Date: 08 May 2022 Version 1.1.5

|  |  |
| --- | --- |
|  | Project document |
|  | **Laura Vallinaho, Marko Tiitto, Jorrit van Triest, Thomas Schenk,**  **Sara van Eersel, Simon Schreurs & Chiara Babiak**  Supervisors: Suzana Andova & Tommi Kokko |
|  |  |
|  | Supervisors: Suzana Andova & Tommi Kokko |

# Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Who | Changes |
| 1.0 | 10-04-2022 | Sara van Eersel | Made a start at the document and wrote the Project organization and the risk analysis.  Started on writing on the Introduction, Project idea and goal. And added parts from the documents of both groups. |
| 1.1 | 10-04-2022 | Everyone | Reviewed the document in a meeting and adding some bits |
| 1.1.5 | 11-04-2022 | Jorrit | Added a context diagram |
| 1.5 | 13-04-2022 | Sara | Made changes according to feedback we received |
| 2.0 | 04-05-2022 | Sara | Added the protocol |

# Introduction

In this document you can read about our project for the Blended international Project. This is a collaboration with the UAS Technikum Wien, Lapin AMK and Fontys University of Applied Science.

# Contents

[Version Control 1](#_gjdgxs)

[1 Introduction 1](#_30j0zll)

[Contents 2](#_1fob9te)

[1 Project 3](#_3znysh7)

[1.1 The idea 3](#_2et92p0)

[1.2 Goal of the project 3](#_tyjcwt)

[1.2.1 Must have 3](#_3dy6vkm)

[1.2.2 Should have 3](#_1t3h5sf)

[1.2.3 Could have 3](#_4d34og8)

[1.2.4 Would have 4](#_35nkun2)

[1.3 System description 4](#_2s8eyo1)

[1.4 Finished products / Deliverables 5](#_17dp8vu)

[2 Approach and Planning 6](#_3rdcrjn)

[2.1 Approach 6](#_26in1rg)

[2.1.1 Test approach 6](#_lnxbz9)

[2.2 Research methods 6](#_1ksv4uv)

[2.3 Time plan 6](#_44sinio)

[3 Project Organization 7](#_2jxsxqh)

[3.1 Team members 7](#_z337ya)

[3.2 Communication 7](#_3j2qqm3)

[4 Risk analysis 8](#_1y810tw)

# Project

## The idea

We are going to make a robot that is going to recognise whether some fruit is ripe or unripe and sort it based on that information.

Our project is designed to be a part of an automated harvesting mechanism. When a botanical product is harvested, there are often products that need to be separated by a specific attribute like colour, size, shape, or texture. The camera can detect the attributes of a product and decide what destination it is guided towards by sending a command to move the robot arm. There is a node-red dashboard that allows the user to observe the robot.

We are doing this project because we’ve discovered that it’s very difficult to move the robot based on camera input. Having the product in motion already means that locomotion is not an issue either. The robot is not very flexible or quick, so having it direct things that are going to roll down prevents overcomplication.

## Goal of the project

Our goal is to be able to examine a product with the camera, and to have it be placed in the correct group based on that examination.

### Must have

* The ablity to distinguish differently coloured objects
  + We want to make sure the camera can recognise green and red to see if the fruit is ripe or not.
* The ability to direct an object in different directions
  + We want to move the robot in two different directions depending on the color that is detected by the camera.
* Wireless communication
  + We want to use wireless communication to make the robot and camera communicate about the color detected and make a dashboard with this data.
* Dashboard
  + A dashboard so the user can see different kinds of information. For example, the error rate, which position the robot is in and how many fruits are being processed.

### Should have

* Be able to quickly change the directory of the track
  + The robot should be able to change track within 2 seconds
* Be able to seamlessly interpret signals coming from the camera
  + The robot should be able to interpret signal from the camera within 2 seconds from receiving
* Be able to store multiple pre-set conveyor positions
  + ?
* Automated efficiency reporting
  + ?

### Could have

* Be able to be controlled manually both physical and digital
  + The robot should be able to be controllable through the dashboard and through buttons connect to the machine
* Multiple input camera’s
  + So, the user can set up multiple tracks in a factory to make the sorting process faster.
* Order accumulation
  + ?

