
SENIOR PROJECT COMP 188

CLASSY NOTES

3/21/2014

CASSIE MESTAYER

TABLE OF CONTENTS

- 3. Customer Statement of Requirements
- 4. Glossary of Terms
- 5. Functional Requirements Specification
- 6. Use Cases
- 8. System Sequence Diagrams
- 10. User Interface Design
- 11. User Effort Estimation
- 12. Plan of Work
- 13. References

CUSTOMER STATEMENT OF REQUIREMENTS

This project will be an android application that users will be able to download specifically on to a tablet. This application is intended to create an easy and more efficient way to take notes in a classroom or a fast-paced lecture environment. The user will be able to quickly open up their notes and insert in information quickly by typing, drawing, or even including pictures. The user will be able to save the notes in folders created by the user and will also be able to download them. The user will be able to download the files to cloud services, their computers, or even print them off from a printer. This project is intended to replace the use of a notebook to keeping everything neat in one portable tablet.

GLOSSARY OF TERMS

Stylus: A writing utensil. It is a computer accessory that is used to assist in navigating or providing more precision when using touchscreens.

FUNCTIONAL REQUIREMENTS SPECIFICATION

a. Stakeholders

This application is intended to be used by students with android tablets, but really is available for anyone who owns an android device. Someone interested in this application besides a student could possibly be business associates, researchers, or teachers.

b. Actors and Goals

Participating actors will be the user, who could be anyone, and they will directly interact with the application.

c. Use Cases

i. Casual Description

1. CreateNewDocument: When a user creates a new document they will be asked to input a name a save location for the new document. The app will then display a blank new page for the user on the screen.
2. OpenDocument: The user will click the 'Open Existing Document' button and select the document they wish to open.
3. Draw: The user will select the draw button and start drawing on the screen. The user can also change the draw features like size or color by selecting the corresponding button.
4. Type: The user will select the type button and be able to select a location on the page for a text curser to appear at. Once the user has selected a spot to type at, they can begin typing otherwise they can't start. The user will be able to change the text features, such as size, color, and font.
5. ExportToPDF: The user will be able to create a PDF of the document and export the document via cloud, saving to the device, or print.
6. TakeAPicture: The user will be able to take a picture with their devices camera or select one from their photo library. Once they have a photo to insert into the document they can select the location and rescale the picture to their liking.

ii. Fully-Dressed Description

1. Use Case: createNewDocument

Precondition: The *user* has selected the 'Create a Document Button' on the main menu.

Main flow of events: The use case starts when is prompted a name and save location for the newly created document. The *user* will type in the name of the document and then select a destination folder or create a folder for it to be placed in. The app will then display a blank page for the user.

Exceptional flow of events: The *user* enters in a name for a document that is the same name as a document already located in the destination folder. The app will then show an error and not let the *user* continue until they enter in a unique name for the document.

Exceptional flow of events: The *user* wants to create a folder for a document that has the same name as a folder that already exists. The app will then

show an error and not let the user continue until they enter in a unique name for the folder.

Postcondition: The app has created a new document with a unique title inside a unique folder.

2. Use Case: draw

Precondition: The *user* has a document open in the app.

Main flow of events: The use case starts when the *user* selects the draw button on the right of the screen activating the *draw* function. The button will display a small menu option to the side of it with different sizes the *user* can choose from. Once the user selects the size, the *selectDrawSize* function is activated. Then the *user* clicks on the color wheel button that activates the *changeColor* function and selects a color from the colors listed. The *user* then uses their finger or stylus and draws or handwrites notes. The use case is over when the user selects another button such as text.

Exceptional flow of events: The *user* does not select a size or color at the beginning of the draw function, the draw function is initialized to a default size or color.

Exceptional flow of events: The *user* has multiple fingers or items touching the screen at the same time. The app will accept only the first touch and not accept anything else until nothing is touching the screen anymore.

Exceptional flow of events: If the user draws something on the screen they would like to erase, they can select the eraser button to the right of the screen and then erase. Once the user is done they can either re-click the eraser button off or click the draw button again.

Postcondition: The app will stay in the draw function until another function is activated.

