

IMPLEMENTATION OF OBJECT-ORIENTED PROGRAMMING WITH PYQT: DEVELOPMENT OF CALCULATION APPLICATION

Isma Elan Maulani¹, Irfan Azis², Melani Nur Cahya³, Komarudin⁴, Ahmad bahar sagita⁵

¹Universitas Muhammadiyah Cirebon, Indonesia

²IPB Invada Cirebon, Indonesia

³Sekolah Tinggi Agama Islam Kuningan, Indonesia

⁴Universitas Catur Insan Cendekia, Indonesia

⁵Sekolah Tinggi Manajemen Informatika dan Komputer IKMI, Indonesia

Email: ismaelanmaulani068@gmail.com, azisirfan0507@gmail.com, mellanicaHY285@gmail.com, jrXkomarudin21@gmail.com, baharsagita1@gmail.com

ABSTRACT

KEYWORDS

Object Oriented Programming, PyQt, Computation Applications User Interface, Modularity, Application Development

This practical report discusses the implementation of object-oriented programming using PyQt in developing calculation applications. The discussion includes application development steps, object structure, and responsive user interface integration. The development method used incorporates object-oriented programming concepts to increase modularity and maintainability of the code. The resulting calculation application is able to provide an interactive and efficient user experience. It is hoped that the results of this practicum can be a reference for students of object-oriented programming and application development using PyQt.

INTRODUCTION

The background of this paper focuses on object-oriented programming (PBO) in the context of creating the PyQt Calculator application (Baihaqi & Kom, 2021). Object-Oriented Programming is a paradigm that bases code structure on the concept of objects, allowing for more modular, flexible, and easy-to-understand software development. (Sari, 2021). A calculator is a tool for calculating from simple calculations such as addition, subtraction, multiplication and division to scientific calculators which can calculate certain mathematical formulas. In its current development, calculators are often included as additional functions on computers, cellphones, even watches, the creation of this application aims to provide a solution that combines the advantages of PBO and the reliability of PyQt in user interface development.

The existence of a calculator calculation application using PyQt is important because it makes the calculation process easier. This calculator application can optimize operational efficiency, reduce the potential for calculation errors, and overall (Rahmahalim, Ramdani, & Rusydi, 2020). Through understanding the concept of PBO. Thus, this paper will explore the implementation of the PBO concept in creating a calculator application using PyQt, as well as its benefits in increasing the effectiveness of human resource management processes in the work environment. Problem Formulation, namely What is PyQt and Qt Designer? and How to make a calculator application using the PyQt method? The aim is to understand the meaning of PyQt and Qt Designer, to know how to create a calculator application using the PyQt method.

The benefit is to provide the ability to create calculation applications using PyQt and make calculations easier, especially for learning mathematics.

METHOD RESEARCH

This practicum report covers the implementation of quantitative methods in developing calculation applications using PyQt with an object-oriented programming (OOP) approach. This approach is outlined in several quantitative evaluation methods to measure performance, efficiency and application code quality. The method involves analysis of execution time, memory usage, number of lines of code, modularity, user interaction, code quality measurements, and performance comparison analysis.

First, application execution time is measured to evaluate performance efficiency between applications that implement OOP and those that do not. Memory usage is recorded and compared to assess aspects of resource usage efficiency. The number of lines of code is evaluated to measure complexity and comparison between OOP and non-OOP approaches. Next, modularity metrics are used to analyze the relationships between modules in an OOP context. Measuring user interaction involves analyzing the response time of a user interface to input. User feedback is collected and used as quantitative data to identify areas of improvement.

Code quality measurements are carried out by applying static analysis tools and comparing quality metrics between OOP and non-OOP approaches. The report also provides a quantitative performance comparison analysis, highlighting advantages and disadvantages in terms of execution time, memory usage, and other quantitative aspects. Through the application of this quantitative method, this practicum report aims to provide in-depth insight into the impact of applying OOP concepts in developing calculation applications using PyQt.

RESULT AND DISCUSSION

Basic theory

Understanding PyQt and Qt Designer

PyQt

PyQt is a cross-platform GUI Qt toolkit that ties into Python implemented as a plug-in (Hangkawidjaja, MT, & Ir Judea Janoto Jarden, 2021). PyQt is free software developed by a British company called Riverbank Computing. PyQt is free software developed by a British company called Riverbank Computing (Khairiyah & Masril, 2023). PyQt is available under the same terms as Qt versions older than 4.5; this means various forms including the GNU General Public License (GPL) and commercial licenses, but not the GNU Lesser General Public License (LGPL).[3] PyQt supports Microsoft Windows as well as various variants of UNIX, including Linux and MacOS (or Darwin OS).[4] PyQt is a GUI (Graphical User Interface) application development framework for the Python programming language (Siahaan & Sianipar, 2019b). Short for "Python bindings for Qt," PyQt provides Python bindings or libraries for the Qt development toolkit, which is a popular cross-platform GUI toolkit. Qt itself was developed by The Qt Company and is used widely in the software industry to build rich and responsive user interfaces. PyQt allows developers to create applications with attractive and interactive user interfaces using Python, which is known as an easy-to-learn and powerful programming language. It enables the use of Qt features, such as GUI widgets, layout management, event handling, and support for cross-platform application development (Arofiq, Laksana, & Saifudin, 2023).

Here are some important points about PyQt:

1. Python Programming Language: PyQt integrates Python with Qt, allowing developers to use the power and flexibility of Python for GUI application development
2. Widgets and User Interfaces: PyQt provides various widgets and GUI components that can be used to build user interfaces, such as buttons, text boxes, tables, and so on (Abidin, 2021).
3. Layout Management: PyQt provides a layout management mechanism to organize widget layouts automatically, making it easier to develop responsive interfaces.
4. 4.Event Handling: Strong support for handling events, such as responding to user input or changes in application state.
5. 5. Documentation and Community: PyQt has strong and active documentation, and is supported by a large developer community. This makes it easier to understand and resolve issues that may arise during development.
6. 6. Cross-Platform: Code written with PyQt can be run on multiple platforms such as Windows, Linux, and macOS, without significant changes.
7. 7. Compatibility with Qt: PyQt matches existing Qt versions, such as PyQt4 which supports Qt 4 and PyQt5 which supports Qt 5, thus ensuring compatibility with the Qt version used. PyQt provides an efficient and effective solution in developing GUI-based applications by using the power of Python and rich features of Qt.

QT Designer

Qt Designer is a graphic design tool included in the Qt development package. It allows developers to design user interfaces (GUI) visually without needing to write code manually (Yanto, Bima, Bahron, & Ikasari, 2023). By using Qt Designer, users can create and organize GUI elements using the drag-and-drop method, making the design process easier.

Some of the key features and concepts of Qt Designer include::

WYSIWYG (What You See Is What You Get)

Qt Designer works on the WYSIWYG principle, which allows users to see the user interface as it is during the design process. This makes the design experience more intuitive, as users can immediately see the results of the elements placed within the form.

Drag-and-Drop

User interface design can be done using the drag-and-drop method. Users can take GUI elements such as buttons, text fields, tables, and others from the Qt Designer palette and place them on forms as needed.

Properti and Layout

Qt Designer provides an interface for setting the properties of each GUI element, such as size, color, and other attributes (Syodiqi, 2013). In addition, users can manage the layout of the form to adjust the position and relationship between elements (Saad, 2020).

Event Handling

Users can define and manage event handling in GUI elements. This includes adding functions or actions that will be performed when a certain event occurs, such as when a button is pressed (Sunyoto & Kom, 2007).

Integration with Python Code

Designs created with Qt Designer can be integrated with Python code using PyQt. This Python code will connect business logic or certain functions with the designed user interface (Maulani, Putra, & Komarudin, 2023).

Visualization and Dynamic Change

Users can easily see dynamic changes in their designs when customizing element properties or layouts. This allows for experimentation and design optimization. Qt Designer helps speed up the GUI development process by providing easy-to-use visual tools, reducing dependence on writing code manually (Fernanda, Kusmaryanto, & Kurniawan, 2020). Design results from Qt Designer can be exported as XML files that can be integrated directly into development projects using PyQt.

Implementation Description

PyQt is a set of Python bindings that allows developers to create highly customizable graphical user interfaces (GUIs) with Python. PyQt widgets can be created in various ways depending on the level of customization required. Qt Designer is a Qt tool used to design and build GUIs using an editor what you see is what you get. Building widgets using this easy-to-use tool allows developers to quickly iterate on designs while seeing immediate feedback. This post will cover step-by-step instructions on how to create a PyQt widget using Qt Designer + Python (Maulani, Herdianto, Syawaludin, & Laksana, 2023).

Implementation steps :

Step 1

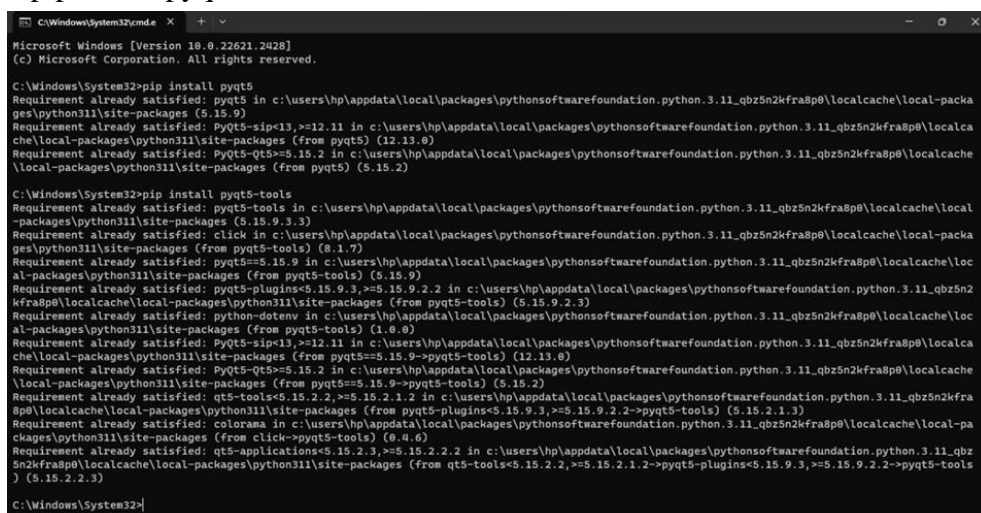
To install PyQt and Qt Designer on Windows, you can follow these steps:

Python installation

Make sure you have Python installed on your system, you can download Python from the official site python.org.

PyQt installation

Open a terminal or command prompt and run the following command to install PyQt5 using pip: ``pip install pyqt5``



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>pip install pyqt5
Requirement already satisfied: pyqt5 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packa
ges\python311\site-packages (5.15.9)
Requirement already satisfied: PyQt5-sip<13,=>12.11 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localca
che\local-packages\python311\site-packages (from pyqt5) (12.13.0)
Requirement already satisfied: PyQt5-Qt5=>5.15.2 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache
\local-packages\python311\site-packages (from pyqt5) (5.15.2)

C:\Windows\System32>pip install pyqt5-tools
Requirement already satisfied: pyqt5-tools in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local
-packages\python311\site-packages (5.15.9.3.3)
Requirement already satisfied: click in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packa
ges\python311\site-packages (from pyqt5-tools) (8.1.7)
Requirement already satisfied: pyqt5==5.15.9 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\loc
al-packages\python311\site-packages (from pyqt5-tools) (5.15.9)
Requirement already satisfied: pyqt5-plugins<5.15.9.3,=>5.15.9.2.2 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2
kfra8p0\localcache\local-packages\python311\site-packages (from pyqt5-tools) (5.15.9.2.3)
Requirement already satisfied: python-dotenv in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\loc
al-packages\python311\site-packages (from pyqt5-tools) (1.0.0)
Requirement already satisfied: PyQt5-sip<13,=>12.11 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localca
che\local-packages\python311\site-packages (from pyqt5==5.15.9->pyqt5-tools) (12.13.0)
Requirement already satisfied: PyQt5-Qt5=>5.15.2 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache
\local-packages\python311\site-packages (from pyqt5==5.15.9->pyqt5-tools) (5.15.2)
Requirement already satisfied: qt5-tools<5.15.2.2,=>5.15.2.1.2 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra
8p0\localcache\local-packages\python311\site-packages (from pyqt5-plugins<5.15.9.3,=>5.15.9.2.2->pyqt5-tools) (5.15.2.1.3)
Requirement already satisfied: colorama in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-pa
ckages\python311\site-packages (from click->pyqt5-tools) (0.4.6)
Requirement already satisfied: qt5-applications<5.15.2.3,=>5.15.2.2.2 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz
5n2kfra8p0\localcache\local-packages\python311\site-packages (from qt5-tools<5.15.2.2,=>5.15.2.1.2->pyqt5-plugins<5.15.9.3,=>5.15.9.2.2->pyqt5-tools
) (5.15.2.2.3)

C:\Windows\System32>
```

Installing Qt Designer

Qt Designer is a visual design tool for easily creating user interfaces. Qt Designer is usually included in the Qt installation package. So, you need to download and install Qt from the official Qt website.

1. Select the latest Qt version that suits your needs and download the installer.
2. Run the installer and follow the steps. Make sure to select the Qt Designer component during the installation process

PyQtTools Configuration (Optional)

PyQt provides several additional tools, including PyQtTools which includes Qt Designer (Fernanda et al., 2020). To enable this tool, you need to install PyQtTools and add it to your PyQt installation. `pip install PyQt5-tools`

After installation, you can check whether Qt Designer is installed by running the following command: `pyqt5-tools designer`

This will open Qt Designer and confirm that the installation was successful. Here is a step by step guide to create a simple calculator using Qt Designer and PyQt on Windows:

Step 2:

Calculator Interface Design using Qt Designer

1. Open Qt Designer and create your calculator interface design. Add buttons and text areas according to your needs. For example, you might want to add number keys 0-9, operators (+, -, *, /), and a text area to display results (Yansen & Prijono, 2014).
2. Each interface element (button, text area) needs to be given a unique name via the "Property Editor". This will make it easier to access those elements in Python code.
3. Save your design with the `.ui` file extension. For example, name your file `calculator.ui`.

Step 3:

Convert UI Design to Python Code

Once you have saved your UI design, convert the `.ui` file to a Python file using the following command in the terminal or command prompt: `pyuic5 -x calculator.ui -o calculator_ui.py`

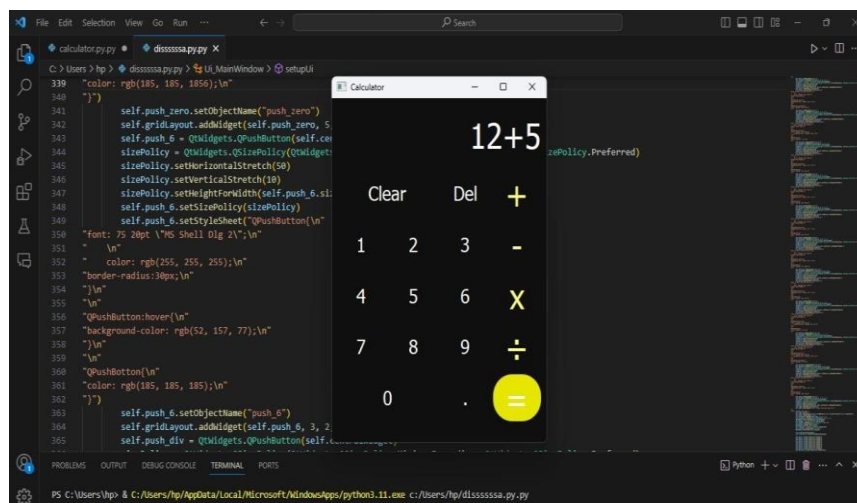
Step 4:

Create Python Code for Calculator Logic, here is the code:


```
def tambah(x, y):  
    return x + y  
  
def kurang(x, y):  
    return x - y  
  
def kali(x, y):  
    return x * y  
  
def bagi(x, y):  
    if y != 0:  
        return x / y  
    else:  
        return "Error: Tidak dapat melakukan pembagian dengan nol."  
  
# Input dari pengguna  
angka1 = float(input("Masukkan angka pertama: "))  
operator = input("Masukkan operator (+, -, *, /): ")  
angka2 = float(input("Masukkan angka kedua: "))  
  
# Pemrosesan operasi  
if operator == "+":  
    hasil = tambah(angka1, angka2)  
elif operator == "-":  
    hasil = kurang(angka1, angka2)  
elif operator == "*":  
    hasil = kali(angka1, angka2)  
elif operator == "/":  
    hasil = bagi(angka1, angka2)  
else:  
    hasil = "Error: Operator tidak valid."  
  
# Output hasil  
print("Hasil: ", hasil)
```

Step 5:

Run the Calculator Application, Run your Python script. This will open the calculator application window with the interface you have designed (Siahaan & Sianipar, 2019a).



CONCLUSION

In this practicum report, we successfully implemented a calculator calculation tool using the PyQt method. The calculation results are displayed clearly and accurately. This implementation makes practical contributions in a variety of contexts, ranging from being used to develop concepts, Adding It Up: Helping Children Learn Mathematics (NRC, 2001), Ease of Use Interface, In applications that have been built, it is clear that user interfaces designed using PyQt provide ease of use. Users can easily understand and use the calculation functions without difficulty, and the Calculator Improves Problem Solving. The application also provides insight into the use of PyQt and Qt Designer in the development of interactive and engaging web interfaces.

In addition to the practical benefits, this research provides an understanding of how PyQt and Qt Designer can be used in software development. PyQt is a set of Python libraries that provides bindings for Qt, a cross-platform software development framework. By using PyQt, Python developers can create applications with graphical interfaces using the functions and features provided by Qt. Qt Designer is a graphical development tool provided by Qt for designing user interfaces (GUI) of applications. With Qt Designer, developers can create user interfaces visually without writing code manually. Interfaces designed with Qt Designer can be saved in XML format and then integrated into Qt applications using those files. This implementation also shows the importance of validating user input in ensuring the data entered is appropriate. This research has the potential to provide a foundation for further development in the use of PyQt in web application development projects.

REFERENCES

- Abidin, Zainal. (2021). *TA: Klasifikasi Jenis Kendaraan pada Gerbang Tol Menggunakan Metode YOLO (You Only Look Once)*. Universitas Dinamika.
- Arofiq, Nur Muhammad, Laksana, Anggita, & Saifudin, Aries. (2023). Pengujian Sistem Schedule Planning Produksi Dengan Metode Black Box Testing pada PT. Smartfren Telecom TBK Untuk Pemula. *TEKNOBIS: Jurnal Teknologi, Bisnis Dan Pendidikan*, 1(1), 71-79.
