Geany IDE

Why to use Geany?

* Is compatible with multiple envirenment like Linux, Windows, MacOs, BSD, Solaris
* Is powerful,stable and lightweight programmer’s texto editor that provides tons of useful features
* Is built-in to support more than 50 programming languages
* Is translated into over 40 lanaguages

What’s Geany?

Is a free and open source multiplataform text editor that makes use of Scintilla and GTK, including a basic feature of na IDE. Geany was developed by Nick Treleaven, Colomban Wendling, Enrico Troeger, Matthew Brush and Frank Lanitz, using C/C++ language.

* Scintilla is a free, open source library that provides a text editing component function, with an emphasis on advanced features for source code editing
* GTK(formerly GIMP ToolKit then GTK+)[4] is a free and open-source cross-platform widget toolkit for creating graphical user interfaces (GUIs).

# Evaluation

To properly evaluate Geany IDE interface, the following set of Heuristics was used:

* Visibility of system status
* User control and freedom
* Error prevention
* Flexibility and efficiency of use
* Aesthetic and minimalist design
* Help and documentation

# Methods used

1. To perform the Heuristic Evaluation (HE) we used the 10 Heuristics proposed by Nielsen
2. Rating Scale(Saverity Scale)

* 0 - Not a problem at all (Disagreeing on the existence of a problem)
* 1 - Cosmetic problema only (need not be fixed unless extra time)
* 2 - Minor usability problem (low priority)
* 3 - Major usability problem (high priority)
* 4 - Usability catastrophe (imperative to fix before releasing to production)

1. To perform the Cognitive Walkthrough (CW) we followed the Streamlined Cognitive Walkthrough proposed by Rick H. Spencer

* Q1 - Will the user know what to do at this step?
* Q2 - If the user does the right thing will they know that they did the right thing, and are making progresses towards their goal?

# Some cons(issue) about Geany IDE

* Does not have code Templates??
* Does not support Error Markup
* Does not support remote file editing protocols
* Does not include Unit Testing
* UI is not more intuitive(some icons may not be needed)
* Does not have integrated Debugger
* Does not have collaborative editing
* Does not have interactive console
* Does not include an Integrated Debugger
* Does not have a quickly help (JIT - Just In Time)
* Does not have a quickly continuous Delivery updates releases
* Does not support RTL (Right To Left – Arabic format)

# Heuristic Evaluation

1. Issue: **Does not have a quickly continuous Delivery updates releases**

Heuristic: Visibility of system status

Severity: 2,3

1. Issue: **Does not support Error Markup**

Heuristic: User control and freedom

Severity: 2

1. Issue: **Does not include Unit Testing/** **Does not have integrated Debugger**

Heuristic: Error prevention

Severity: 3,4

1. Issue: **Does not have collaborative editing**

Heuristic: Flexibility and efficiency of use

Severity: 2

1. Issue: **UI is not more intuitive (some icons may not be needed)**

Heuristic: Aesthetic and minimalist design

Severity: 2

1. Issue: **Does not have a quickly help**

Heuristic: Help and documentation

Severity: 2