*Florida International University*

*School of Computing and Information Sciences*

CIS 4911 - Senior Capstone Project

Software Engineering Focus

Feature Document

User Story #698

**Team Member:**

Andrew Mitchell

**Product Owner(s)**:

Francisco R. Ortega

**Mentor(s)**:

Francisco R. Ortega

...

**Instructor**: Masoud Sadjadi

**User Story –** Implement Basic Text

* As a User I would like to write text into the program so I can have nice letters available when I want them.

Acceptance Criteria

* Create an On Screen Keyboard to type using the Multitouch.
* Must be able to designate where I want to write text.
* Must be able to drag on screen keyboard.

**Use Case**

Use Case - Type Text

User presses a text button and selects where to place text. On screen keyboard then shows up and can type with it.

Details:

Actor: User

Pre-conditions:

Program must be running.

Multitouch must be enabled.

Description:

Use case begins when the user presses the text button. The user must then choose where to type text. Then the user can use the on screen keyboard to type text. Touch away from the keyboard to exit typing mode (or press the keyboard ‘done’ button).

Post-conditions:

The text should type whatever the user types onto the on screen keyboard.

Decision Support:

Frequency: Medium. Being able to type is a feature that may be used to draw nice letters that the user cannot normally make legible with a multitouch or leap motion draw.

Criticality: High. Text support is one of the features the product owner really wants implemented.

Risk: High. A large moving button such as a keyboard may be hard to implement. There may be unexpected bugs that arise when dealing with such a large button. In addition ‘text’ is fairly new to the program, working with this new set of functions may lead to unexpected results.

Reliability: Highly.

Mean time to Failure – Almost Never. The keyboard should almost always properly detect a touch on the correct key.

Availability – Available with multitouch currently.

Performance:

Should have very low performance impact (Drawing one additional FBO), with a few additional checks.

Supportability:

Multitouch support.

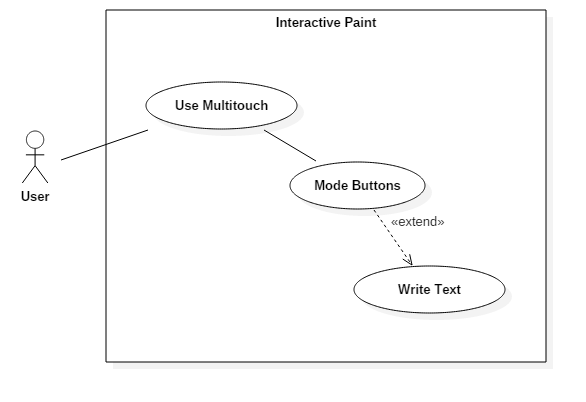
Modification History:

