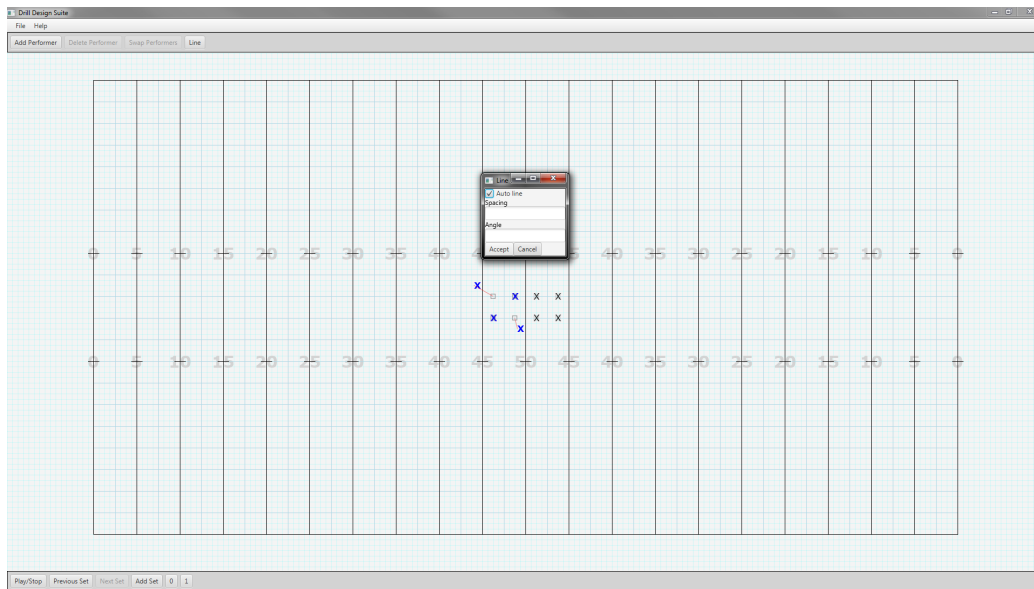


Figure 7: Auto line mode in the line tool prompt

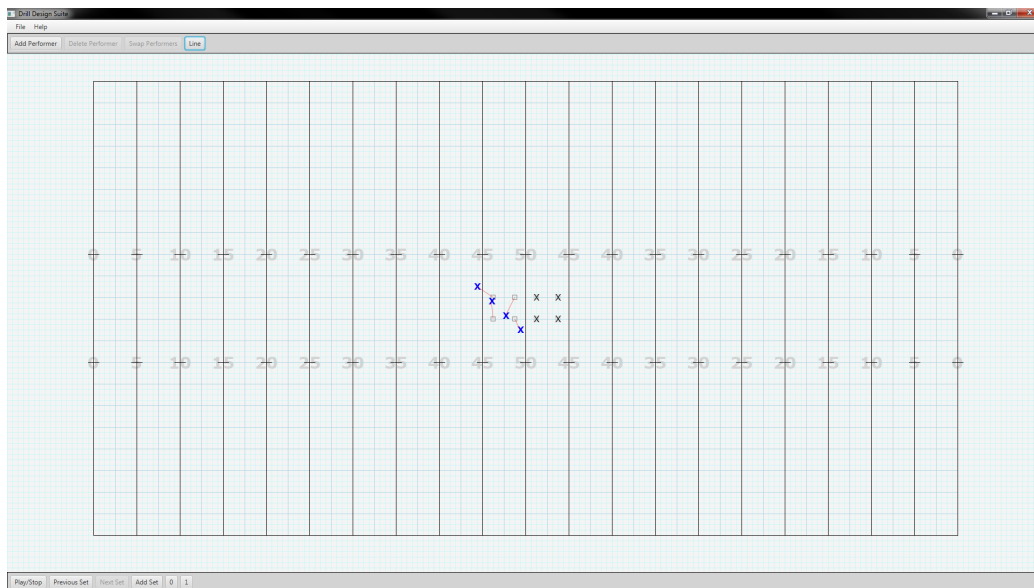


Figure 8: The performers are now positioned in a line

The other mode for the line tool is manual mode. In this mode, the user

specifies a spacing and the angle for the performers. Then the position of the first performer in the selection is used for the starting point, and the other performers are positioned from there. In Figure 9, the four selected performers are set up to make another line, with the bottom right performer being the first selected. Instead of checking the 'Auto line' box in the line tool prompt, the user will instead want to fill out the spacing and angle fields. The spacing is measured in steps (1 space on the grid is 1 step), and the angle is degrees, going counter-clockwise with 0 degrees pointing to the right. Pressing accept will position the performers in a 30 degree line with 4 step spacing, as seen in Figure 10

