Software Requirements Specification

for

Fire detection system

Name	ID
FAHAD ZAID HAMZI	202208800
KHALID YAHYA ALFAIFI	200887040
AHMED AFGHANI	202208980

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Khalid Alfaifi	23/4/2023	Add user story	V2

1. Introduction

1.1 Purpose

The purpose of this document is to define the Software Requirements Specification (SRS) for a fire detection system that connects an ESP 32 microcontroller with two sensors, temperature, and gas detector. The system is designed to detect fires and send alarms to a platform in the cloud. The microcontroller will send notifications every hour to indicate its status. During a fire, the system will send notifications every 5 seconds. There is also a maintenance mode that sends notifications every 20 minutes.

1.2 Document Conventions

This document will follow the IEEE 830-1998 standard for software requirements specification.

1.3 Intended Audience and Reading Suggestions

This document is intended for the development team responsible for building the fire detection system. The document can also be used by stakeholders, such as project managers, quality assurance personnel, and other interested parties, to gain an understanding of the system requirements.

1.4 Product Scope

The fire detection system is designed to detect fires using two sensors, temperature, and gas detector. The system will send alarms to a cloud-based platform when a fire is detected. The microcontroller will send notifications every hour to indicate its status. During a fire, the system will send notifications every 5 seconds. There is also a maintenance mode that sends notifications every 20 minutes.

1.5 References

No references are required for this document.

2. Overall Description

2.1 Product Perspective

The fire detection system is a standalone system that connects an ESP 32 microcontroller with two sensors, temperature, and gas detector. The system is designed to detect fires and send alarms to a platform in the cloud.

2.2 Product Functions

The system shall provide the following functions:

Detect fires using two sensors, temperature, and gas detector.
Send alarms to a platform in the cloud when a fire is detected.
Send notifications every hour to indicate the status of the microcontroller.
Send notifications every 5 seconds during a fire.
Send notifications every 20 minutes during maintenance mode.

2.3 User Classes and Characteristics

The system is designed for users who require a fire detection system that can be remotely monitored. The users should have basic knowledge of the system's operation and be able to use the platform in the cloud.

2.4 Operating Environment

The system shall operate in the following environments:

Temperature range: 0 to 50 degrees Celsius
Humidity range: 20 to 80 percent
Altitude: up to 2000 meters above sea level

2.5 Design and Implementation Constraints

The system must be designed and implemented using the ESP 32 microcontroller and the two sensors, temperature, and gas detector. The system must also be designed to be scalable and easily maintainable.

2.6 User Documentation

The system shall be accompanied by user documentation that describes how to use the system and the platform in the cloud.

2.7 Assumptions and Dependencies

The system assumes that the sensors, temperature, and gas detector are functioning correctly.
The system depends on the availability of the platform in the cloud.

3. External Interface Requirements

3.1 User Interfaces

The system shall provide a user interface that allows users to view the status of the system and the notifications sent by the microcontroller.

3.2 Hardware Interfaces

The system shall connect to the ESP 32 microcontroller and the two sensors, temperature, and gas detector.

3.3 Software Interfaces

The system shall interface with the platform in the cloud to send alarms and notifications.

3.4 Communications Interfaces

The system shall communicate with the platform in the cloud using Wi-Fi.

4. Requirements

- ***As a** user, **I want** the ESP 32 system to monitor the temperature and gas levels, **so that** I can be alerted in case of a fire.*
- ***As a** user, **I want** to receive an alarm notification in case a fire is detected, **so that** I can take appropriate action.*
- ***As a** user, **I want** the ESP 32 system to send a wake notification to the cloud platform every hour, **so that** I can be reassured that the system is functioning properly.*
- ***As a** maintenance personnel, **I want** to put the ESP 32 system into maintenance mode, **so that** I can perform maintenance tasks without triggering false alarms.*
- ***As a** maintenance personnel, **I want** the ESP 32 system to send a maintenance notification to the cloud platform every 20 minutes during maintenance mode, **so that** I can monitor the status of the system.*
- ***As a** developer, **I want** to design the ESP 32 system **so that** it is reliable and accurate in detecting fires, and can handle communication with the cloud platform without errors.*
- ***As a** developer, **I want** to design the ESP 32 system with error handling and logging capabilities, **so that** I can troubleshoot any issues that may arise.*

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The system shall detect fire within 5 seconds of the threshold being exceeded and send the alarm notification within 2 seconds of the fire detection.

5.2 Safety Requirements

The system shall be designed and implemented to minimize the risk of harm to users, environment, and property.

5.3 Security Requirements

The system shall ensure the security of the communication channel with the cloud platform using encryption and authentication mechanisms.

5.4 Software Quality Attributes

The system shall be reliable, maintainable, and usable.

5.5 Business Rules

The system shall comply with all relevant regulations and guidelines related to fire safety.