Department of Computer Science & Engineering
The University of Texas at Arlington

System Requirements Specification
CSE 4316: Senior Design I
Fall 2020

Arm Solutions
Pack Man

Gregory Ferguson
Krishna Patel
Donny Pham
Inara Rupani
Mathew Zinke

Revision History

Revision	Date	Author(s)	Description
0.1	10.15.2020	IR	document creation
0.2	10.21.2020	IR, KP	Completed Safety(Section 6) and Other Requirements (Section 8)
0.3	10.25.2020	MZ	Completed Maintenance and Support (Section 7)
0.4	10.26.2020	GF, DP	Completed Customer (Section 3) and Performance Requirements (Section 5)
0.5	10.30.2020	IR, MZ, GF, DP, KP	complete draft
1.0	10.30.2020	IR, MZ, GF, DP, KP	official release

CONTENTS

List of Figures

1 Product Concept

The overlying idea of Arm Solutions is to reduce contact between people and packages. This will be achieved through an automated robotic arm that picks up packages and moves them to an area within the work cell.

1.1 Purpose and Use

The ongoing global pandemic is an issue that affects many aspects of society, including supply chains. Companies are adapting to new conditions to allow for social distancing and more hygienic work environments. The goal of Arm Solutions is to reduce contact between people and mediums of transmission. By automating parts the movement sorting of packages, exposure is reduced.

1.2 Intended Audience

The intended audience includes all major shipping and acquisitions companies. This includes not only postal companies, but companies that have high volumes of incoming and/or outgoing materials. Additionally, this can apply to companies it large amounts of internal mail. This reasoning is, because of Arm Solutions' goal to reduce personnel required to handle packages.

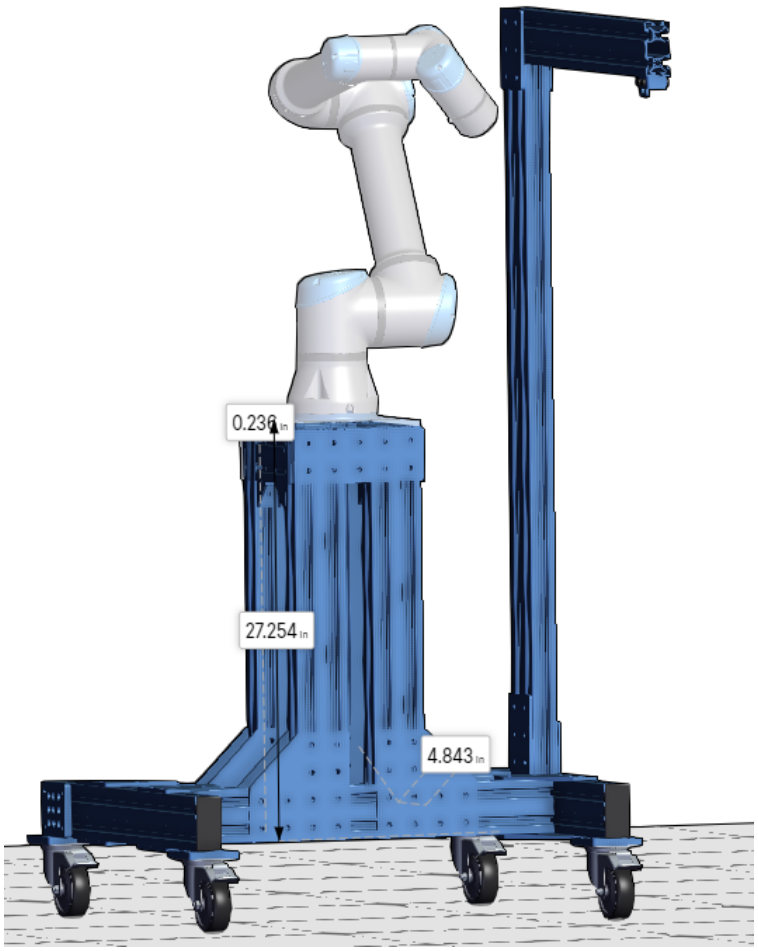

Figure 1: Conceptual drawing

2 Product Description

This section provides the reader with an overview of Arm Solutions robotic parcel sorting. The primary operational aspects of the product, from the perspective of end users, maintainers and administrators, are defined here. The key features and functions found in the product, as well as critical user interactions and user interfaces are described in detail.