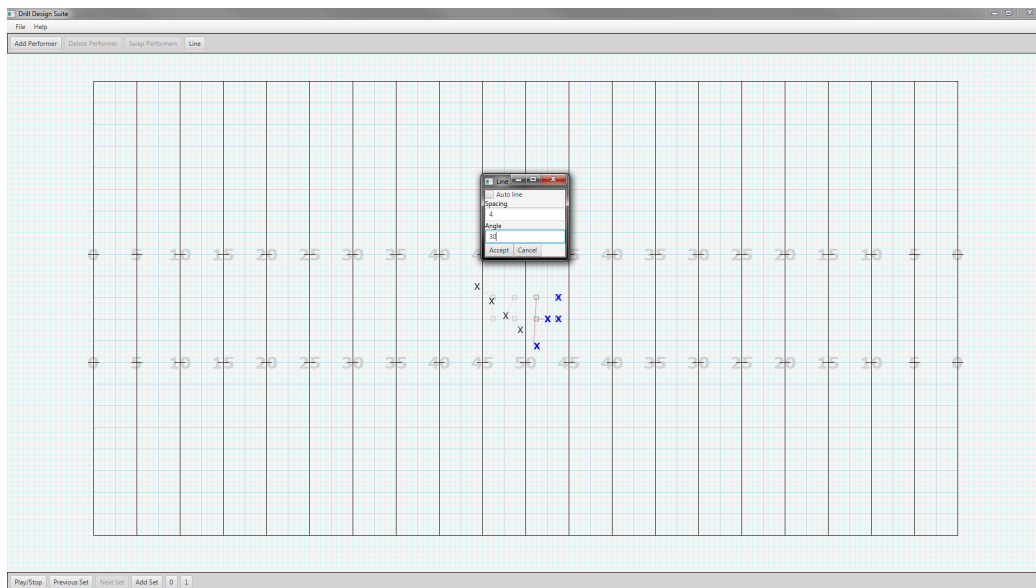


Figure 9: Performers positioned for another line, with the manual mode for the line tool prompt filled out

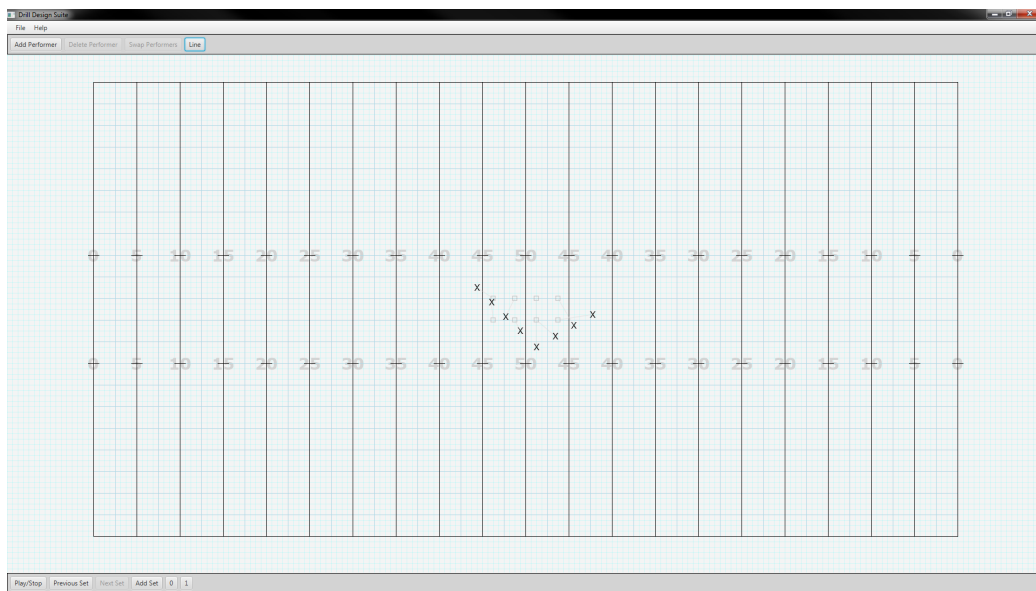


Figure 10: The performers are now positioned in another line

3 Software Architecture

The structure of Drill Design Suite’s architecture is fairly straightforward. The Performer class extends the text class to make displaying the performers on the field simpler. It will also help in the future when a display is made to list the performers, both selected and unselected, and any order in them. Performer also has a list of Coordinates to keep track of their dots. The Move class contains information about the number of steps and the tempo of a move. SetButton extends JavaFX’s button class to allow more functionality for keeping track of the set and for future coloring purposes. Main’s aggregation of Performer, Move, and SetButton are fairly simple. There is a list for all of the Performers, along with selected performers. There is a list to keep track of each Move, and the list of Setbuttons grows as more Moves are added. DrawTools depends on Main to send it information about which Performers to make a line out of.