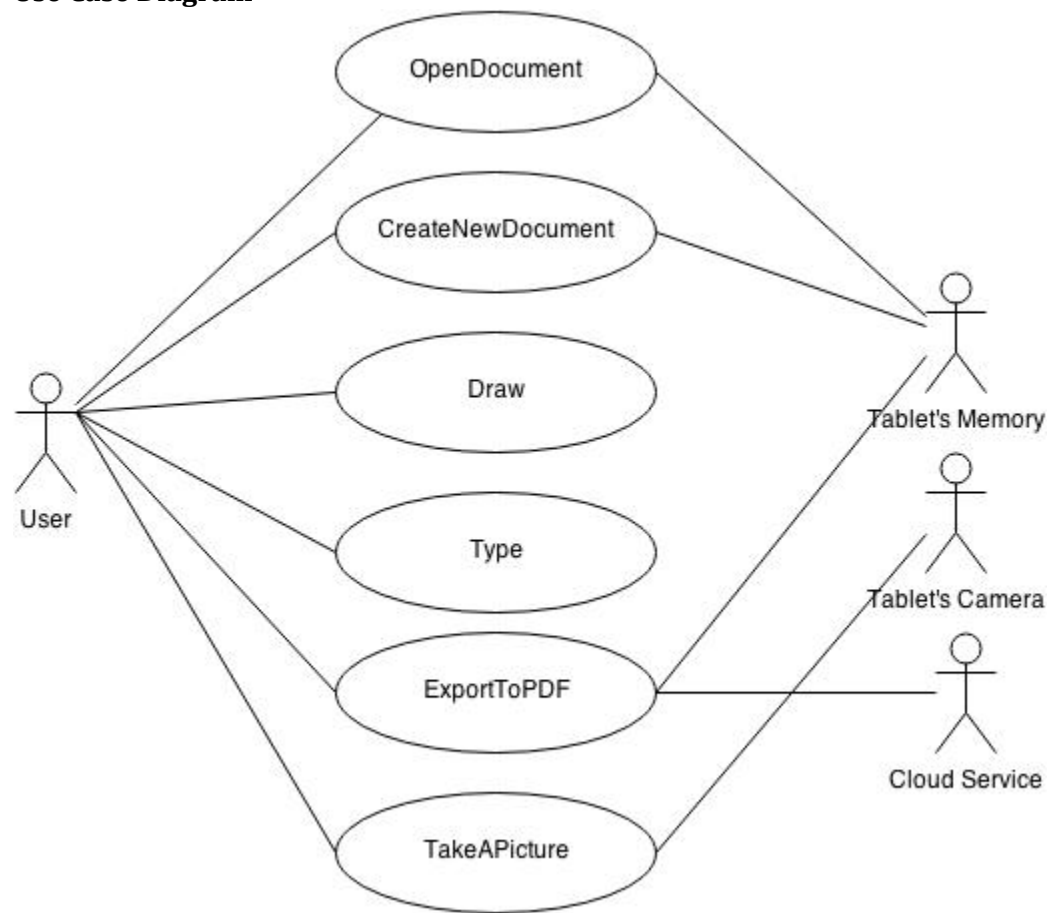
3. Use Case: Type

Precondition: The *user* has a document open in the app.

Main flow of events: The use case starts once the *user* has clicked on the type button. Once the type button is activated the app will open the *type* function. The *user* will then click on screen where they would like the type and begin typing using a keyboard or built in keyboard. The *user* will click the font button and select a different font activating the *changeFont* function. In the same menu the *user* will select a size for the font and activate the *changeFontSize* function. The use case ends when the *user* selects the draw button or closes the document.

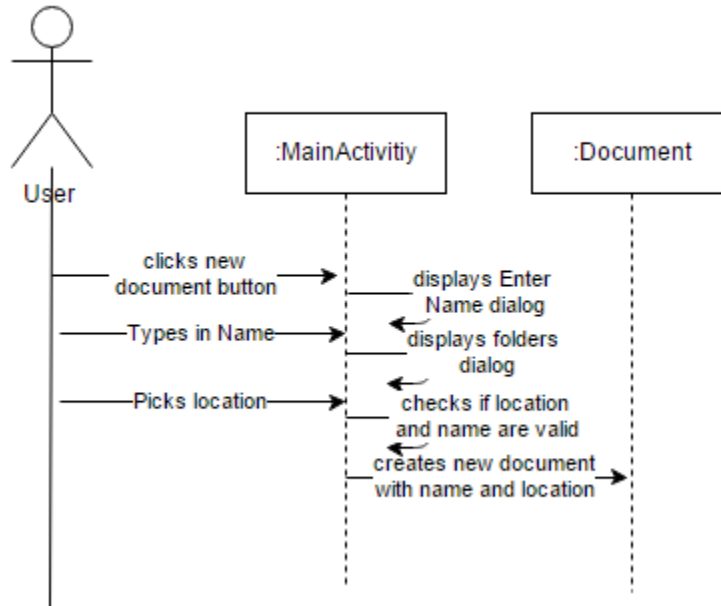
Exceptional flow of events: If the user doesn't click the screen where he/she wants to type, nothing will happen until the user does.