2.1 Features & Functions

What the product does do:

- Scan, identify, and move parcels.
  - Using a Intel realsense (or similar depth sensor), identify the dimensions of the package.
  - Using a weight sensor, get the weight of the package.
  - Using a Zed Camera, scan the top of the package to identify what the package is.
  - Using data from the three sensors, choose the pickup points for the parcel.
- User interface to allow for human operators to take manual control of the system and define work cell rules.
- AI (machine learning model) to recognize parcels, search for packaging information, and placing parcels facing up.
- AI to sort parcels based on user defined work cell rules.

What the product does not do:

- Will not be able to pickup more than 5kg (11 pounds).
  - Due to safety factors, the arm will physically be able to pickup more than 5kg, but the product is designed for a maximum of 5kg.
- Will not be able to pickup parcels with dimensions greater than 12" x 12" x 12" (cubic foot)
- Will not be able to detect objects that are not parcels.
- Will not be able to automatically determine input regions and output regions of the work cell.

2.2 External Inputs & Outputs

Input/Output	Name	Description	Use
Input	Intel Realsense	Active IR Stereo	Identify package dimensions
Input	Zed Camera	3D camera with stereo vision	Identify package (reading labels)
Input	Weight Sensor	Sensor to identify weight	Help choose pick points for the arm
Input and Output	GUI	User Interface	Provide manual control for operators
Output	ROS	robot operation system	control and process sensor data to control robotic arm
Output	UR5	motors in UR5	move the arm to the destination.

2.3 Product Interfaces

The GUI will have a few panels. One will display what pick location is selected and one to display what place location is selected for the next object. When the user is in intervention mode the user can select each pick point for the objects in the scene.

3 Customer Requirements

The Robotic parcel sorting system is designed to picks parcel from a pile and place them neatly onto a conveyor. Our goal for this product is develop a more market-ready system that can be replicated to be installed in distribution centers where large volumes of unsorted and non-uniform parcels are processed. Below are the specified customer requirements (and other requirements that were not explicitly stated that need to be met to meet for the system to function as the customer wants). These requirements will not change unless agreed upon between the engineering team and the customer.

3.1 Tool Effector and Weight Handling

3.1.1 Description

A pneumatic suction cup tool effector (or similar) that will be able to handle up to 5 kilograms (with a safety factor of 2).

3.1.2 Source

Customer

3.1.3 Constraints

n/a

3.1.4 Standards

None

3.1.5 Priority

Critical

3.2 Package Handling

3.2.1 Description

Workcell (entire system) must be able to handle boxes and flat-ish parcels (12” x 12” x 12”).

3.2.2 Source

Customer

3.2.3 Constraints

Package can not be bigger than a cubic foot.

3.2.4 Standards

ASTM Standard D3951 - Standard Practice for Commercial Packaging

3.2.5 Priority

Critical

3.3 Depth Sensor

3.3.1 Description

A sensor to identify the dimensions of the packages (this is to be used by the picking software).

3.3.2 Source

Customer

3.3.3 Constraints

Placement of sensor may affect the results of the system depending of the dimensions of the package (idk)

3.3.4 Standards

None

3.3.5 Priority

Critical

3.4 RGB Camera

3.4.1 Description

A sensor to identify the packages and the pick and place points of the packages.

3.4.2 Source

Customer

3.4.3 Constraints

...

3.4.4 Standards

None

3.4.5 Priority

Critical

3.5 Picking Software

3.5.1 Description

Using data from the systems sensors, identify the pick points of the package.

3.5.2 Source

Customer

3.5.3 Constraints

The varieties in texture of parcel may cause difficulty inn 3D vision thus hindering the efficiency in picking up objects. In addition, deformed parcels sizes can also be cause of delaying the pickup software to activate

3.5.4 Standards

A specific version of software shall be used in order to ease porting to newer versions of ROS

3.5.5 Priority

Critical

3.6 Sorting Software

3.6.1 Description

Using data from the systems sensors, identify the place points of the package (similar to the picking software, this software tells the system where to place the package).

3.6.2 Source

Customer

3.6.3 Constraints

The type of object surfaces like (glossy, reflective, patterns or black) may be a problem for 3D vision and performing efficient sorting. In addition, placement of parcels in an unstructured way (in and overlapping form) could be hard to localize the objects.

3.6.4 Standards

A specific version of software shall be used in order to ease porting to newer versions of ROS

3.6.5 Priority

Critical

3.7 User Interface

3.7.1 Description

A user interface that allows a human to bolster the capabilities of the system when dealing with more complex shapes. This includes displaying pick location and place location. As well as allowing the user to select the pick points of the object.

