Josh Lee

https://github.com/JoshLee0915/Arc

Arc

February 27, 2019

Overview

Arc is an open source UML/diagram designer that is meant to act as a more modern successor to Dia (http://dia-installer.de/). Dia is probably one of the premier diagraming solutions that can be obtained for free, but its UX is fairly antiquated resulting in cumbersome workflows that require the user to navigate a multitude of windows and tabs to modify UML or other diagram symbols. Along with the inadequate UX the maintenance of the code generation tools for Dia (http://dia2code.sourceforge.net/) seems to have fallen to the wayside (note: seems that their rewrite of the tool in c++ is finally in an early release state).

Due to the above issue Arc's primary focus will be on a smooth, more modern UX that can be used on multiple types of platforms to create and modify UML in a more intuitive manner. Along with the UX improvements Arc aims to integrate the code generation tools directly into the diagramming tool to allow users to easily export UML to code. Finally, Arc will aim to be easily extensible to increase its flexibility.

Goals

1. **Create a UML editor with a modern UX:** UX of the system will be the foremost factor of Arc. The more modern UX design will focus on creating more intuitive and straightforward work flows. Along with improving workflows, the new UX will be aimed at working well with both touch as well as traditional input.

2. **Built in code generation tools:** Arc will be distributed with a built in code generation tool that will allow users to convert UML into skelton code that can be filled in later. Initially we will focus on the following languages with additional languages added later down the line:
    - C
    - C++
    - C#
    - Python
    - Java
3. **Extensibility:** Arc will be extensible, allowing users to easily extend available diagram symbols, or expand what code or how code is generated.
4. **Protablity:** Arc will be able to be run across multiple platforms as well as allow custom symbols to be embedded into save files so the diagram can be opened by users who may no have the custom symbols.

Proposed Tech Stack

Arc will at least have its frontend and the UML editor implemented using python 3.0 and Kivy for the GUI library. The code generators may be implemented in pure python, or using C++ and python depending on what might work better for their implementation.

Risks or Possible Issues

The choice of using Kivy for the GUI library requires users and devs to have hardware that support OpenGL (ES) 2.0 or greater. While I believe most standard GPUs support it anymore, this does put a hardware requirement on using Arc that could increase the barrier of entry for some users/devs.