iii. Use Case Diagram

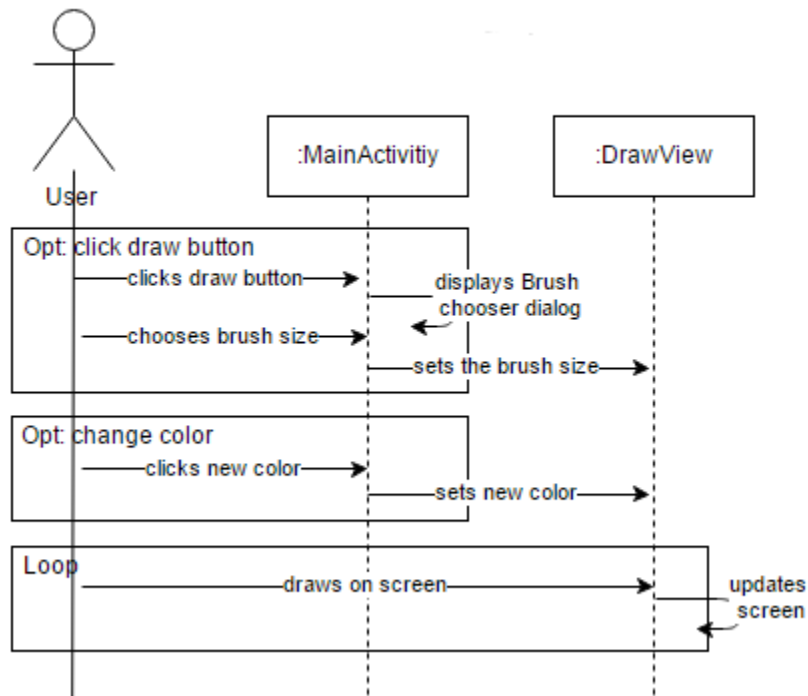


SYSTEM SEQUENCE DIAGRAMS

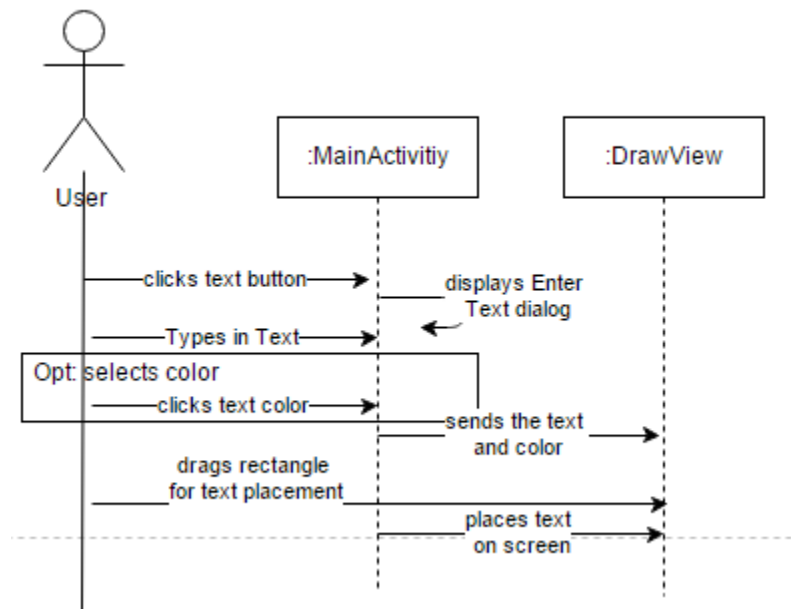
1. Use Case: Create New Document



2. Use Case: Draw



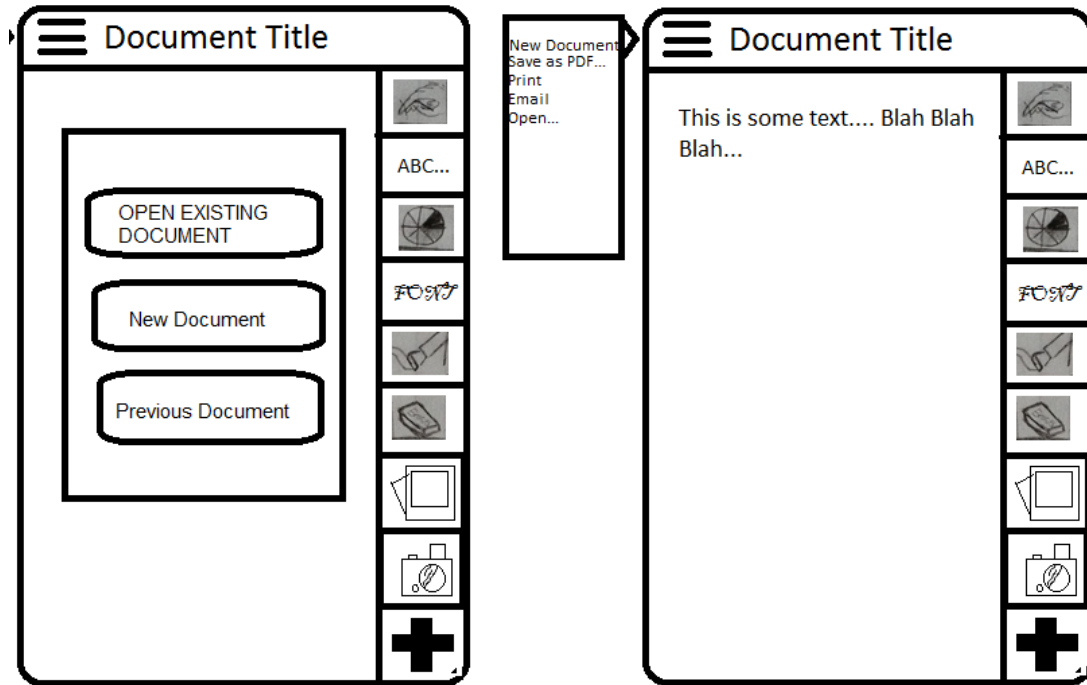
3. Use Case: Type