3.7.2 Source

Customer

3.7.3 Constraints

A specific version of software shall be used in order to ease porting to newer versions of ROS

3.7.4 Standards

None

3.7.5 Priority

Critical

3.8 Movable Workcell

3.8.1 Description

Both the robotic arm and sensors should be attached together to a rigid body system, allowing for the entire system to be 'easily' movable.

3.8.2 Source

Customer

3.8.3 Constraints

System needs to be designed to fit within the senior design labs.

3.8.4 Standards

The design of workcell will depend on the specifications of UR5 and therefore make use of: UR5 Technical specifications

3.8.5 Priority

High

4 Packaging Requirements

This section entails the packaging requirements for UR5 parcel sorter. The parcel sorter will be modular in design. It will able to be moved to any desired spot customer or the team would like. The device after the development will be include an assembled UR5, user manual, separate end-effector used on the device, and a software package with installation instructions.

4.1 Software Packaging: ROS

4.1.1 Description

Since the UR5 is ran on ROS and it largely depends on it's ecosystem. Therefore all the software installations and configurations shall be consistent will ROS conventions. i.e custom software shall be written to manage dependencies

4.1.2 Source

Inara Rupani

4.1.3 Constraints

The software must be configure able that undesired pre-packaged software functionalities could be disabled and any complementary functions could be added.

4.1.4 Standards

ROS Standards can be found in the ROS Enhancement Proposals (REP)
REP-0136: http://www.ros.org/reps/rep-0136.html

4.1.5 Priority

High

4.2 Work Cell

4.2.1 Description

The work cell is a space built around the robotic arm. It is a standard environment that could be duplicated and mass produced for more parcel sorting arms. All the algorithms and additional hardware (end-effector and sensors) will be placed in this work cell around the arm.

4.2.2 Source

Inara Rupani

4.2.3 Constraints

The dimensions of this work cell needs to be determined and those will be major constraint for each piece

4.2.4 Standards

ASTM Standard D3951 - Standard Practice for Commercial Packaging

4.2.5 Priority

High

4.3 Software Packaging: Python

4.3.1 Description

Additional topics such as machine learning and computer vision tasks will be implemented in python. It is therefore, important for us to follow python conventions and packaging as possible.

4.3.2 Source

Inara Rupani

4.3.3 Constraints

A specific version of python shall be used in order to ease porting to versions of ROS

4.3.4 Standards

ROS Standards can be found in the ROS Enhancement Proposals (REP)
REP-0008 (Style Guide for Python): http://www.ros.org/reps/rep-0008.html

4.3.5 Priority

High

4.4 User Manual

4.4.1 Description

A user manual will be designed and added in the packages to help user understand the entities inside the package. It will include power up and any additional installation required for device.

4.4.2 Source

Inara Rupani

4.4.3 Constraints

n/a

4.4.4 Standards

n/a

4.4.5 Priority

Medium

5 Performance Requirements

The Pack Man's performance requirements deals with ensuring parcels reach their destination with minimal failure. This also addresses the product's ability to adapt and respond to users.

5.1 Robust Parcel Detection

5.1.1 Description

The Pack Man will recognize parcels in a region defined where incoming parcels will be.

5.1.2 Source

Customer

5.1.3 Constraints

Requires a robust computer vision system to minimize item recognition errors and scan for packaging information like recipient's name and address for processing.

5.1.4 Standards

None

5.1.5 Priority

High

5.2 Robust Parcel Handling

5.2.1 Description

The Pack Man will move the parcels to a region defined where parcels will be outputted.

5.2.2 Source

Customer

5.2.3 Constraints

Expects parcels to be in the shape of boxes or flat-padded rectangles. Expects parcels to weigh at most 11 pounds. Requires an algorithm to identify the best way to grab and hold parcels to minimize events of the product losing the package while processing. Requires an algorithm to detect if package is no longer in its grasp and regain it. Requires an algorithm to assist in the search of packaging information for the vision system to process.

5.2.4 Standards

None

5.2.5 Priority

High

5.3 GUI Responsiveness

5.3.1 Description

The Pack Man will have a GUI that will prioritize user input over certain actions in case of emergency or for readjustment.

5.3.2 Source

Project Owner

5.3.3 Constraints

If user commands the product to stop, the UR5 will immediately halt in any event. If user commands the product to reset, the UR5 will attempt to return its current parcel and go back to its starting position. If user commands the product to start, it will check the work space and begin operations. If users informs the product of work space change, it will finish dropping off the current parcel, update the work space, check the work space, and begin operations.

5.3.4 Standards

None

5.3.5 Priority

Critical

5.4 Workplace Flexibility

5.4.1 Description

The Pack Man will be able to adjust for customized and dynamic work environments.

