

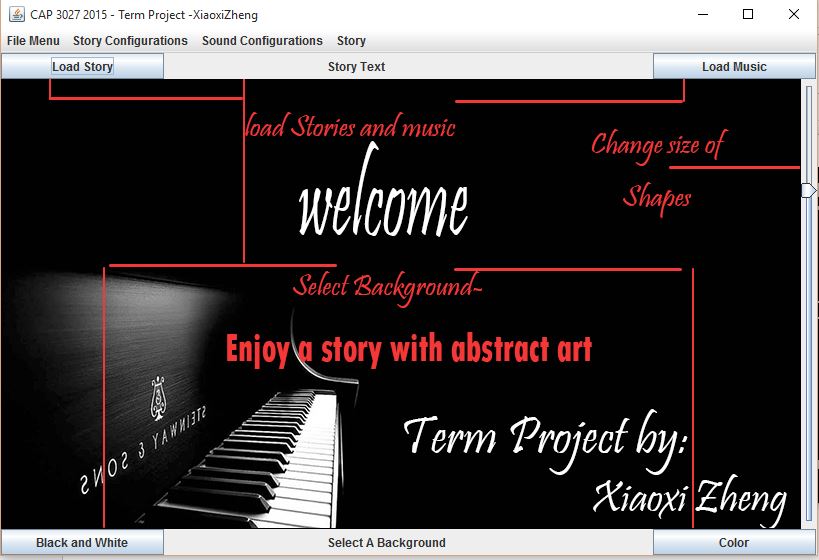
Xiaoxi Zheng | Cap3027 | December 8, 2015

User Manual

Iteractive art viewer

# Program Overview

The goals for my term project was to create an interactive abstract art viewer. Like a musical birthday cards, a large part of this project was concentrated on the music. I wanted to create a viewer that users can click on that reflected the corresponding line of text and music. The major functionality of this project was being able to create text stories line by line, and respectively being able to save and reload them back into the program. The same functionality applies to the music aspect of this program. I allowed users to “write” music in text input of music notes, instruments, and the tempo they’d want to use. When all properties are loaded along with the background, the screen will display an array of “Shapes” they could click on that correspond to the number of lines in the story, and upon clicking on those shapes, the story will begin with text, corresponding image assets and some background music.



# Compiling and running the program

To compile and run with jFugue(Mandatory):

Compile: javac -cp .:jFugue.jar storyTelling.java

Run: java -cp .;jFugue.jar storytelling

Note: I been compiling jFugue on cmd and can’t get jFugue to properly work on Windows PowerShell.

# Configuration:

Please follow this configuration so the program can run most smoothly. Please finish setting up all the story line, music, and background before triggering to have all the shapes to display. After everything is set up, trigger the program to draw all the shapes by selecting either [Story]→[Load Black or White Shapes] or [Story]→[Load Color Shapes].

Create, load, and save Story Line as .txt files -under the [Story configuration] on the menu bar, options to create, load and save story lines are available.

1. To [Create] a story line, user is first prompt for the number of lines they wish to input, and then follow by the contents they wish to input sequentially.
2. To [Save] a story line, [Save inputted story] is available and will save whatever user inputted so far as a text file.
3. To [Load] a story line, the program will expect any text file. The program will parse the documents line by line, and know the total number of lines it receive.

Create, load, and save Sound Track using jFugue library as .txt files-under the [Sound configuration] on the menu bar, options to create, load and save music are available.

1. To [Create] a music line, user is prompt for the following inputs:
   1. Instrument type, the program will accept string representation of instruments in the jFugue library.
      1. Sample input can be [Piano, 0 (int representation of piano), flute, guitar, rain and etc…please see the attachment to the end of this document for full list of possible inputs.
   2. Tempo
      1. Expects an int that represents the beat per minute you wish the music to be played.
   3. Music notes
      1. First expects the number of lines they wish to input, and then follow by the contents they wish to input sequentially.
         1. Note: the number of lines here should equal to the number of lines inputted when creating a story (So 1 line of music corresponds to 1 line of story text, if not - the less of the 2 number is picked to avoid crashing.)
         2. Sample music explained:
            1. Sample string: “Eq Ch. | Eq Ch. | Dq Eq Dq Cq V1 I[Flute] Rw | Rw | GmajQQQ CmajQ"
            2. '|' (pipe) for indicating measures (optional). To specify durations, use 'q' for quarter duration, 'qqq' for three quarter notes (multiple durations can be listed together), and 'h' for half, 'w' for whole, and '.' for a dotted duration; 'R' for rest, and the chords G-Major and C-Major. Whitespace is not significant and can be used for visually pleasing or helpful spacing.
2. To [Save] a User inputted Music, [Save inputted story] is available and will save whatever user inputted so far as a text file.
3. To [Load] a music, the program will expect any text file. The program will parse the documents line by line, and know the total number of lines it receives.