USER INTERFACE DESIGN

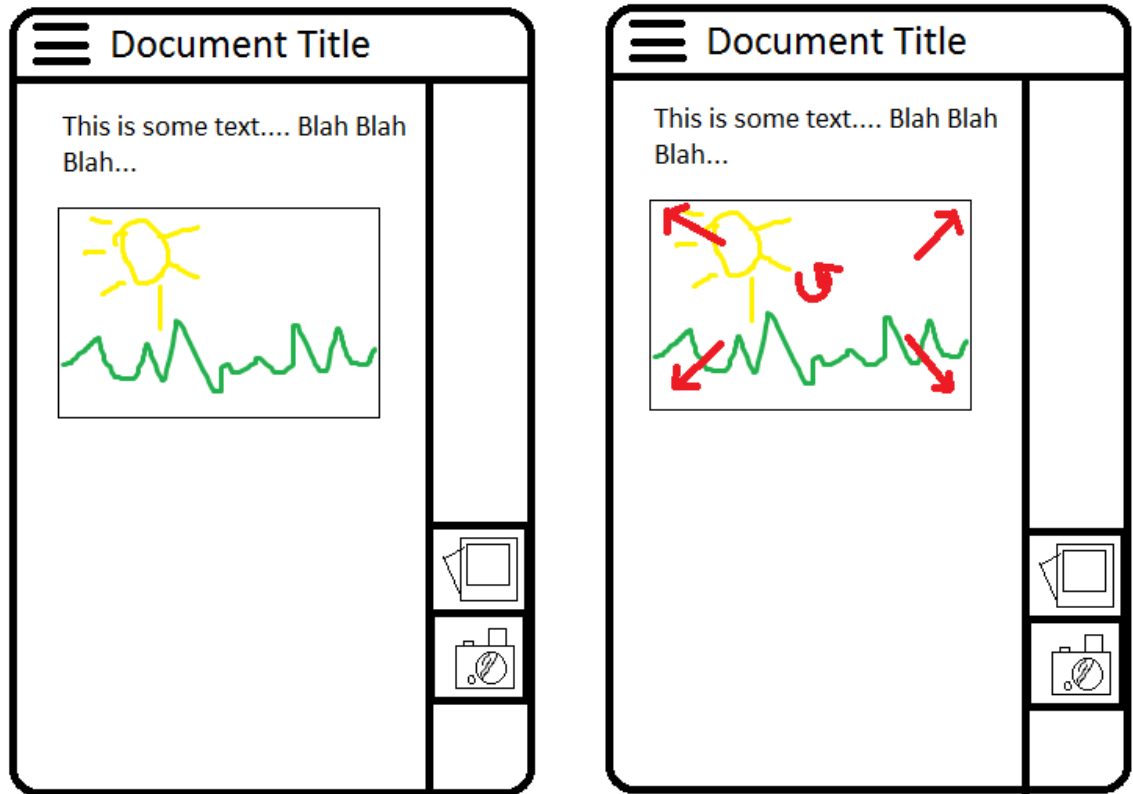
a. Preliminary Design

When the application opens, the user will be shown a small menu on the screen to either open an existing document, create a new document, or open a previous document.



