Ian Fair (izf0002)

Software Modeling HM3

1. The following classes should be eliminated:

* *SelectedObject, SelectedLine, SelectedBox, SelectedText*, reasoning to delete these is that they’re redundant for *Selection*. Also *Connection* because its redundant with *Link*.
* *Computer*, reasoning to delete this is because its irrelevant since were developing a model for the purpose of computer implementation.
* *GraphicsObject,* reasoning to delete this is that its too vague and needs to be more precise.
* *Position, length, width, fileName, lineSegmentCoordinate, name, origin, scaleFactor,* reasoning to delete these is that they’re attributes.
* *X-coordinate, y-coordinate, Menu, Mouse, Button, Popup, MenuItem, CornerPoint, EndPoint, and Character*. Reasoning to delete these is because of the implementation constructs.

After eliminating these classes were left with *Line, Link, Collection, Selection, Drawing, DrawingFile, Sheet, Point, Box, Buffer,* and *Text.*

1. The following associations should be eliminated:

* A box has a position. (Position is an attribute that’s been eliminated. Replace with *a box has a point.*)
* A character string has a location. (Location is also an attribute. Were also using the term ‘text’ and not ‘character’ string.)
* A line has length. (Length is an attribute that’s been deleted)
* A line is a graphical object.( Isn’t a class worth modeling)
* A point is a graphical object.(Isn’t a class worth modeling.)
* A point has an x-coordinate.(Attribute that’s been deleted.)
* A point has a y-coordinate.(Attribute that’s been deleted.)
* A character string has characters.(Not important so not necessary to include)
* A box has a character string.(Same thing as a “box has text”.)
* A box is moved. (this is an action)
* A link is deleted.(this is an action)
* A line is moved. ( this is an action)
* A link has points.(association is derived)
* A link is defined by a sequence of points. (association is derived, replace with a ‘link corresponds to one or more lines’)
* A drawing has one or more sheets.(missing from the given list)
* A drawing is stored in a drawing file.(missing from the given list)

After all of this is cleaned up were left with the following associations and generalizations: *a box has a point, a box has text, a link logically associates two boxes, a link corresponds to one or more lines, a selection or a buffer or a sheet is a collection, a collection is composed of links and boxes, a line has two points, a drawing has one or more sheets, a drawing is stored in a drawing file.*

1. The following classes require state diagrams: Buffer and Selection.

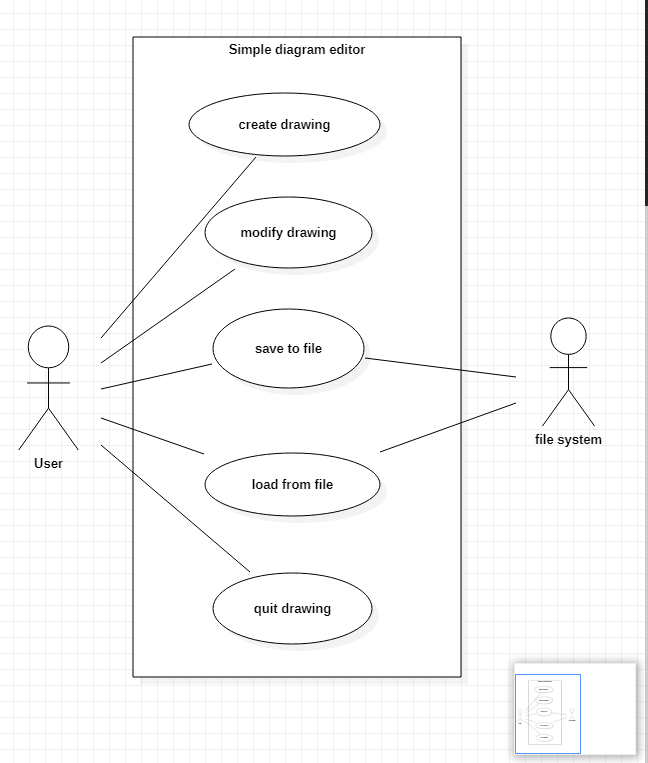
* The Buffer is used for copy, cut, and paste operations. The state of the buffer is either empty or full.
* The Selection state diagram tells us whether one or more objects are selected. Thus there are two states, Something selected and Nothing selected. The class model contains information for the Something selected state, to be specific it precisely selects which boxes and links are selected. The ‘pick’ operation needs to be more distinguished by either picking the first object selected or picking other objects.