- Baihaqi, Wiga Maulana, & Kom, S. (2021). *Pemrograman Berbasis Event dengan PyQt5* (Vol. 1). Zahira Media Publisher.
- Fernanda, Aufa Rifky, Kusmaryanto, Sigit, & Kurniawan, Dwi Fadila. (2020). Rancang Bangun OpenBTS Menggunakan USRP N210 Berbasis Asterisk Untuk Layanan SMS dan LMS. *Jurnal Mahasiswa TEUB*, 8(3).
- Hangkawidjaja, Ir Aan Darmawan, MT, Dr Yosafat Aji Pranata, & Ir Judea Janoto Jarden, M. T. (2021). *Perancangan Dan Realisasi Detektor Retak Permukaan Dinding Bangunan*. Penerbit YLGI.
- Khairiyah, Hani, & Masril, Muhammad Abrar. (2023). Implementasi Teknologi Computer Vision Pada Alat Penghitung Botol Air Minum Kemasan Di Industri Menggunakan Metode HSL. *JURNAL QUANCOM: QUANTUM COMPUTER JURNAL*, 1(2), 11-15.
- Maulani, Isma Elan, Herdianto, Tedi, Syawaludin, Dwi Febri, & Laksana, Medika Oga. (2023). Penerapan Teknologi Blockchain Pada Sistem Keamanan Informasi. *Jurnal Sosial Dan Teknologi*, 3(2), 99-102.
- Maulani, Isma Elan, Putra, Dwi Rayhan Sunandar, & Komarudin, Komarudin. (2023).

- Sistem Deteksi Intrusi Cerdas: Studi Perbandingan Algoritma Pembelajaran Mesin Untuk Keamanan Siber. *Jurnal Sosial Teknologi*, 3(11), 918–923.
- Rahmahalim, Muhammad, Ramdani, Fatwa, & Rusydi, Alfi Nur. (2020). Design and development of land surface temperature calculation plugin of QGIS. *Proceedings of the 5th International Conference on Sustainable Information Engineering and Technology*, 206–212.
- Saad, Muhammad Ibnu. (2020). *Otodidak Web Programming: Membuat Website Edutainment*. Elex Media Komputindo.
- Sari, Indah Purnama. (2021). *Buku Ajar Rekayasa Perangkat Lunak (Vol. 1)*. umsu press.
- Siahaan, Vivian, & Sianipar, Rismon Hasiholan. (2019a). *Langkah Demi Langkah Pemrograman Database Sqlite Dan Mysql Dengan Gui Python*. SPARTA PUBLISIHING.
- Siahaan, Vivian, & Sianipar, Rismon Hasiholan. (2019b). *Pemrograman Gui Python Dan Database: Widget Pyqt, Sqlite, Mysql, Dan Grafika*. Sparta Publishing.
- Sunyoto, Andi, & Kom, M. (2007). *Ajax Membangun Web dengan Teknologi Asynchroneuse JavaScript&XML*. Penerbit Andi.
- Syodiqi, Agus. (2013). *Aplikasi rekomendasi perguruan tinggi berbasis semantic web dengan metode multi criteria decision making (MCDM)*. Universitas Islam Negeri Maulana Malik Ibrahim.
- Yansen, Yansen, & Priyono, Agus. (2014). Pengembangan Aplikasi Mobile dengan Qt Sdk: Studi Kasus Monitoring Ruang Menggunakan Kamera. *ComTech: Computer, Mathematics and Engineering Applications*, 5(1), 473–484.
- Yanto, Mugii, Bima, Panji Eky, Bahron, Muhammad, & Ikasari, Ines Heidiani. (2023). Pemrograman Menggunakan Java NetBeans. *Buletin Ilmiah Ilmu Komputer Dan Multimedia (BIIKMA)*, 1(3), 367–377.

Copyright holders:

Isma Elan Maulani, Irfan Azis, Melani Nur Cahya, Komarudin, Ahmad bahar sagita (2024)

First publication right:

Devotion - Journal of Research and Community Service



This article is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International](https://creativecommons.org/licenses/by-sa/4.0/)