Projektarbeit

Developing Speed Measurement Algorithm

with OpenCV and Raspberry Pi

Muhammad Haziq Bin Mohd Sabtu

Technische Hochschule
Brandenburg
University of
Applied Sciences
Fachbereich
Wirtschaft

Technische Hochschule Brandenburg
Fachbereich Technik
Studiengang Maschinenbau
Betreuer: Prof. Dr.-Ing. Christian Oertel

Brandenburg, den 25. Oktober 2027

Ich, Vorname Name, Student im Studiengang ??? der Technischen Hochschule Brandenburg, versichere an Eides statt, dass die vorliegende Abschlussarbeit selbststndig verfasst und nicht mit anderen als den angegebenen Hilfsmitteln erstellt wurde.

Sie wurde in dieser oder hnlicher Form noch keiner Prfungskommission vorgelegt.

Vorname Name

Abstract

Abstract in english and german

Contents

1 Introduction

Project Introduction

Part I

Prototype Development

1 Methodology

1.1 Design Methodology

explaination of the design methodology from VDI 2221 [Con]

2 Task Clearification and Specification

2.1 Requirement of the Prototype

List of requirements for the prototype

Must have:

- Ergonomic - Comfortable to hold, Easy to use
- Portable - Lightweight, Small
- Size (MAX)
  - Length: 25 cm
  - Width: 25 cm
  - Height: 25 cm
- Weight (MAX): 3 kg
- Compliance and Regulation - Comply with the regulations of the country of use
- Cost - Affordable, < 300 Euro (including Pi, Camera and Screen)
- Appointments - Completed within 3 months
- Design - Components are packed in a chasis
- Camera - Camera must be presented in the prototype
- Power - Battery powered, Rechargeable battery, Duration min. 1 hour

- Control - Control via touch screen

Optional Requirements:

- Durability - Water resistance, Dust resistance
- Modular - Easy to assemble and disassemble, Swappable parts
- Features - Mountable on a tripod
- Fertigung - 3D printed parts

2.2 Requirement List

List of requirements will be generated from the must have and optional requirements (Section 2.1)

3 Concept Generation

3.1 Function Structure

3.2 Idea Generation

Idea Generation via Market Research, Competitive Analysis, Brainstorming

Method is suitable, due to the face that handheld devices are common in the market

3.3 Combination of Ideas with Morphological Chart

List of ideas from brainstorming will be combined with the function structure to generate a morphological chart

Atleast 3 Design Concepts will be generated from the morphological chart

4 Design

4.1 Concept Selection and Evaluation

explaination of the design and discussion of advantages and disadvantages

4.1.1 Design 1

4.1.2 Design 2

4.1.3 Design 3

4.2 Final Design

4.2.1 CAD Drawing

Final CAD Design will be presented here. Including with the features

4.2.2 Part List

List of parts used in the final design

5 Conclusion

Conclusion of the project

Part II

GUI Development

1 Methodology

1.1 MVC Pattern

The Model-View-Controller (MVC) pattern is a software architectural pattern that separates an application into three interconnected components: the model, the view, and the controller. The model represents the data and logic of the application, the view displays the data to the user, and the controller handles user input and updates the model and view accordingly. This pattern promotes separation of concerns, modularity, and code reusability in software development. [Ver19]

1.2 Design Patterns - Thread Pool

A thread pool is a software design pattern that manages a pool of worker threads to efficiently execute tasks. Instead of creating a new thread for each task, a thread pool reuses existing threads, minimizing the overhead of thread creation. It improves performance and resource utilization by limiting the number of concurrent threads and providing a queue to handle incoming tasks.[Bro22]

2 Designing

2.1 Requirement

Must have:

- Usability - Easy to use
- performance - Fast processing by utilising multiple threads
- Responsiveness - Responsive GUI, avoid methods that block the GUI thread
- Error Handling - Handle errors gracefully, avoid crashing the application
- Scalability - For future development
- Documentation - user guides, Tooltips, comments
- Design - Clean and simple design

2.2 Wireframe

Program flow and GUI design will be presented in a wireframe.

* Flow of the program is not finalized, will be updated in the future

2.3 GUI Design

Design of the GUI will be presented here. Panels, Buttons, Textfields, etc.

3 GUI-Implementation

3.1 Model

Implementation of the model

3.2 View

Implementation of the view

4 Testing

4.1 Unit Testing

Unit testing is a software testing approach that involves testing individual components or units of code in isolation to ensure they function correctly. It verifies the behavior of small, independent units of code, such as functions or methods, to validate their expected functionality and catch any defects early in the development process. [Ham23]

5 Conclusion

Conclusion of the project

Part III

Indexes and Appendix

List of Figures

List of Tables

Bibliography

[Bro22] BROWNLEE, JASON: *Threadpoolexecutor in Python: The complete guide*, Dec 2022.

[Con] CONRAD, KLAUS-JÖRG: *Konstruktionsphase Konzipieren*, pages 169–249.

[Ham23] HAMILTON, THOMAS: *Unit testing tutorial - what is, Types; Test example*, May 2023.

[Ver19] VERSTEHEN, DATENBANKEN: *Model View Controller Pattern Definition; Erklärung: Datenbank, DWH; Bi Lexikon*, Oct 2019.

A Appendix

- Docs
- Repository

A.1 CAD Drawings

A.2 Bill of Materials

A.3 Code snippets

A.4 Additional information, pictures, handout, etc.