Alex Marek

11/8/20

Object Oriented Programming

History of Java GUI

When Java was first released one of it’s massive advantages was having a standardized graphical user interface library. Their first go at this was called AWT or Abstract Windows Toolkit, the goal of this GUI toolkit was for it to have a large support platform, and make porting fairly easy. This would however mean that it would come with some downsides. One major downside is that the Abstract Windows Toolkit relied upon the graphics library provided by the operating system. The reason this is a downside is because not all operating systems have equal functionality meaning that some features/designs that work for one person may look entirely different for another on a different operating system. Another downside of this approach is simply that it allowed for less component types, with less complex features. This means that although it was supported on a wide list of platforms and porting was very easy, you couldn’t do nearly as much with it as you’d want to.

Oracle a few years later decided they wanted to work on improving their GUI, so they worked on a new graphical interface system called Swing. Swing would work to address all of the bad stuff in AWT, and not provide any less functionality/compatibility. One of the most important things Swing fixed was it added a much more thorough and useful set of component types. On AWT previously there weren't tables, trees, or many other commonly used features. Swing fixed that by adding those and allowing for third party components to be added. This means that even today there is a still open source component libraries ready to be used despite Swing’s current age. Swing also brought an even greater amount of stability, at launch it obviously had no problems integrating with AWT as Swing could do everything AWT could do, but in the years following Swing has proven to also work nicely with SWT. There were however a few downsides to Swing. One important downside is a problem a lot of programs saw around that time, larger memory consumption.

Java has had a rich history of GUI tools that stretches across decades of development. There were many features introduced then that are still used today and I think it’s safe to say that Oracle has helped shape the way modern programmers use GUI.

Sources:

(n.d.). Retrieved November 09, 2020, from https://topic.alibabacloud.com/a/advantages-and-disadvantages-of-awtswingswt-in-java-gui-programming\_1\_27\_30391034.html

Tuck, M. (2001, August 13). The Real History of the GUI. Retrieved November 09, 2020, from https://www.sitepoint.com/real-history-gui/