Select background to paint Black and white or Color

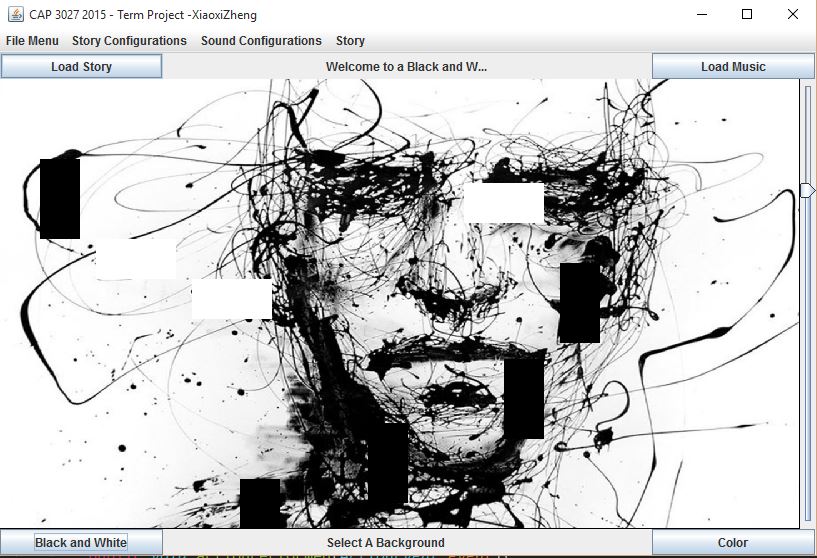
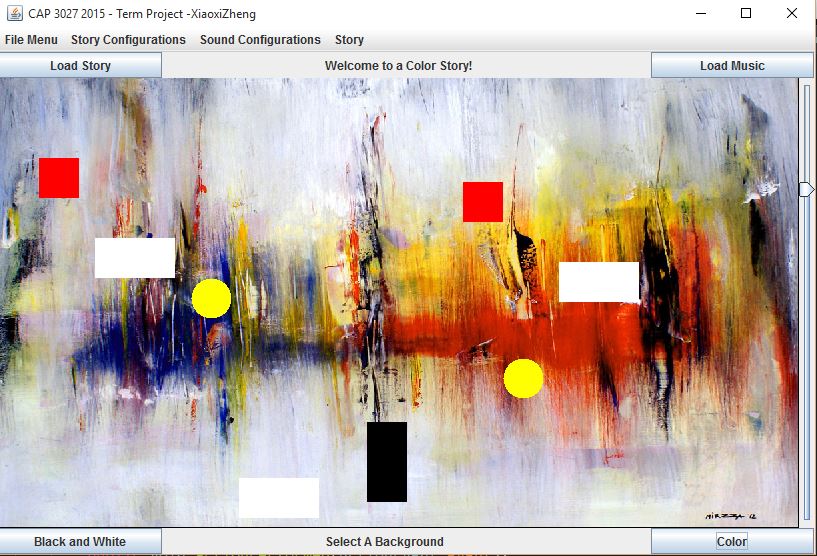
1. I have 2 pre-defined background images or the Abstract Art Pieces to be displayed when the program is at rest for receiving a mouseEvent. This needs to be set before “loading black or colored shapes”.

Change folder name to look for image assets

1 . This prompts users for the directory the program should go look in for the image assets triggered by mouseEvents. Please save the images assets in [num].jpg, where num is a placeholder for int starting at 0. If this never got selected, the default path is [xmas] inside the termProject folder.

# Run

After everything is set up, the trigger to view the interactive part of this program is by going to either [Story]→[Load Black or White Shapes] or [Story]→[Load Color Shapes]. Then, you can click on the individual shapes to reflect the corresponding storyline and music.



[Story]→[Load Black or White Shapes] [Story]→[Load Color Shapes].

These two are the default art pieces I decided to use. I purposely made the shapes drawn on the canvas a bit hard to distinguish from the background to give the illusion that it’s actually part of the piece. So now, from left to right, simply just click on the different shapes in sequence to view and hear back what you previously configured.

# Attachments

Instruments allowed:

