Code Workstation

Design Document

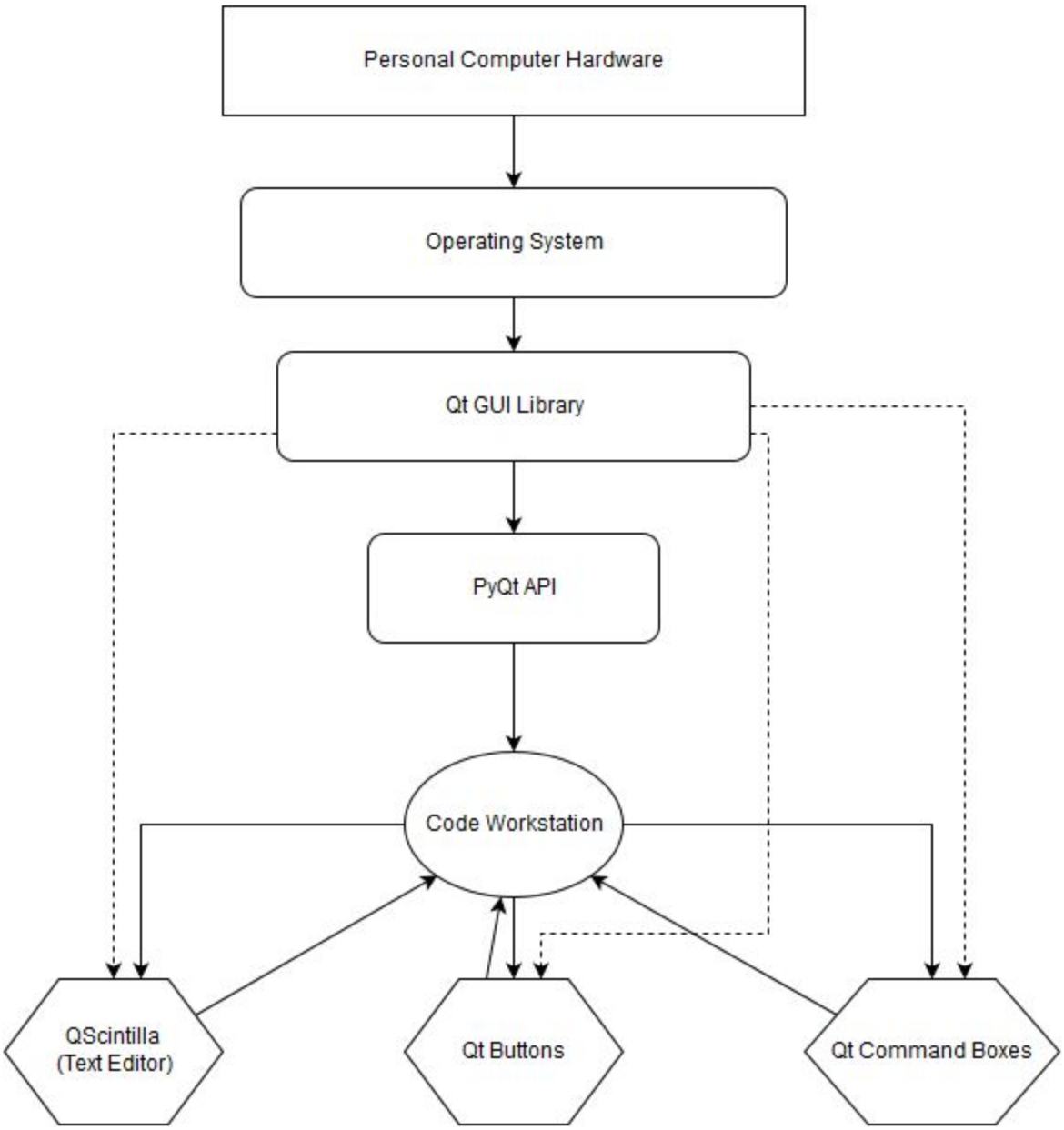
https://github.com/lrickard/CodeWorkstation/wiki

Lukas Rickard - Project Leader - l_rickard@u.pacific.edu

Jake MacMillan - Integrations Lead - j_macmillan1@u.pacific.edu

Last Revision: 11/18/2015

System Architecture



Personal Computer Hardware - Modern personal computer with i386 or x86 architecture
Operating System - Debian at first but hopefully available on all common operating systems
Qt GUI Library - Provides event handling, interface with OS, GUI elements, and windowing
PyQt API - Provides interface with Qt libraries through python
Code Workstations - Initialized window, loads settings, saves settings, and runs commands
QScintilla - Text editor build into Qt will allow users to edit text
Qt Buttons - Creates button press events with run commands and activate code
Qt Command - Created with an amalgamation of Qt interface types to provide text display and command entering with radio button settings

System Requirements

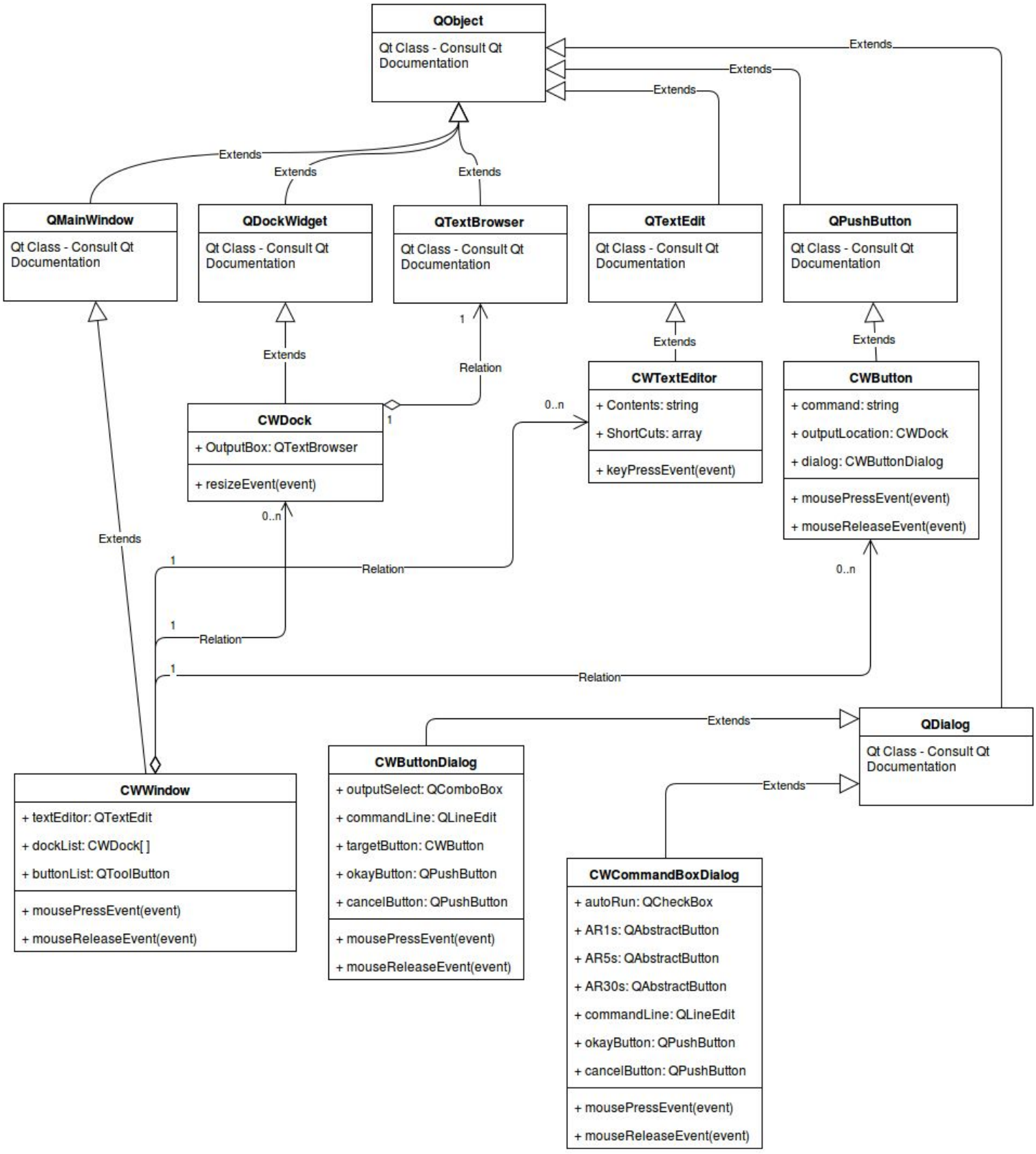
Hardware Requirements:
We suspect a single core 1 GHz CPU and 512MB of RAM will enable responsive use.
Software Requirements:
Python 3, Qt5, and PyQt5.
System Requirements:
32 or 64 bit operating system, Debian Linux and OSX are what we are developing for.

External Interfaces

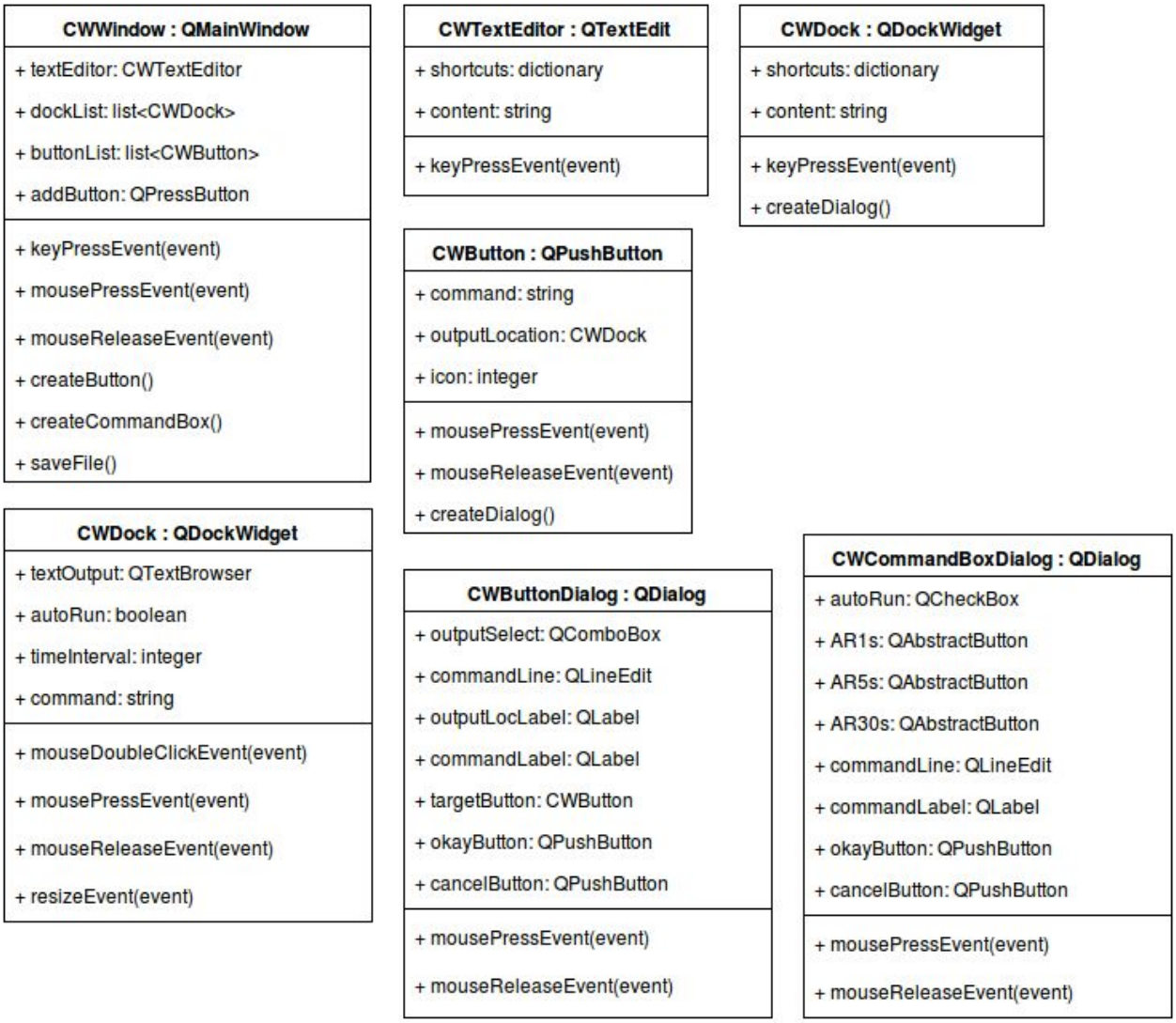
Qt - Will be used for generating windows, interfaces, and handling events.
http://doc.qt.io/
PyQt - A Python API to interface with Qt Libraries
https://riverbankcomputing.com/software/pyqt/intro

Software Design

Class Diagrams:



Class Specifications:

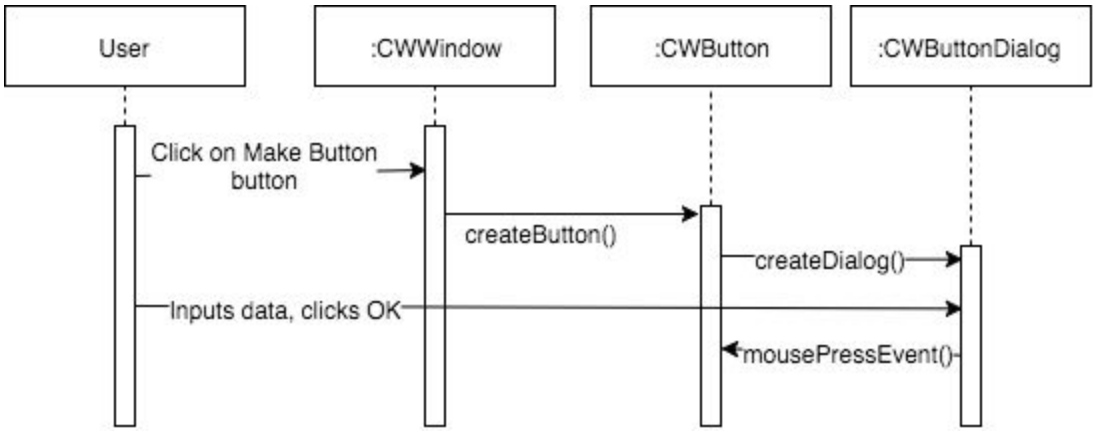


Interaction Diagrams:

Use Case 1:

Use Case 2:

Use Case 3:



Use Case 4:

Use Case 5:

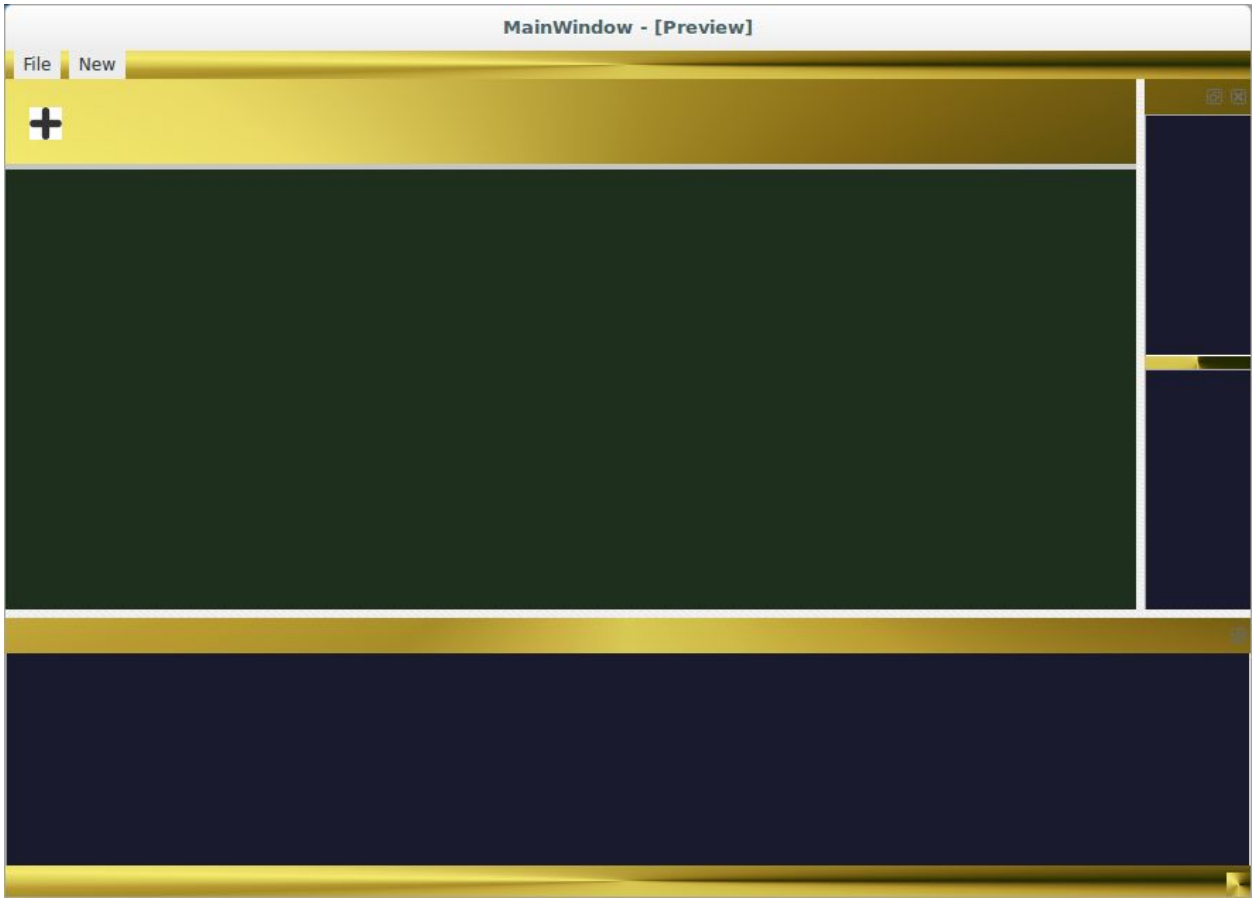
Use Case 6:

Use Case 7:

Design Considerations:

Upon consideration of performance and scalability, we decided to integrate all our features and event handling by extending Qt classes and rewriting their primary functions. This seemed to be the most efficient way to do event handling and helps to keep things organized.

User Interface Design



Buttons:

- Left Click: This should run the command stored in the button
- Right Click: This should open the menu for entering a command and an output location for the button

Text Editor:

- Right Click: This should expose a list of well know commands such as copy, paste, select all, etc.
- (Possible) Shortcuts: While most linux systems have keyboard command shortcuts, we are considering adding additional configurable shortcuts which will move the cursor and select text in convenient ways

Command Boxes:

- Left Click: This should expose the menu for entering commands and setting up automatic run
- Right Click: This should expose the option to delete or duplicate(pending) the command box and its settings

File Dropdown:

- Click Save Settings: This should save the current commands and window settings to the default location
- Click Save File: This should save the currently editing file
- Click Save File As: This should launch a save dialog/file explorer(part of Qt) which will allow the user to save the text in the main text box anywhere on the system where they have permissions

New Dropdown:

- Click Button: A new button will appear with the menu for command entering and output location exposed
- Click hover over CommandBox: Options for Left Dock, Right Dock, Top Dock, and Bottom Dock will appear

Glossary of Terms

API - Application Programming Interfaces (APIs) allow a programmer to access functionality of a library or previously created code with a specifically exposed functions and variables.
Python - A high level interpreted scripting/programming language.
OS - Operating Systems (OSs) interact with hardware and firmware to form a base for other software to interact with hardware and display things to a user. Examples: Windows 7, Debian Linux, OSX
OSX - A Macintosh operating system (OS)
UI/GUI - User Interface or Graphical User interface
UML - Unified Modeling Language is used to create class diagrams in a consistent manner
Command - Throughout this document this generally refers to a word, phrase, or set of words and phrases which can be executed in an operating system's shell/terminal.

References

Qt - http://www.qt.io/
PyQt - https://riverbankcomputing.com/software/pyqt/intro
Python - https://www.python.org/