Owner: Andrew Mitchell

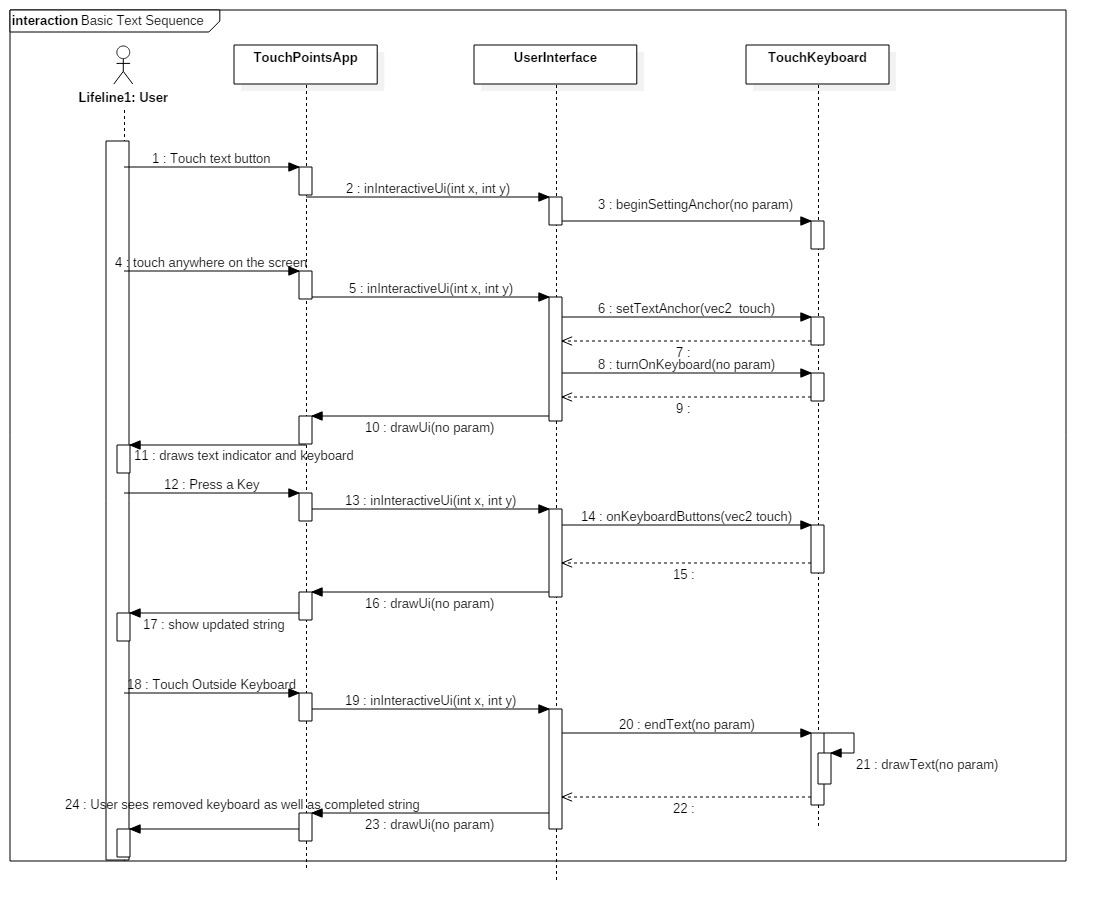
Initiation date: 4/11/2016

Date last modified: 4/15/2016

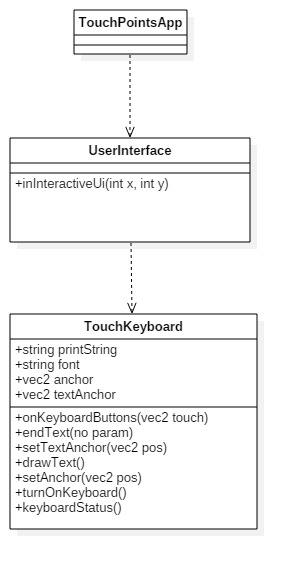
**Use Case Diagram**



**Sequence Diagram**



**Class Diagram**



**Unit Test**

Sunny Day:

Test Case: Keyboard Key Test

Test Purpose:

Ensure that all the keys function properly.

Test Setup

1. Press the ‘Text’ Button
2. Press a location on the screen to write text.
3. Press every button on the keyboard except Shift, Font, Enter, and ‘Done’. (Starting from the top left, continue all the way right. Then the next line starting from the left).
4. Press ‘Font’ four times to change through the fonts.
5. Press Shift.
6. Press Enter
7. Repeat step 3.
8. Press outside the keyboard to shut it off.

Test Output:

Pressing Font changed fonts

Pressing Shift changed some keys

Pressing Enter made the new typing happen on a new line.

The ending string was

“`123456789 qwertyuiop\asdfghjkl;zxcvbnm,./

~!@#$%^&\*( QWERTYUIOP|ASDFGHJKL:ZXCVBN<>?”

After tapping outside the keyboard it disappeared and I could draw again.

Expected Output:

Pressing font should cycle through the font.

Pressing shift should change the key types

Pressing Enter should do ‘New Line’

The resulting string should be

“`123456789 qwertyuiop\asdfghjkl;zxcvbnm,./

~!@#$%^&\*( QWERTYUIOP|ASDFGHJKL:ZXCVBN<>?”

Rainy Day Tests:

Test Case: Backspace on Enter

Test Purpose:

Ensure that backspace is working properly. It is properly deleting strings, as well as ‘Enter’ keys, and not breaking when ‘backspacing’ nothing.

Test Setup:

1. Press the Text Button
2. Press where you want the text
3. Type ‘asdf’
4. Press ‘Enter’
5. Type ‘qwer’
6. Backspace 4 times
7. Type ‘qwer’
8. Backspace 5 times
9. Type ‘qwer’
10. Backspace 12 times.

Test Output:

Step 3

“asdf”

Step 4

“asdf”

Step 5

“asdf

qwer”

Step 6

“asdf”

Step 7

“asdf”

qwer”

Step 8

“asdf”

Step 9

“asdfqwer”

Step 10

“”

Expected Output:

Step 3

“asdf”

Step 4

“asdf”

Step 5

“asdf

qwer”

Step 6

“asdf”

Step 7

“asdf”

qwer”

Step 8

“asdf”

Step 9

“asdfqwer”

Step 10

“”

Pressing backspace with an empty string should not crash the program.