1. This application is a simple editor that supports boxes, links, and text. Text is allowed only in boxes and the size of text and the font are fixed. The boxes automatically adjust to fit the text. The links consist of horizontal and vertical lines.

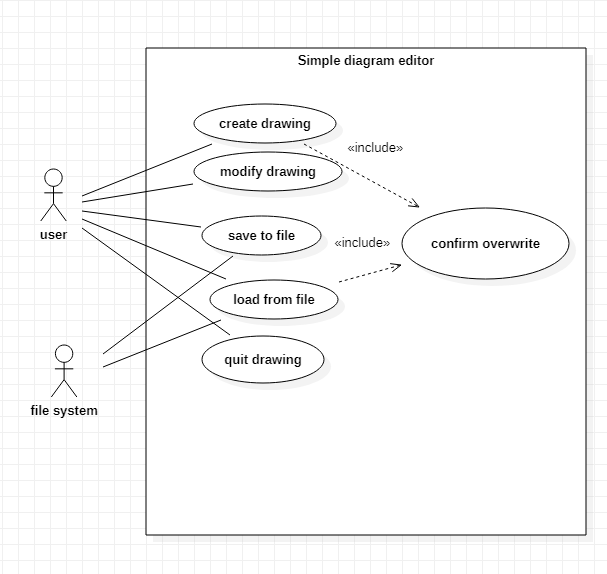
5a.) The user and file system are actors.

5b.) - **Create Drawing**. Starts a new empty drawing in memory and overwrites any prior content. Has the user confirm this if theres a prior drawing that hasn’t been saved.

* **Modifying drawing.** Changes the contents of the drawing that’s loaded into memory.
* **Save to file.** Saves the drawing in memory to a file.
* **Load from file.** Reads a file and loads a drawing into memory, overwriting any prior content. Has the user confirm this if there is a prior drawing that hasn’t been saved.



6.)



7.) - User loads an existing drawing. The editor retrieves the document and sets the cursor to last sheet that was referenced.

- User goes to the first sheet. The editor moves the cursor.

- User goes to the next sheet. The editor moves the cursor.

- User deletes the sheet. The editor asks to confirm. User confirms it.

- User goes to the last sheet. The editor moves the cursor.

- User deletes all existing sheets. The editor asks to confirm. User confirms it.

- User creates a box. The editor highlights the new box that was created. User enters the text ‘x’.

- User selects the text in box. The editor highlights text and unhighlights the box. User cuts the text. User selects the empty box. The editor highlights the empty box. User enters the text ‘y’.

- User copies x-box. The editor highlights the new copy of box. User moves the selected box.

- User copies y-box. The editor highlights the new copy of box. User then moves selected box. User then edits text ‘y’ and changes it to a ‘+’.

8.) Error Scenario 1:

- User enters command: load existing drawing file and supplies a file name.

- Command fails: file not found

Error Scenario 2:

* User selects the x-box. Editor highlights the x-box.
* User also selects the y-box. Editor highlights the y-box.
* User also selects the +-box. Editor highlights the +-box.
* User then tries to select the link box command.
* Command failed: must pick only two boxes for linking.

Error scenario 3:

* User selects the x-box. Editor highlights the x-box.
* User attempts to select the ‘enter text’ command.
* Command failed: box already has text.

9.) Sequence diagram for error scenario 1:

A screenshot of a cell phone

Description automatically generated

Sequence diagram for error scenario 2:

A close up of text on a white background

Description automatically generated

Sequence diagram for error scenario 3:

A close up of text on a white background

Description automatically generated