Software Requirements Specification
Mechtronics Enigeering

Team 32, Wingman
Edward He
Erping Zhang
Guangwei Tang
Peng Cui
Peihua Jin

Contents

Table 1: Revision History

Date	**Developer(s)**	**Change**
2022-10-04	Edward He, Erping Zhang Guangwei Tang, Peng Cui Peihua Jin	Revision 0

List of Tables

List of Figures

This document describes the requirements for Mechtronics Enigeering. The template for the Software Requirements Specification (SRS) is a subset of the Volere template **?**. If you make further modifications to the template, you should explicitly state what modifications were made.

1 Project Drivers

1.1 The Purpose of the Project

1.2 The Stakeholders

1.2.1 The Client

1.2.2 The Customers

1.2.3 Other Stakeholders

1.3 Mandated Constraints

1.4 Naming Conventions and Terminology

1.5 Relevant Facts and Assumptions

User characteristics should go under assumptions.

1.6 The Scope of the Work and the Product

1.6.1 The Context of the Work

1.6.2 Work Partitioning

1.6.3 Individual Product Use Cases

2 Behavior Description

2.1 Assumptions and Dependencies

AD1: The user can perform simple computer opertions.
Rationale: The user can type, moving mouse and clicking on the window.

AD2: The user can understand English.

AD3: The operation of the project should take place indoors with adequate lighting.

AD4: All objects can be seen by the product without hiding.

AD5: Every motion of the user can be seen by the device.

2.2 Finite State Machine Description

To make the behaviour of the product to achieve the target task, the Finite State Machine is created to describe the behaviour with detailed description provided after the picture.

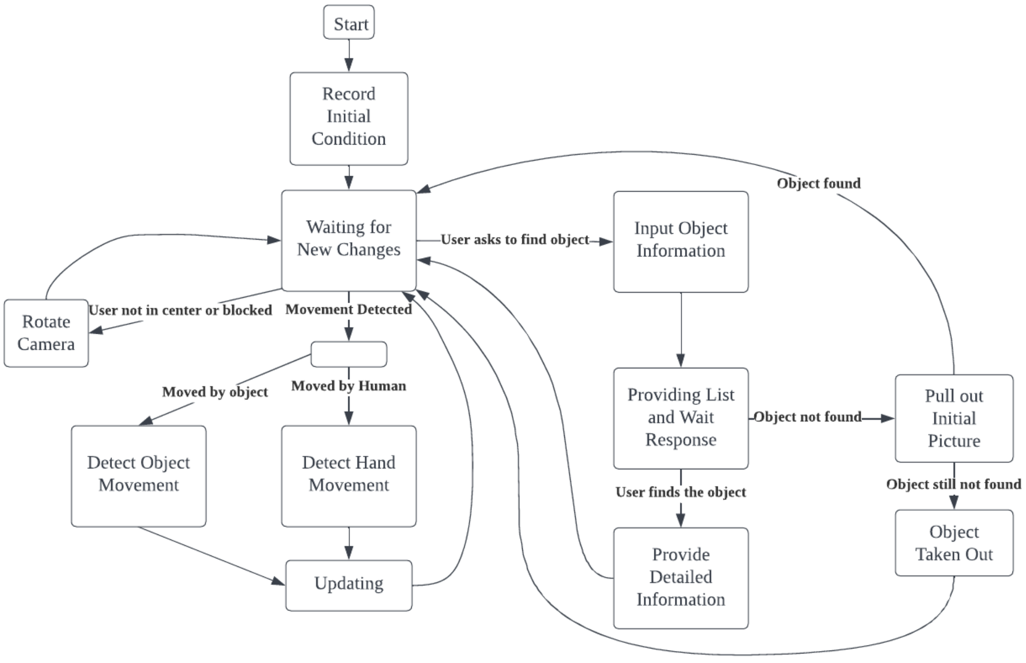

Figure 1: The Picture of Finite State Machine

Record Initial Condition: The device will take a picture of the room as the initial position information for each object and save it to the database.

