Department of Computer Engineering

CENG350 Software Engineering

Software Requirements Specification for FarmBot

Group 52

By

Emre Çam

Tufan Özkan

Monday 18th March, 2024

Contents

List of Figures

List of Tables

Revision History

(Clause 9.2.1)

You can use a table to show your reports' versions and their date.

To create tables, you can use https://www.latex-tables.com/; the Table 1 is generated by it.

A	B	C	D

Table 1: Table Example

1. Introduction

For further explanation and details, please, turn to the referred clauses of IEEE 29148-2018, the highlighted and commented version; pay attention to the comments. Clause numbers are written in *italic* in the sections.

Refer to *(Clause 9.6.1)*.

1.1 Purpose of the System

The purpose of the FarmBot system is to revolutionize agricultural practices through automation. FarmBot aims to regularize farming methods, enhance crop yields and minimize resource waste. Additionally; FarmBot provide a platform to manage and monitor their crops and agricultural operations efficiently. Through automation and monitoring, the system aims to:

- Automate planting, watering, weeding and harvesting processes to reduce human work and improve efficiency.
- Provide precise control over the environmental factors such as soil moisture, harmful weeds.
- Facilitate real-time monitoring from web application to enhance crop health.
- Provide an environment to design a layout for crops and implement that design in real world.