## System description

Diagram

Description automatically generated

*Image 1: rough sketch of system concept*

Diagram

Description automatically generated

*Image 2: Context Diagram*

## Protocol

We are going to use MQTT as a protocol to communicate between the different parts.

For the program to communicate well, we need to set up what messages we are gonna send. The camera only detects color. In our case only red (ripe) and green (unripe).

To make it the easy way, we define the colors to integers. This makes the message smaller so this will be faster to send.

**Green**: 0

**Red**: 1

This will be the only thing that the Jetson with the camera will output towards the robot arm. When the robot arm receives the message, it will put it in an queue, to prevent total failure because of timing issues. The Jetson is not receiving information and only transmitting information.

Main MQTT Server name: FruitSystem

So, every subtopic must be called with Fruitsystem/desired topic

Robot Messages: Used by the dashboard

-> Status Info

Camera Messages: Used by the robot

-> Detection Messages (0: ripe, 1: unripe)

Error Messages: Used by both the dashboard and the robot (depends a bit on what the actual error message is)

-> Possible errors (I am thinking of doing like a distance sensor to detect when something is going in)

### Messages

|  |  |  |
| --- | --- | --- |
| Topic | Message | Meaning |
| Fruitsystem/color | 0 | The fruit is green so unripe |
| Fruitsystem/color | 1 | The fruit is red so ripe |
| Fruitsystem/robot | OnPosition | The robot is on position |
|  |  |  |

## Finished products / Deliverables

* A robot that can sort products based on the output of the camera.
* Project document
* Presentations
* Document of the robot group
* Research document of the camera group
* Testing report
* Notes of the meetings

# Approach and Planning

## Approach

We are working in sprints of three weeks with a demo at the end of each sprint for our supervisors. We will have meetings every Sunday at 19:00 CEST with our group, and we will have a meeting with the supervisors every Monday at 12:30 CEST.

### Test approach

We are making two different test setups that contain the same hardware. One of these setups will be built in Finland and one of them will be built in the Netherlands. The difference will be a degree of tuning to increase efficiency

## Research methods

## Time plan

|  |  |
| --- | --- |
| Week 8 | New/improved document, moving robot, robot test input, node-red overview, camera colour output |
| Week 9 | MQTT protocol, Cardboard track prototype, mechanical design, ping pong balls |
| Week 10 | Robot sorting, Node-Red interval calculation, MQTT Protocol integration |
| Week 11 | Sorting mechanism prototype, Node-Red visualisations, test system start |
| Week 12 | Build of test systems in 2 locations |
| Week 13 | Full System Testing & Integration, document review |
| Week 14 | Demo in Vienna. |

# Project Organization

## Team members

We have a group of 8 people. Our group leader is Sara van Eersel, and the note taker is Chiara Babiak. We spilt the group into two groups one being the camera group and the other one being the robot group. The camera group exist of Sara van Eersel, Chiara Babiak, Simon Schreurs and Miko Koivula. The robot group exist of Laura Vallinaho, Marko Tiitto, Jorrit van Triest and Thomas Schenk. The robot group is responsible for making the robot move and the camera group is responsible for image recognition.

## Communication

The communication within the group goes through Discord. The communication with the supervisors goes through Teams. All the code and documentation are uploaded into git. The notes of every meeting are posted in the Discord channel ‘Notes’ and in the GIT. The notes will be posted there shortly after the meeting by Chiara. The Scrum board for task tracking and distribution is on Trello.

# Risk analysis

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Probability | Impact | Counter measures |
| The Morobot breaks | Low | Moderate |  |
| Members leave the group | Low | High | Be aware of all aspects of the project so we can redistribute work easily, and make sure that all code is uploaded to git. |
| Members do not attend meeting | Medium | Moderate | Talk about it as a group, then prevent it in the future. On repeated occurrence, enquire from the teacher what can be done. |
| Members do not follow deadlines | Medium | Moderate | Communicate in-between deadlines to assess progress and set realistic deadlines. |
| Lack of knowledge | Medium | High | Attain additional information with other groupmates; do additional research. |
| Losing code | Medium | High | Make sure everything is uploaded to git |
| Lack of motivation | Medium | Moderate | Set realistic goals for the project.  Support team members in case of a lack of motivation. |