Waiting for New Changes: The device will wait for future operations made by human or changes of objects detected by the camera.

Detect Object Movement: When the device detects the movement of an object that is made by another object (for example, colliding with another object and fall), it will trace the new location oof the object.

Detect Hand Movement: When the device detects the movement of the user, it will track the movement of the hand to identify which object is being moved and remember its final location.

Updating: The device will record the updated information (including time, Location, picture of object, and so on) of the object into the database.

Input Object Information When the user want to find one object in the room, the user interface will ask the user to input information of the object (like size, color, last time saw it, and so on).

Providing List and Wait Response: After the user has input the information, the system will provide a list of objects with pictures that satisfy the input information.

Provide Detailed Information: WHen the user find the target object, the device will display the detailed information about the object like showing the current location on the screen.

Pull Out Initial Picture: When the object is not found in the list, the devide will display the picture taken just after the program starts to let the user find the object.

Object Taken Out: If the object is still not found, the system will think that the object is taken out of the room or is hidden behind another object.

Rotate Camera: If the device found the human body detected is not in thhe center of the screen or the camera is covered by something else, it will send signals to the motor to rotate the camera until problem sloved.

2.3 Use Case Diagram

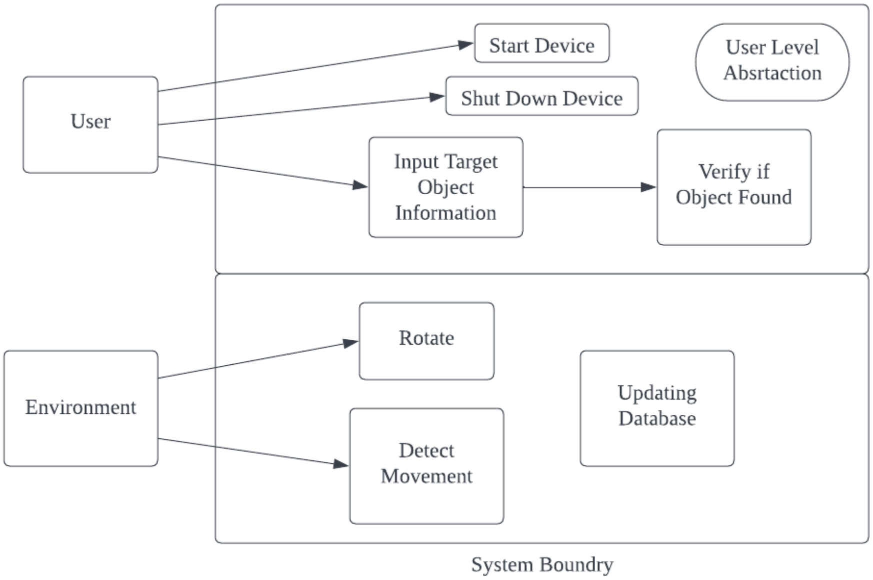

Figure 2: The Picture of Use Case Diagram

The Use Case Diagram shown above describes some actions that the user can interact with teh system. The line in the middle shows the boundaries between user and the internal program. The outmost large box shows the boundary for the whole system. All user can do is just to start and shut down the program. Th user can also find the target object by typing the information about the object and verify if the object is found. The system will interact with the environment by rotating its camera and detecting the movement of the object in the room. It will also ba able to update its database when the position of the object is changed.

3 Functional Requirements

The following are the functional requirements of the project. They are separated into 2 main parts: Image Processing and Storage, and UI Interface Menu.

3.1 Image Processing and Storage Requirements

IPR1: The device should be able to identify human's body.

IPR2: The device must be able to identify human's hand.

IPR3: The device must be able to take a photo automatically when required.

IPR4: The device should be able to identify all the small items being exposed to the camera.

IPR5: The device should be able to take a photo once the change of the location of an item is captured.