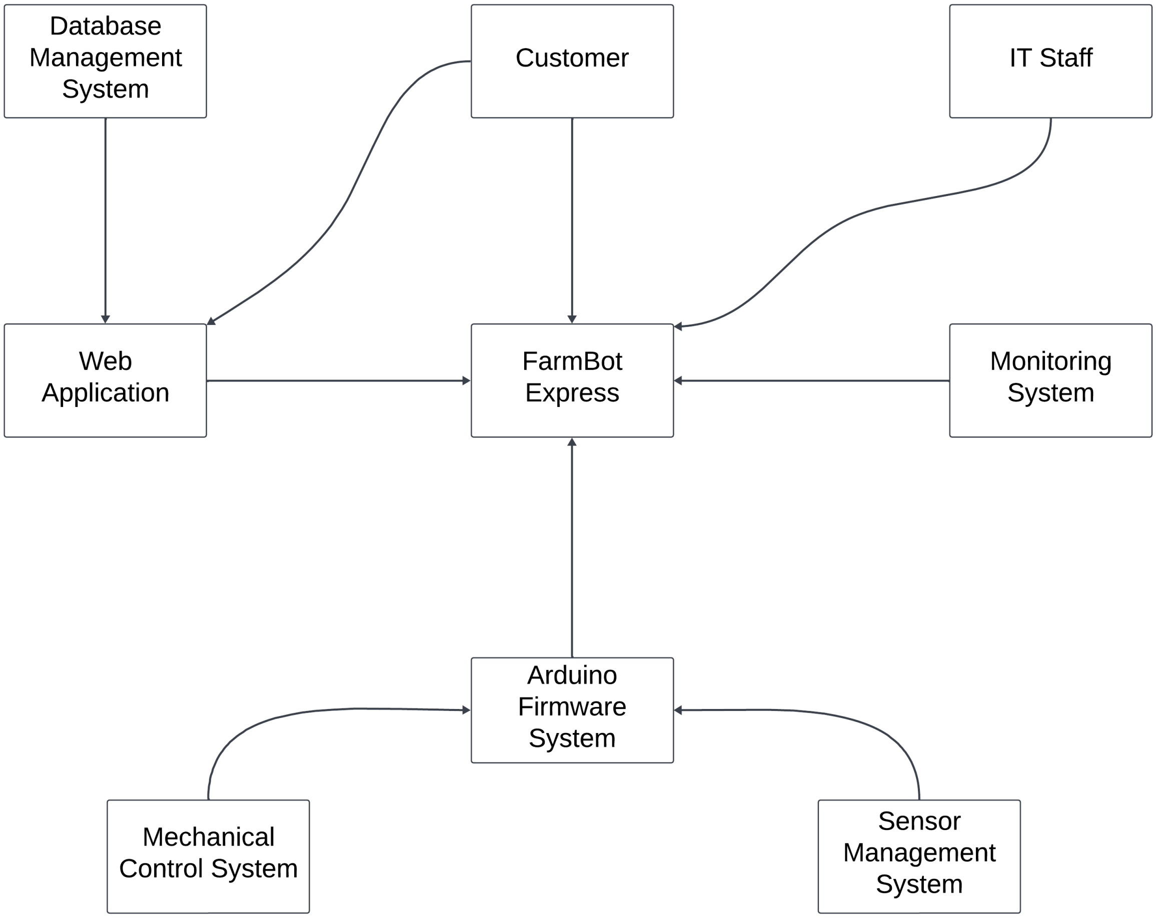

Figure 1.1: Context Diagram of FarmBot Express

1.2 Scope

Refer to *(Clause 9.6.3, 9.5.3, 9.4.3)*

1.3 System Overview

1.3.1 System Perspective

Refer to *(Clause 9.6.4, 9.5.4.1)*

1.3.1.1 System Interfaces

Refer to *(Clause 9.6.4.1)*

1.3.1.2 User Interfaces

Refer to *(Clause 9.6.4.2)*

1.3.1.3 Hardware Interfaces

Refer to *(Clause 9.6.4.3)*

1.3.1.4 Software Interfaces

Refer to *(Clause 9.6.4.4)*

1.3.1.5 Communication Interfaces

Refer to *(Clause 9.6.4.5)*

1.3.1.6 Memory Constraints

Refer to *(Clause 9.6.4.6)*

1.3.1.7 Operations

Refer to *(Clause 9.6.4.7)*

1.3.2 System Functions

Refer to *(Clause 9.6.5, 9.5.4.2)*. If you want to add any figure or diagram, you can use a figure environment.In Figure 1.2

1.3.3 Stakeholder Characteristics

Refer to *(Clause 9.6.6, 9.5.4.3, 9.4.5)*

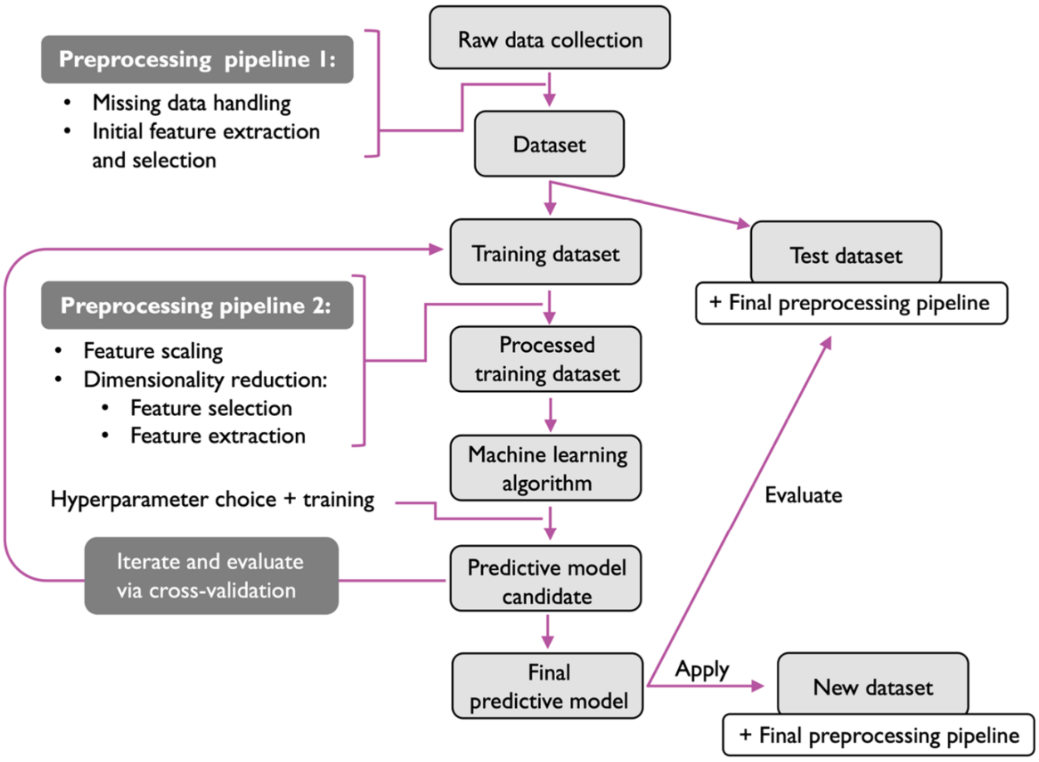
Raw data collection
Preprocessing pipeline I:
Missing data handling
Initial feature extraction and selection
Dataset
Training dataset
Test dataset
+ Final preprocessing pipeline
Preprocessing pipeline 2:
Feature scaling
Dimensionality reduction:
Feature selection
Feature extraction
Processed training dataset
Machine learning algorithm
Hyperparameter choice + training
Evaluate
Iterate and evaluate via cross-validation
Predictive model candidate
Final predictive model
Apply
New dataset
+ Final preprocessing pipeline