**Integration Test**

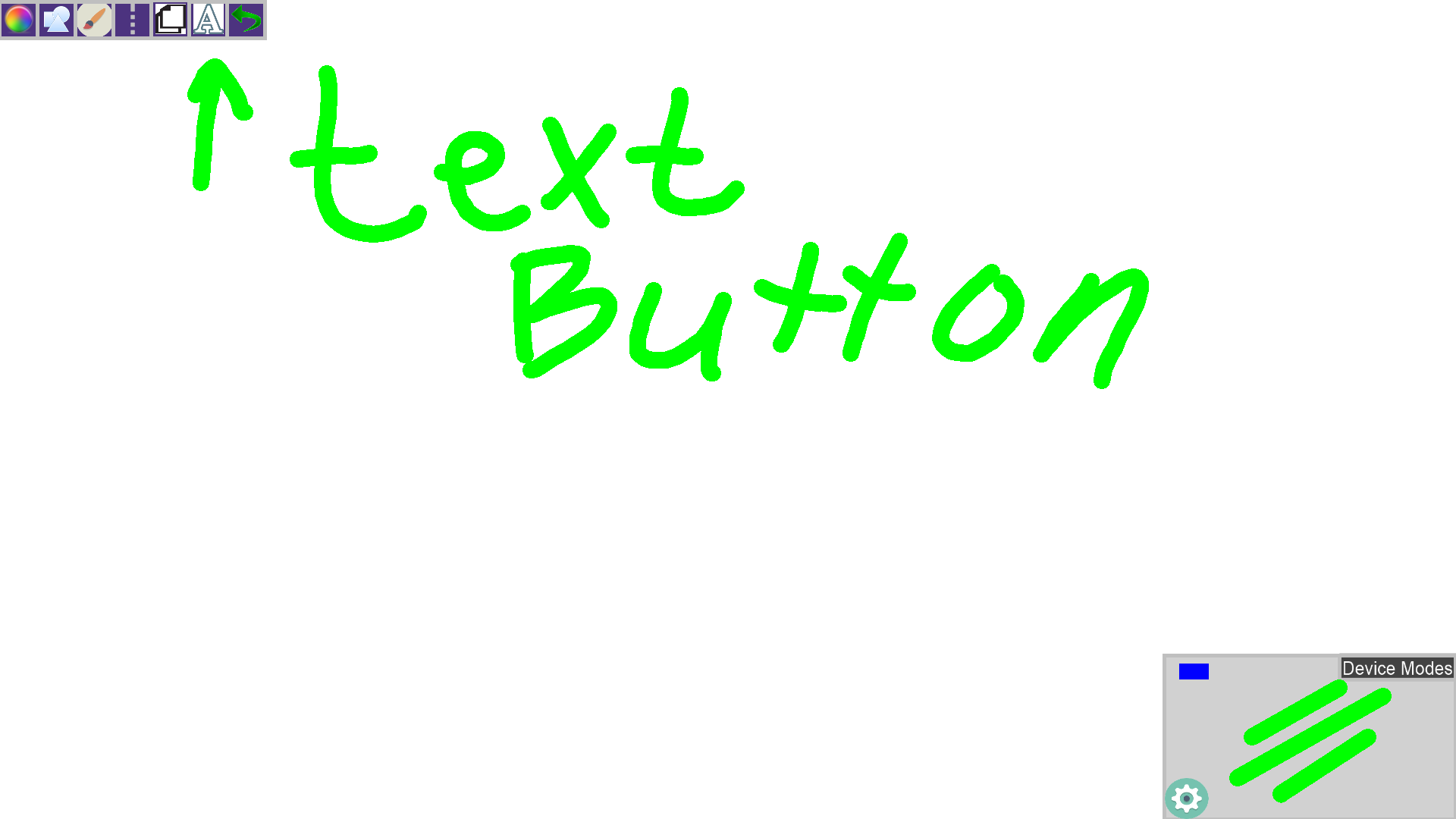
Activing the keyboard currently disables drawing with the multitouch, in a sense that the next tap away from the keyboard will turn it off instead of begin drawing. Drawing with the leap motion as well as gestures is still enabled though.

After pulling up the keyboard, and then moving it around and writing text and then shutting it down, we are still able to draw with the multitouch.