IPR6: The device should be able to differentiate one item from another items which are identified by the system through 3 main parameters,item_shape, item_color and item_size.

IPR7: The device must be able to store all the photos into a file, and indicate the time when it was taken.

IPR8: The device must be able to name each item with a unique ID.

IPR9: The device should be able to arrange the photos stored in the file in ascending or descending order according to the time it was taken.

IPR10: The device should be able to arrange the photos stored in the file in ascending or descending order according to their IDs.

3.2 UI Interface Menu Functional Requirements

UIR1: The UI should be able to let user to choose whether to highlight a certain item or not.

UIR2: The UI should be able to let user to switch the ordering method.

UIR3: The UI must be able to notify the user when the WIFI signal is weak or unstable.

UIR4: The UI must be able to allow the user to view the system's status at any given point in time.

4 Nonfunctional Requirements

The next paragraphs will talk about about nonfunctional requirements in the designing of the smartVault, which will be discussed in several different parts.

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

APR1: The device should not have exposed internal electronic wiring.

APR2: The sharp corners should not be exposed to the users annd should be covered by some soft materials.

4.1.2 style

Not Applicable.

4.2 Usability and Humanity Requirements

4.2.1 Easy of Use Requirements

EUR1: The device should be easy to installed in the room.

EUR2: Fonts, blanks and graphics shown on the screen should be big and visiable enough for the user to use without inserting wrong information.

4.2.2 Personalization and Internationalization Requirements

Not applicable.

4.2.3 Learning Requirements

LR1: The software part of the product should be easy to install.

LR2: The product should be easy to set up in its working environment.
Rationale: The device can identify objjects in the room in a short tiime after powering up.

4.2.4 Understandability and Politeness Requirements

UPR1: The graphics used should be readeable and visiable to the user.
Rationale: Easy for clients to find objects that wish to find in the room.

4.3 Accessibility Requirements

AR1: The display window should be concise enough for user to understand

4.4 Performance Requirements

4.4.1 Speed and Latency Requirements

SLR1: The response time of the product to show the location of the object should be less than 5 seconds.

4.4.2 Safety-Critical Requirements

SCR1: The base of the device should be strong enough without falling from the floor.

SCR2: The device should not infringe on user's privacy.

SCR3: The rotating speed of the motor should be slow enough without hurting people.

4.4.3 Precision and Accuracy Requirements

PAR1: The time based value used in the device should be precised to minute.

PAR2: The location value used in the devce should be precised to whole number.

PAR3: Other numatic values used in the device should be rounded to one decimal place.

4.4.4 Reliability and Avaliability Requirements

RAR1: The device should not over-rotate to an unexpected angle.

RAR2: The device should be avaliable to work for the whole time of except for maintaince or updating time.

4.4.5 Robustness or Fault-Tolerance Requirements

RFR1: The data stored in the file should not be deleted or changed even after the program shuts down.

RFR2: The device should be able to notify user for an error occurs in the program.
Rationale: The user should be notified if he or she has a wrong input or havin inappropriate actions.

4.4.6 Capacity Requirements

CR1: The device sould be able to store the any information about the location of each object detected from the camera.

4.4.7 Scalability or extensibility Requirements

Not applicable.

4.4.8 Longevity Requirements

LR1: The device should store the information about the objects until it is changed into a different environment.

4.5 Operational and Environmental Requirements

4.5.1 Expected Physical Environment

EPE1: The device is supposed to work in any indoor space.

4.5.2 Requirements for Interfacing with Adjacent Systems

Not applicable.

4.5.3 Productization Requirements

Not applicable.

4.5.4 Release Requirements

No applicable.

4.6 Maintainability and Support Requirements

4.6.1 Maintenance Requirements

MR1: The maintenance for the device should be done by the developers.

4.6.2 Supportability Requirements

SR1: The device is supported by any computers supports both C and python programming languages.

4.6.3 Adaptability Requirements

Not Applicable.