Once the user selects the document they want to open they will be able to start editing a new document. There will be a top menu for the user to select if they want to start a new document, save as a PDF...(or more), print, email, or open a different document.

While the user is editing their document they have menu options to the right side. In descending order starting from the top is the free hand tool, text tool, color selector, font selector, highlighter tool, eraser, insert picture, insert picture from camera, and add another page. If the user selects the insert picture or camera button, they will select a picture and then select its location like in the left UI below and then rescale the photo with their fingers like in the right UI below.



b. User Effort Estimation:

1. Create a new document from start screen(4-25 clicks) – “Create New Document” button -> type in name(1-20 clicks) -> select location (1 – 3 clicks) -> “Create” button
2. Create a new document from the menu(5-26 clicks) – Menu button -> “Create New Document” button -> type in name(1-20 clicks) -> select location (1 – 3 clicks) -> “Create” button
3. Open a document from the start screen(3-5 clicks) – “Open Existing Document” button -> select document(1-3 clicks) -> “Open” button
4. Open a document from the menu (4-6 clicks) – Menu button -> “Open Existing Document” button -> select document(1-3 clicks) -> “Open” button
5. Freehand/draw(1-7 clicks) – Draw button -> size (optional, 1-3 clicks) -> color (optional, 1-3 clicks)
6. Insert Text(1-10 clicks) – Text button -> size (optional, 1-3 clicks) -> color (optional, 1-3 clicks) -> font size (optional, 1-3 clicks)
7. Highlighter (1 click) – Highlighter button
8. Eraser (1 click) – Eraser button
9. Insert Picture from photo library(5-18 clicks) – Picture button -> select picture from photo library(1-10) -> “Select Photo” button -> tap the screen at the correct location -> resize or rescale (1-5 clicks/swipes)
10. Insert Picture from camera(4-8 clicks) – Camera button -> take the picture -> tap the screen at the correct location -> resize or rescale (1-5 clicks/swipes)

PLAN OF WORK

			February				March				April				
			6	7	8	9	10	11	12	13	14	15	16	17	18
Activity	...	Status	1 03 - 07	10 - 14	17 - 21	24 - 28	03 - 07	10 - 14	17 - 21	24 - 28	31 - 04	07 - 11	14 - 18	21 - 25	28 - 02
Proposal		Completed	✓												
Analysis Documents		IP													
Plan of Work		Completed													
Use Cases															
System Sequence Diagrams															
UI design															
Design Documents		Not Start...													
Interaction Diagrams															
Class Diagrams															
Algorithms/Data structures															
UI design review															!

- **Proposal (2/03 - 2/10)**
- **Analysis Documents (2/17 - 3/21)**
 - a. Plan of work (2/17 - 2/26)
 - b. **Progress Report #1 (2/26)**
 - c. Use Cases (2/24 - 3/21)
 - d. System Sequence Diagrams (3/10 - 3/21)
 - e. UI design (2/24 - 3/07)
 - f. **Progress Report #2 (3/19)**
- **Design Documents (3/24 - 4/25)**
 - a. Interaction Diagrams (3/24 - 4/04)
 - b. **Progress Report #3 and Midterm Review (4/2)**
 - c. Class Diagrams (3/31 - 4/18)
 - d. Algorithms/Data structures (4/07 - 4/18)
 - e. **Progress Report #4 (4/16)**
 - f. UI design review (4/21 - 4/25)
- **Progress Report #5 and Final Review (4/30)**

REFERENCES

- Gantt Chart – www.tomsplanner.com/
- Android Pdf API - <http://developer.android.com/reference/android/print/pdf/PrintedPdfDocument.html>
- Android Camera API - <http://developer.android.com/reference/android/hardware/Camera.html>
- Text Editor - <http://developer.android.com/guide/topics/text/index.html>
- Data Storage - <http://developer.android.com/guide/topics/data/index.html>
- Definitions/Glossary of terms - Wikipedia