5.4.2 Source

Project Owner, Customer

5.4.3 Constraints

If the product detects an unexpected object, halt operations in order to reduce workplace accidents and awaits user input to resume work. Users will be able to define the product's input, output, and barrier regions.

5.4.4 Standards

None

5.4.5 Priority

Low

6 Safety Requirements

This project will have a variety of hazards that need to be addressed when design the package sorting application.

6.1 Emergency Stop

6.1.1 Description

Due to the wide range of motion with the capacity to move 10 lbs of payload, if at anytime the user recognizes the robot behaving in an abnormal manner. This button is read and can be pressed to halt all operations.

6.1.2 Source

Product Owner, ARM SOLUTIONS

6.1.3 Constraints

The emergency stop mode shall not cause the whole system to restart if not needed. It should be able to resume operations before Emergency Stop Mode was activated.

6.1.4 Standards

UR5 User Manual

6.1.5 Priority

Critical

6.2 Watchdog

6.2.1 Description

Due to various communications between computer and robot, this will allow the system to auto reset or halt operations if the connection between robot and computer is broken. This watchdog will also enable auto reset if the robot is not responding to commands.

6.2.2 Source

Product Owner, ARM SOLUTIONS

6.2.3 Constraints

The watchdog timer should not erase the state of operations before an auto reset had occurred.

6.2.4 Standards

UR5 User Manual

6.2.5 Priority

Critical

6.3 Work Cell cage

6.3.1 Description

This will limit the proximity between the user and the robot. If a human needs to be close to the robot for any reason, this will limit the distance between them.

6.3.2 Source

Product Owner, ARM SOLUTIONS

6.3.3 Constraints

The cage should not limit the range of motion for the robot. The cage and robot should never come in contact with each other.

6.3.4 Standards

None

6.3.5 Priority

High

6.4 Payload Limits

6.4.1 Description

The UR5 robot has a limit of 5Kg. Anything beyond this will hinder the range of motion and or damage the robot.

6.4.2 Source

UR5 Manual

6.4.3 Constraints

This weight limit needs to take inconsideration of other modules or sensors being fitted on the robot

6.4.4 Standards

UR5 User Manual

6.4.5 Priority

High

6.5 Non-Sharp Edges

6.5.1 Description

Sharp edges will potential damage packages or other workers nearby.

6.5.2 Source

Project Owner ARM SOLUTIONS

6.5.3 Constraints

No constraints currently

6.5.4 Standards

None

6.5.5 Priority

High

7 Maintenance & Support Requirements

For proper future usage of the package sorting mechanism, the following requirements shall be implemented before and during usage. They concern hardware, software, and the work area. Some of the following is more critical that others, but all should be followed.

7.1 All parts of the process shall be run on one platform

7.1.1 Description

Ease of updating and part communication, all software (excluding firmware) is to be run on a single device. Proposed devices are Raspberry Pi 4 and NVIDIA Jetson. Because of its being opensource, Linux is the preferred platform.

7.1.2 Source

ARM Solutions

7.1.3 Constraints

Inter-platform communication issues purchasing power of team

7.1.4 Standards

Using one platform (EG Raspberry, Linux Desktop, etc) should be able to consolidate process

7.1.5 Priority

High

7.2 User Manual

7.2.1 Description

A user manual will be used for packaging, transportation, working, and maintenance of the product.

7.2.2 Source

ARM Solutions

7.2.3 Constraints

N/A

7.2.4 Standards

N/A

7.2.5 Priority

Medium

7.3 Basic maintenance of the UR5 shall be followed

7.3.1 Description

Maintenance of the UR5 robot is based on suggested maintenance of the UR5 manual. This is to ensure longevity and safety of robotic arm.

7.3.2 Source

Universal Robots, Product Owner

7.3.3 Constraints

Team allowance of maintenance of the robot

7.3.4 Standards

UR-5 robotic standards

7.3.5 Priority

High

7.4 The workcell shall be cleared of all debris before usage

7.4.1 Description

A simple cleaning of debris (paper, trash, etc) to ensure no interference with the robot

7.4.2 Source

Product Owner

7.4.3 Constraints

N/A

7.4.4 Standards

N/A

7.4.5 Priority

Medium

7.5 The workcell shall be checked before usage

7.5.1 Description

Check the connections alignment of devices before each day's usage of the workcell to ensure proper execution of tasks.

7.5.2 Source

ARM Solutions

7.5.3 Constraints

N/A

7.5.4 Standards

N/A

7.5.5 Priority

Medium

7.6 The system shall be powered through standard A/C sockets

7.6.1 Description

Each device should have a standard (non-custom) plug, which can be powered into a commonly used socket.

