

Lovelace Inc.

To: Dr. Page
From: Ashley Nanni
Subject: Design and implement the control software for a new kind of nuclear reactor
Date: November 15, 2011

The purpose of this memo is to present the finding of the programing environment research. The goal of this research is to determine what programing environment would be best to use for the new nuclear reactor.

Summary

It was important to look at the different programing environments available in order to find one that work best for the new kind of nuclear reactor. We want to make sure that nothing goes wrong with the program that may cause the nuclear reactor to react and cause detrimental damage.

Discussion

In order to determine a programing environment that would work best with the new kind of nuclear reactor we had to look what the nuclear reactor required and the environments that could fulfill these needs. We found that a good programing environment needed to have the ability to have many windows open, a good checking system, and the ability to keep the code neat and readable.

The Dracula environment uses a four window view. The four windows are important because they allow you to see the program you are writing, the console where you are testing and running the program and the two windows that you use to check the definitions and properties that you are using. With Dracula you can write your code and then run the checker that will highlight the functions in your code green once confirmed that the function runs as assumed. With the checker ability the Dracula has the four windows make it easier for the programmer to see where their code does not check out and how they can change the code to make it better. The tab function is a helpful feature that the Dracula programing environment provides. This allows the programmer to have everything they need open in the Dracula window at once. Most importantly it allows the programmer to have their definitions and properties open. The ability to have the definitions and properties programs open allow the programmer to make sure all of the code is correct when using the checker. It makes it easier to look at the definitions given to a function if a check is not running as planed without the programmer needing to search through many windows to find the code they are looking for. Finally the parentheses matching that Dracula has is an important function that makes Dracula a good programing environment for a nuclear reactor. In order to keep track of the code the parentheses matching function is used consistently in the Dracula programing environment.

Recommendation

The Dracula environment provides window options, tab functions, and parentheses matching which makes Dracula a great environment for a programmer to use when programing software for an important program such a one for a nuclear reactor that may cause detrimental damage to many people if something where to go wrong.

1. Describe three services that a good programming environment should provide

A good programming environment should have methods to help you keep track of your code and keep your code neat. Some features a programming environment may provide to help keep your program neat and readable would include auto indenting, parentheses matching, color coding and an easy way to comment in the code. A good programming environment should also have a tab function to allow you to have all parts of your project or different projects open as needed. The tab function should also allow you to arrange and close and open functions easily as needed. Finally a good programming environment will allow you to have multiple windows open at a time to help you look at more of your code at once.

2. Describe three things you like about the Dracula programming environment

The Dracula programming environment is set up so you can have all four windows open at once. It is helpful to be able to read your errors while you are looking at your code. The parentheses matching feature is another part of the environment that is very helpful. The parentheses matching helps you keep track of where you are in your code. The parentheses matching feature allows you to know what functions you have closed and which ones you are still working in. The third feature I like is the ability to have multiple programs open with the tab function. This came in handy during projects so that you could keep the properties, definitions, and test in separate files.

3. Describe three things you dislike about the Dracula programming environment

The Dracula programming environment I feel is slow. It can take a bit for the Dracula environment to load and run a program. It is also frustrating that times when it is loading slowly the screen will go completely white before it starts running your program. The tab function in Dracula can be very helpful however it is sometimes hard to use. The environment does not allow you to re-arrange the tabs once they are open, and you have to be very careful when opening and closing tabs since you can only open and close tabs from the file menu. Also since you can only close tabs from the file menu if you are closing a tab you have to make sure the tab you want to close is the one you have open on the screen. Dracula clears your work in the console when you have made changes to the program and run the program again. There are times when it would be nice to see the tests and inputs that you had run in the console before making changes.

4. Suppose you are a software engineer in an organization that has received a contract to design and implement the control software for a new kind of nuclear reactor to be constructed in a densely populated area. You have been assigned the task of choosing the programming environment to be used in the project. Draft a one-page memo that specifies the programming environment and provides a convincing rationale for that choice.