Design 922
-
Collective Report

Problems encountered

- **Description**:The layout of the elements is not always responsive to CSS command and using a variable. **Workaround**: Nestings GObjects and inserting variables afterwards. **Improvement**: Giving the Objects 2 optional constructors, value and CSS Properties.
- **Description**:Using GUIT requires knowledge in C++ making it hard for beginners to grasp the complexities behind it. **Workaround**:Building an app as simple as possible. **Improvement**:A deeper explanation of basic concepts needed for development.
- **Description**:The GuitDemo is the only source of information on the toolkit, so it requires running 2 IDE instances. **Workaround**: Switching from an instance to another. **Improvement**: Giving a web version of the tutorial at least or a way to look for the documentation of the gadgets.
- **Description**: the *TextField* default callback, which should only trigger when hitting “Return”, is also triggered when selecting the widget. **Workaround**: use the *On.keydown* callback (and manually find the keycode for the “Return” key). **Improvement**: fix this issue, and also check all the other widgets which could have a similar behavior.
- **Description**: there is no ready-made color widget. **Workaround**: copy-paste the color-palette code from the *drawing* app provided with the toolkit. **Improvement**: add a color-picking modal dialog.
- **Description**: the *GInt* class is not working properly when initialized with *min* and *max*. **Workaround**: don’t set minimum and maximum values in the UI and clamp the values afterward in the calculations. **Improvement**: fix this issue and search for other properties that could have the same problems.
- **Description**: the *GInt* class is not working properly when initialized with *min* and *max*. **Workaround**: don’t set minimum and maximum values in the UI and clamp the values afterward in the calculations. **Improvement**: fix this issue and search for other properties that could have the same problems.
- **Description**: The installation process ended incorrectly due to the absence of the required file, given that all the necessary files were downloaded. Since this step avert the work with GUIT at the initial stage, the criticality of the problem is maximum. This problem occurs for users working on the MacOS operating system. **Solution:** The first solution will be to run the GUIT application using Xcode.For this solution, it is highly recommended to update your version of OS and Xcode to prevent any other problems in advance . The second possible solution is described here.
- **Description:** During demo execution, the initial scale of the opened program slightly decreases the resolution which makes it complicated to read the labels appearing on the screen. The problem is not that big a deal but it is highly noticeable. **Solution:** The short term solution is to widen the window size to the required size customized by the user.

- **Description:** During the installation process described in the documentation, after self updating MacPort value you can face the problem: `sudo: port: command not found` . The criticality is minimum since the solution can be easy integrated. **Solution:** The quick fix for this issue will be addition of these lines into your system's PATH:
  ```
  /opt/local/bin
  /opt/local/sbin
  ```
  You can edit your PATH by executing this: `sudo vi /etc/paths`

- **Description:** Special characters cannot be displayed normally, such as "#". **Solution:** Add at least one in one of the demo application.

Toolkit qualities

- The GUI code is compact, human-readable, and reflects directly the visual output
- The uniform way of treating different *gadgets* allows to easily test and improve the UI (e.g make a label, and later realize you want to edit the text, then all you have to do is change *Label* with *TextField* and the rest will follow)
- The callback system is clear, compact and human-readable
- Using CSS allows fine-tuning the interface without recompiling the codebase
- The program is well and logically divided into files
- The code structure and language are compressed
- Requires minimal knowledge in app programming to understand the source code.
- Large sectioned documentation with detailed information related to toolkit structure
- Provided a supportive demo of the application with a collection of possible implementation
- The use of *gadgets* as building blocks for the application is an efficient way to build clear and efficient visual apps

Improvements

Short term proposals

- The Doxygen documentation should use alphabetical order for more standardized and efficient searches
- File dialogs should be improved as the UI is hard to read and displays error messages on the first click of a double click
- We should be able to specify which hotkeys should and should not require focus (e.g Ctrl+Z for *undo* should not require focus as users want the ability to use it at any time)
- A hello world app to build your app from scratch with explanations on how to start and where to go from there

Long term proposals

- The toolkit name is quite confusing and makes it hard to *talk* about the project as it is quite close to *GUI* and *git*, two ubiquitous terms in software development. I think it would be fine to have a name that is less descriptive but which stands out from other commonly used names in that domain, so that it is easy to remember and communicate.
- Additional documentation for layouting will be a good inclusion for newcomers. It will improve the understanding and faster the integration in programming for GUIT. Plus it will be a guideline for users with no experience of styling/layouting or accomplishing the frontside of the application.
- Be able to develop some code inside the code tester, so we would have an easier and more GUIT friendly IDE than Visual Studio or Xcode.
- Build a community that allows you to share more real-world use cases for your code. Since guit is a very portable programming language, the presence of this community will help even more. Uploading an existing `guitdemo` is a good starting point.

Missing elements

- How to use different cursor icons
- A detailed tutorial on how CSS works in the context of the toolkit (which is similar to CSS in web development but still has some differences). Specify how much control on users to manipulate it, what is the priority.
- An easier way to install the toolkit