Design and Refactoring of a 3D Robotic Arm Control System

Eduards Abisevs

October 23, 2024

Abstract

This report documents the design and refactoring process of a 3D robotic arm control system implemented using PySide6 for the GUI and ESP32 for hardware control. It includes an analysis of the original system design, the challenges encountered, and the improvements made during the refactoring process. The final system demonstrates enhanced modularity, maintainability, and performance.

Contents

1 Introduction

1.1 Project Background

Provide context for the project, explaining the motivation behind controlling a 3D robotic arm via GUI and ESP32.

1.2 Objectives

State the main goals, both in the original implementation and in the refactor (e.g., modularity, reduced dependencies, better performance).

1.3 Overview

Briefly outline the structure of the report, touching on both the original system and the planned refactoring.

2 Current System Overview

2.1 System Architecture

Describe the architecture, including hardware and software components. Use diagrams where necessary.

2.2 How it Works

Explain the flow of data and interaction between the components (e.g., GUI inputs, WebSocket server, servo motor control).

2.3 Advantages

List the strengths of the current implementation, such as modular project structure and ease of deployment.

2.4 Challenges

Highlight problem areas, such as circular dependencies in the GUI or limitations in communication between components.

3 Refactoring Motivation

3.1 Problem Areas

Go into more detail on the problems identified in the current system, such as maintainability issues or performance bottlenecks.

3.2 Refactoring Goals

List the goals of the refactoring process. For example, improved modularity, reduced circular dependencies, and optimized communication.

4 Refactoring Design & New Architecture

4.1 New System Overview

Provide an updated system architecture that reflects the refactored system. Include diagrams showing the new modular design.

4.2 GUI Refactor

Discuss the changes made to the PySide6 GUI, such as resolving circular dependencies or improving the event flow.

4.3 Backend Refactor

Describe any improvements to the WebSocket server or backend communication protocols.

4.4 New Features

Detail any new features or functionality introduced during the refactoring process, such as the ability to update Wi-Fi credentials dynamically.

4.5 Code Examples

Provide code snippets that show key improvements. Use a side-by-side comparison to highlight how the refactor improves the code.

5 Implementation & Testing

5.1 Step-by-Step Implementation

Explain how the refactor was implemented, including any challenges or roadblocks encountered.

5.2 Testing the New Design

Discuss the testing process, including unit testing, integration testing, and performance testing.

5.3 Challenges During Refactoring

Highlight any new issues that arose during the refactoring process and how they were resolved.

6 Results

6.1 Before and After Comparison

Provide a comparison of the old and new systems, focusing on performance metrics, code maintainability, and system scalability.

6.2 Performance Metrics

Include any specific performance improvements, such as reduced latency or faster communication times.

6.3 Modularity and Scalability

Discuss how the new design improves modularity, making the system easier to extend in the future.

7 Discussion

7.1 Analysis

Reflect on the overall effectiveness of the refactoring. Was it successful in addressing the original challenges?

7.2 Lessons Learned

Discuss what you learned throughout the refactoring process, both technically and in terms of project management.

8 Future Work

8.1 Further Improvements

Mention any additional changes or improvements that could be made in future iterations of the project.

8.2 Next Steps

Outline potential future developments, such as adding more advanced arm control or expanding the GUI.

9 Conclusion

Summarize the key takeaways from the project, focusing on the success of the refactoring process and the final outcome.

10 References

List any sources, documentation, or references used throughout the project.

A Appendices

A.1 Code Listings

Include full code listings if necessary, or link to a GitHub repository.

A.2 Additional Diagrams

Any additional diagrams or technical schematics can be included here.