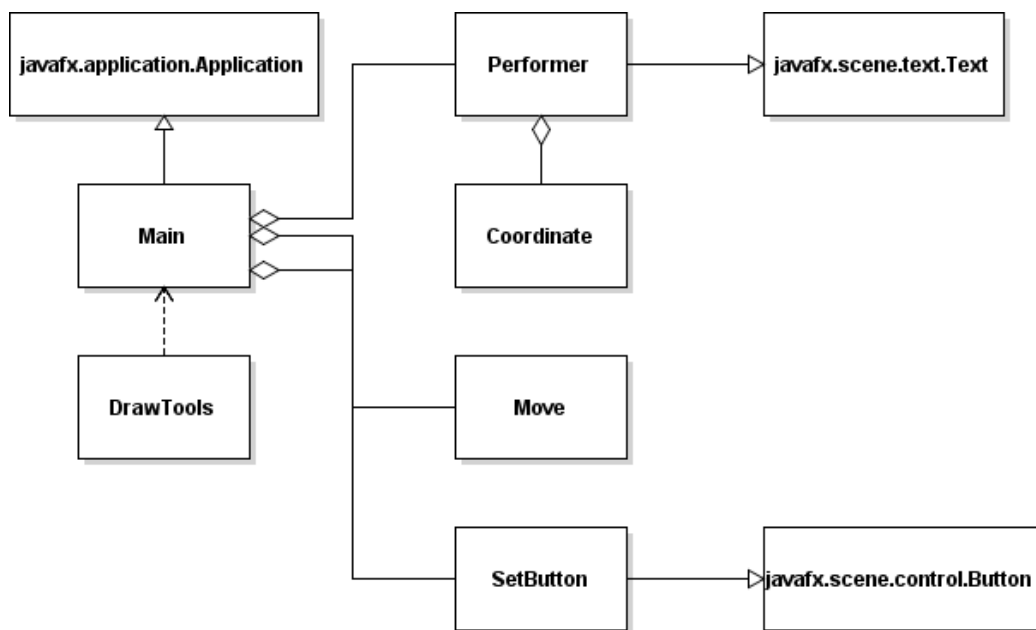


Figure 11: UML Diagram of Drill Design Suite classes

4 HCI Design and Implementation

The interface for Design Drill Suites has 3 main areas. The toolbar at the top of the screen and the time line bar at the bottom are HBoxes with their various buttons in them. Many of these buttons are dynamic, and have to listen for changes in the number of performers selected or what the current set is to determine whether or not they are clickable.

The center portion containing the actual working area is the most interesting. There is a JavaFX Group called Field, that contains a JavaFX Canvas to draw the actual field, and all Performers. The field was drawn programmatically, which allows it to adapt to various screen resolutions and zoom levels. The location of the Performers is determined by using an offset of the top left corner of the field in the canvas, and then converting their coordinate to an absolute format. The offset and absolute coordinate are combined to determine where the Performer is actually positioned on the screen.

5 Persistent Show Storage

One of the most requested features early on in testing was the ability to save shows. This feature is very import for drill creation, since writing a show can take dozens of hours to do, especially for larger ensembles. The user can save their work by using the save option in the file menu at the top of the screen. They can also open a previously saved show from here, as seen in Figure 12, and ecreate a new show from here too. The user can also access these commands using their respective keyboard shortcuts. Currently, only one show can be saved in the save location at a time, but the user could potentially circumvent this by moving the file out of the save location if they want multiple shows saved. Future plans involve using the built-in file chooser to make have multiple show saves a simple process.

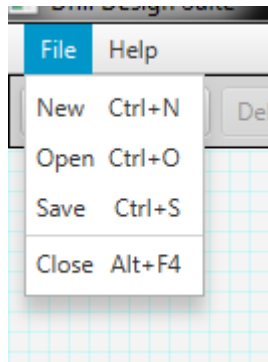


Figure 12: The File menu contents.

