Project Title

# Problem statement

Create an editor that will allow the user to create and modify UML documents.

# System Personnel

## Description of Users

The users will

## Description of System Developers

The developers will be Sam Carroll, Torrance Graham, Quinn Minnich, Tommy Russoniello

# Operational Setting

## Target Platforms

The application will be designed to be potable to many platforms, including Mac, Windows, and Linux

## Required Software Environment

Java must be installed on any device that uses this program

## Useful Optional Software Environment

None.

# Functional Requirements

## Functional Description

### Overview

A graphical interface where the user can draw and remove points, as well as connect the points with arrows.

### Feature List

Add points

Remove points

Draw solid lines between points

Delete solid lines between points

Keep track of a current “selected” object which will appear in an inspector

## User Interface

### Overview

The user will be able to select one of two drawing modes (either points or lines) from a menu. If the user clicks on the canvas in “point” mode, a point will be drawn at that location. If the user is in “line” mode, the application will draw a line between the next two points that the user clicks. Nothing will happen if the two points are the same. If the user in in “select” mode, then the inspector will change to reflect what object has been most recently clicked.

### Menus

There will be one menu on the left side of the screen which will allow the user to select what type of drawing mode they want (either points, lines, or select mode)

### Windows

There will be one window, which will be divided into sections for a menu on the left, an inspector on the right, and a canvas in the middle.

### Inspectors

There will be one inspector on the right side of the screen which will show the name of the currently selected object (either a line or a point) along with a button to delete it.

## Use Cases

### Use Case 1

### Use Case 2

## Other Interfaces

# Non-Functional Requirements

## Reliability

## Performance

## Usability

## Portability

# Future Enhancements

Specification by Quinn Minnich  
Design by Tommy Russoniello  
Implementation by Torrance Graham  
Testing by Sam Carroll

Things that need to be done

Revise specifications (as needed)

Design

Implementation

Build interface (make buttons, blank canvas, ect)

Allow for drawing points

Select point

Connect with lines

Add ability to select lines?

Testing