Figure 1.2: Example

1.3.4 Limitations

Refer to *(Clause 9.6.7)*

1.4 Definitions

You should add acronyms and abbreviations here. Refer to *(Clause 9.6.7)*

For any citation, refer to it as [1].

2. References

[1] M. C. Younis and H. Abuhammad, “A hybrid fusion framework to multi-modal biometric identification,” *Multimedia Tools and Applications*, vol. 80, no. 17, pp. 25799–25822, 2021.

3. Specific Requirements

Refer to *(Clause 9.6.10)*.

3.1 External Interfaces

External Interfaces Class Diagram and its explanations go here. Plus, other content as appropriate

Refer to *(Clause 9.6.11, 9.5.8)*.

3.2 Functions

Use-case diagram goes here; **detailed use-case descriptions in a reasonable template** follow. You are expected to have **about 10 use cases covering major system functionality**. Have some associations in your use-case diagram, e.g. include, extend, specialization. Choose *three* most complicated use cases. Construct three diagrams **(one sequence diagram, one activity diagram, and one state diagram)** to elaborate on these three use cases. Plus, other content as appropriate.

Refer to *(Clause 9.6.1)*.

3.3 Logical Database Requirements

Key data objects (persistent or not) and their major attributes. Draw the **Class Diagram** with associations. A class dictionary can be omitted, provided that the naming is understandable.

Refer to *(Clause 9.6.15)*.

3.4 Design Constraints

Specify constraints on the system design imposed by external factors, such as official standards, regulatory requirements, or organizational/managerial limitations.
Refer to *(Clause 9.6.16)*.

3.5 System Quality Attributes

Important quality attributes (Usability *(Clause 9.6.13, 9.5.6)*, Performance *(Clause 9.6.14, 9.5.7)*, Dependability properties, Maintainability, and so on) in the order of priority with associated requirements.

Refer to *(Clause 9.6.18)*.

3.6 Supporting Information

Refer to *(Clause 9.6.20)*

4. Suggestions to Improve The Existing System

4.1 System Perspective

Context diagram and explanations of context diagram go here for suggestions to improve the existing system. Plus, other content as appropriate.

Refer to *(Clause 9.6.4, 9.5.4.1)*.

4.2 External Interfaces

External Interfaces Class Diagram and its explanations go here for suggestions to improve the existing system. Plus, other content as appropriate

Refer to *(Clause 9.6.11, 9.5.8)*.

4.3 Functions

Use-case diagram for suggestions to improve the existing system goes here; **detailed use-case descriptions in a reasonable template** follow. You are expected to have **about 4 use cases covering suggestions to improve the existing system**. Have some associations in your use-case diagram, e.g. include, extend, specialization. Choose three most complicated use cases. Construct three diagrams **(one sequence diagram, one activity diagram, and one state diagram)** to elaborate on these three use cases. Plus, other content as appropriate.

Refer to *(Clause 9.6.12, 9.5.5, 9.5.10)*.

4.4 Logical Database Requirements

Key data objects (persistent or not) and their major attributes for suggestions to improve the existing system. Draw the **Class Diagram** with associations. A class dictionary can be omitted, provided that the naming is understandable.

Refer to *(Clause 9.6.15)*.

4.5 Design Constraints

Specify constraints on the system design imposed by external factors, such as official standards, regulatory requirements, or organizational/managerial limitations for suggestions to improve the existing system.

Refer to *(Clause 9.6.16)*.

4.6 System Quality Attributes

Important quality attributes (Usability *(Clause 9.6.13, 9.5.6)*, Performance *(Clause 9.6.14, 9.5.7)*, Dependability properties, Maintainability, and so on) in the order of priority with associated requirements for the improved system.

Refer to *(Clause 9.6.18)*.

4.7 Supporting Information

Refer to *(Clause 9.6.20)*.