The save file is stored as a .txt file, and all of the information is human readable, as shown in Figure 5. The show saves by first going through all of the moves, and storing the number of counts and the tempo for the move. Once all of the moves are written, the Performers are saved next. Each Performer has its id and label stored, and then goes through each coordinate to store those. Both sections are headed by a line lisintg the number of moves (NS) and the number of performers (NP).

```

NS=2
  8,120
  8,120
NP=4
  PId=0,PSym=X0
    76.0 , 44.0
    74.0 , 42.0
    72.0 , 39.0
  PId=1,PSym=X0
    84.0 , 44.0
    82.0 , 38.0
    78.98051240207859 , 39.558051789953005
  PId=2,PSym=X0
    76.0 , 36.0
    78.0 , 40.0
    75.9902562010393 , 36.2790258949765
  PId=3,PSym=X0
    84.0 , 36.0
    86.0 , 36.0
    88.97076860311789 , 35.837077684929504

```

Figure 13: The contents of a typical save file

6 User Feedback

Drill Design Suite went through one phase of user testing around the halfway point of the project. Most of the feedback was positive, with the largest demand being for persistent show storage. This issue was quickly addressed with the ability to save/load shows. Another issue was that Mac users were having difficulty running the program on their machines. This was due to the project being distributed as a .jar, so it would require command line arguments to run properly. There was an attempt at making an executable file, but due to Eclipse not being able to create a cross-platform executable, I instead had to make two different ones. Making a .exe for Windows users was easy, but due to me developing on Windows, I was unable to make a .DMG file for Mac users.

7 Program Testing Design

Writing systematic tests for the software isn't very easy to do in its current state, since many of the important functions have visual results that are hard to measure with numbers. As more tools are added to the program, it will be easier to write tests to make sure these functions are working properly. The current easiest way to test is to just create shows, with setting up formations to test these features.

8 Related Work

There are two major software that were used as reference in the creation of this project. They are Pyware3D, one of the leaders in drill writing software, and Micro Marching League, which is a web-based app that is based around online competitions of fictional marching groups.

8.1 Pyware3D (<http://www.pyware.com/>)

Pyware3D is one the most popular drill writing software, but is also the most expensive. A lot of the design decisions, especially visually, we influenced by this software. Pyware3D does a lot of things well, but also has the issue of having a bit of a learning curve. The controls aren't the most intuitive, but can be used very powerfully once they are mastered. It also has the ability to group performers in different ways for use with it's tool set. There is also a 3D portion of the software, where the user can watch their drill be acted out by 3d models, along with animations of marching and doing some basic poses. The user can also import an mp3 of source music to make it easier to see how the music lines up with their drill. And the final major feature is the ability to print out the drill, in both dot sheets for performers to use and drill sheets for staff to use to see the shapes the performers should be forming.

8.2 Micro Marching League (<http://micromarching.com/>)

Micro Marching League (MML) is a somewhat popular free app that runs in a browser, where users can create shows for others to watch and judge. The controls in MML are very easy to pickup, but are also a bit limited compared to Pyware. The tools are also less robust, and the shows are limited in how long they can be. There is also little control over tempo and counts, so it can be awkward to get sets to flow together without use of a 'metronome member'. This is because the length of a move is determined by the farthest distance a user has to travel. Having a member that only does massive moves off screen can help control the duration of moves. There is also only 2D sprite-based animation, and there is no ability to print the drill you have created.

9 Conclusion

Drill Design Suite aims to strike a balance between powerful and cost-effective drill writing software. While not all of the intended features are implemented in this version of the project, there are enough to lay the foundation for future development of the software, and potentially even a release for public use. I personally gained a lot of insight from this project, especially working with a UI Library I have no experience with. Although JavaFX is fairly familiar, there is still a lot to discover in the library that can make my software better. I also learned about how difficult it is to develop on multi-platform, and the extra effort it takes to release for users on both Mac and Windows.

9.1 Future Work

I do plan to continue working on this project, with an emphasis on expanding and refining the systems already in place. Many of the important features are already in Drill Design Suite, but the most lacking is the ability to print the drill out. This would be a priority for future work. Other plans involve setting up the software to be open source, and to start getting a team or community to continue developing the project. Once the project reaches an adequate state, reaching out to various marching ensembles to try out the software could be big for spreading the project amongst the marching community, and could also provide valuable feedback for further developments.