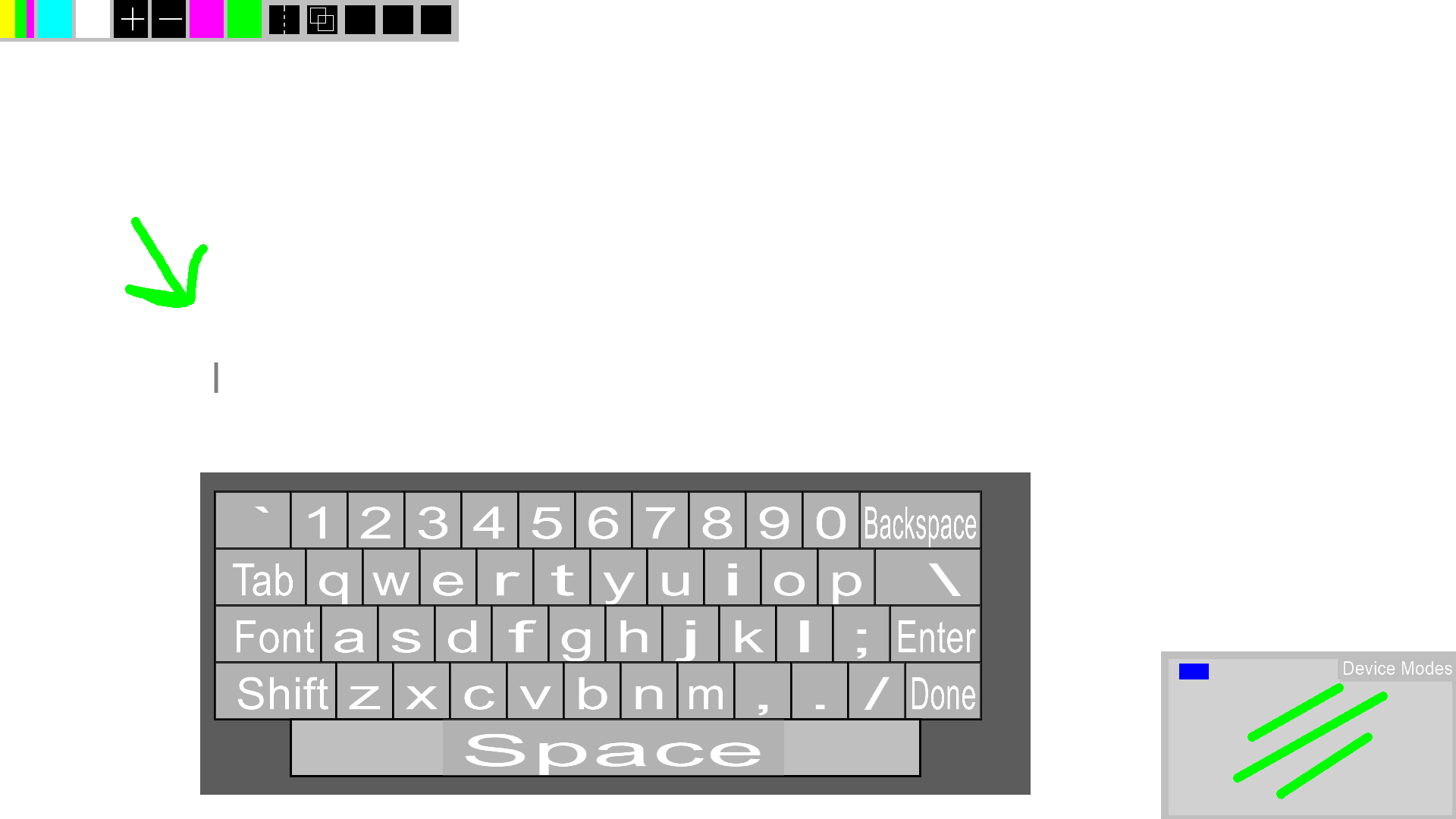
It is also integrated with illustrators ‘Undo’ functionality, allowing you to erase the last text you ‘wrote’ to the FBO.

**User Guide**

You can use the ‘Text’ button on the mode buttons (top left buttons) to make your next tap choose where you want to type text (No visual indication at the moment when you press the button).



After tapping the text button, tap the screen again at any location where you want to type your text. A Grey line will pop up indicating where the text will be typed.



You can then drag the keyboard around by grabbing the edges of the keyboard and moving your finger. You can type using the various keys as well as changing to upper case using shift, which also changes the numbers to symbols. You can also choose ‘Font’ to change to one of 4 available fonts. Press enter to start a new line right below your current line. Pressing anywhere on the screen that is not the keyboard (or pressing the ‘done’ button) will shut down the keyboard and save your text to the screen.



**Glossary**

On Screen Keyboard – A keyboard that appears on screen that allows you to use it using the multitouch device.