4.7 Security Requirements

4.7.1 Access Requirements

AR1: Any one except for the users is not allowed to access the any file that stores the information about the objects in the room. **AR1:** Any one except for the users is not allowed to access the any file that stores the information about the objects in the room.

4.7.2 Integrity Requirements

IR1: Data in teh files should not be changed unnecessarily.

IR2: The files are locked whhen teh device shuts down.

4.7.3 Privacy Requirements

PR1: Other users are not allowed to access any file in the computer.
PR2: The camera should only work for a specific user.

4.7.4 Audit Requirements

Not applicable.

4.7.5 Immunity Requirements

Not applicable.

4.8 Cultural and Political Requirements

4.8.1 Cultural Requirements

Not applicable.

4.8.2 Plitical Requirements

Not applicable.

4.9 Legal Requirements

4.9.1 Compliance Requirements

CR1: The performance of the product should not violate the laws that protect the privacy of thee user.

4.9.2 Standards Requirements

Not applicable.

5 Project Issues

5.1 Open Issues

- Limit of 180 rotation degree of servo motor

- Accuracy of object detection
- How to distinguish two objects with very limited resources
- How to recognize the same object in different angles
- How to guarantee the stability of the serial communication

5.2 Off-the-Shelf Solutions

- Huskylens - is a AI camera which can learn new objects and recognize them. It has the machine learning technology enables projects to interact with people and environments which allows many kinds of system control.
- NVIDIA Jetson TX2 - is an embedded AI computer device. It has 8GB memory and 59.7GB/s of memory bandwidth which provides a high quality AI performance to build efficient AI models including computer vision.

5.3 New Problems

5.3.1 Effects on the Current Environment

Any changes to the exist database may cause the data related to each object missing

5.3.2 Effect on the Installed Systems

Changes to the motorized camera mount will affect the algorithm or logic of the controller board

5.3.3 Potential User Problems

Changes to the user interface may change the way that user used to search the object

5.3.4 Limitations in the Anticipated Implementation Environment

NA

5.3.5 Follow-Up Problems

The changes in computer vision algorithm may cause the whole system malfunction

5.4 Tasks

5.4.1 Project Planning

NA

5.4.2 Planning of the Development Phases

NA

5.5 Migration to the New Product

5.5.1 Requirements for Migration to the New Product

- All the objects data should be stored
- Motorized camera mount should be calibrated before using
- The spec of camera should be kept in same as possible

5.5.2 Data that Has to be Modified or translated for the new system

- Objects data need to be transfer ed into the new system

5.6 Risks

- Connection lost between the board and camera during the motor movement.
- Inappropriate distance measure and control between the object and the motor which furthermore cause damage or stuck.
- Physical damage from collision with objects .
- Physical damage from wire twisting during rotation.
- Unexpected movement caused by the delay of data transfer

5.7 Costs

Product	Price
USB Camera	$30
PTZ Mount	$25
Arduino	$30
Motors	$15
Total	$100

5.8 User Documentation and Training

5.8.1 User Documentation Requirements

- User manuals
- Installation manuals
- Technical specifications to accompany the product

5.8.2 Training Requirements

NA

5.9 Waiting Room

- Expanding chassis's activity area including rotation angle and planar movement.
- Developing new algorithm regarding data transfer to enable faster real-time reaction.
- Adding alarm in case that object not found in the assigned area as an application of storage security.

5.10 Ideas for Solutions

- Use DC motor with decoder to implement the unlimited rotation degree of camera mount

- Use Field Oriented Control algorithm to implement the unlimited rotation degree of camera mount

- Use frame-to-frame compassion to detect the relocation of objects

- Predict the possible location of objects by tracking the users path in the room

- Beeper alert when the camera view is not cleared

6 Appendix

This section has been added to the Volere template. This is where you can place additional information.

6.1 Symbolic Parameters

The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance.

6.2 Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

2. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?