7.6.2 Source

ARM Solutions, Device Manufacturer, Product Owner

7.6.3 Constraints

Device Manufacturing

7.6.4 Standards

IEC 60364

7.6.5 Priority

High

7.7 All libraries shall be updated through the project repo

7.7.1 Description

To ensure that library updates do not interfere with operation, they shall only be updated through the project repository.

7.7.2 Source

ARM Solutions, Project Owner

7.7.3 Constraints

Device OS access

7.7.4 Standards

N/A

7.7.5 Priority

Medium

8 Other Requirements

This section contains requirements that are important for the project for it's development. For the project to keep up-to date and for customers to easily reuse the components. These are the requirements that do not fit well with other required requirements. We often call these requirements tech related or as "non functional requirements."

8.1 Robot Operating System

8.1.1 Description

The control computer shall run on ROS MoveIt

8.1.2 Source

1. Client (James Staud - FT Designs)
2. https://wiki.ros.org/universal_robot

8.1.3 Constraints

As our client have told us that other robots that they designed are able to run and work on that OS, we are also tasked to make use of that Operating System.

8.1.4 Standards

n/a

8.1.5 Priority

High

8.2 Software Application

8.2.1 Description

The team shall install a software application on a linux PC to move the robot arm.

8.2.2 Source

Client (James Staud - FT Designs)

8.2.3 Constraints

There are two constraints:
1. The team shall have access to Linux PC
2. The application might be accessible to use on a certain version of Linux.

8.2.4 Standards

n/a

8.2.5 Priority

High

8.3 Computer Vision

8.3.1 Description

The Team will use an Intel Realsense Camera for visioning of objects on the conveyor belt

8.3.2 Source

Client (James Staud - FT Designs)

8.3.3 Constraints

This requirement is specifically described by the customer, but it also depends on our budget. The team could directly request from customer, or CSE department could provide us or we could make changes in our budget to make use of specific product.

8.3.4 Standards

n/a

8.3.5 Priority

High

8.4 Scalability of Parts used

8.4.1 Description

All the parts of process shall run on one platform

8.4.2 Source

Mathew Zinke

8.4.3 Constraints

This requirement is placed to prevent inter-platform communication issues.

8.4.4 Standards

Using one platform (EG Raspberry, Linux Desktop, etc) should be able to consolidate process

8.4.5 Priority

High

9 Future Items

In this last section, you will reiterate all requirements that are listed as priority 5. This is repetitive, but necessary as a concise statement of features/functions that were considered/discussed and documented herein, but will NOT be addressed in the prototype version of the product due to constraints of budget, time, skills, technology, feasibility analysis, etc. Use the following format for this section.

9.1 Autonomous Dispatch Mode

9.1.1 Description

In a large warehouse, there would be multiple work cells in various locations. This can take a lot of people to properly place around specific conveyors. To reduce the set-up time, work cells can be dispatched to particular areas of the warehouse from one location or specific hub. Since the work cell is modular, the wheels can be motorized and controlled over a network to move work cell as needed.

9.1.2 Source

Product Owner, ARM SOLUTIONS

9.1.3 Constraints

Adding motorized wheels will increase cost. This will also require modification of software and design of work cell.

9.1.4 Standards

Work cell autonomous drive mode needs to be self-aware. Robot Arm should be locked in placed when in motion.

9.1.5 Priority

Priority Medium

9.2 Weight sensor for packages

9.2.1 Description

A weight sensor that allows the system to identify the weight of the packages.

9.2.2 Source

Gregory Ferguson (for now)

9.2.3 Constraints

Dimensions of packages may affect the sensor (I'm not sure)

9.2.4 Standards

None

9.2.5 Priority

Future

9.3 AI Integration

9.3.1 Description

Machine learning system (likely a CNN) to identify different labels to perform a sorting operation. While this is similar to the sorting software, this requirements goal is to automate that process.

9.3.2 Source

Customer

9.3.3 Constraints

A single scan shall scan the objects placement and signals the robot to pickup the object. Time delay during this processes could be a constraint

9.3.4 Standards

None

9.3.5 Priority

Future

9.4 Eco Mode

9.4.1 Description

The movement of robotic arm shall be able to adjust based on the amount of parcels scanned in the chute.

9.4.2 Source

Gregory Ferguson

9.4.3 Constraints

No constraints currently

9.4.4 Standards

UR5 Technical specifications

9.4.5 Priority

Low

REFERENCES