

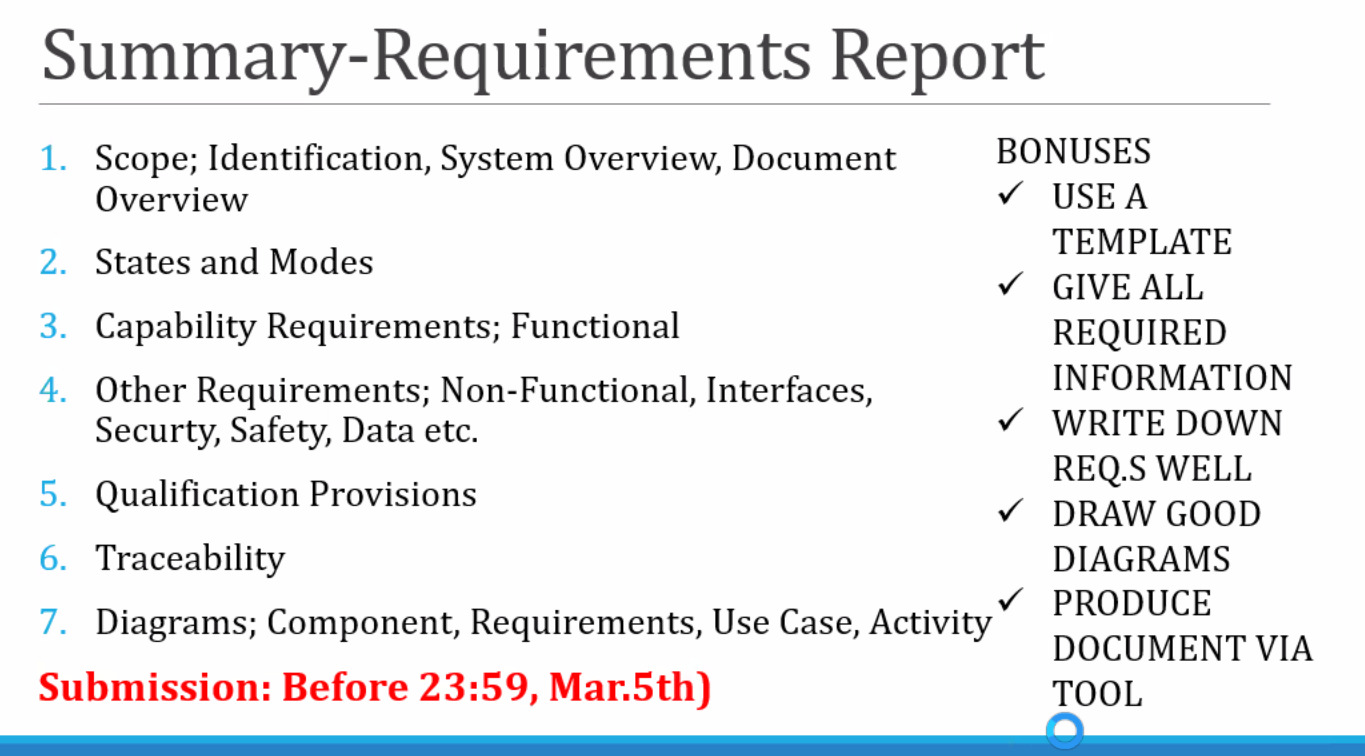
CTIS - 477 System Engineering Project

Project Requirements



**Ömer Levent Durdalı**

**21702600**



Check

<https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database>

<http://web.cse.ohio-state.edu/~bair.41/616/Project/Example_Document/Req_Doc_Example.html>

<http://www.ofnisystems.com/services/validation/user-requirement-specifications/>

Edge Onetab

# Introduction

The projects name is, Smart Life and Home Security System, this document will refer to the project with following abbreviation; **SLS**.

SLS is a small Smart Home System that manages and tracks curtains, lights, temperature and

motion in a given area.

# Scope

## System Overview

SLS’s main purpose is to be a Decision support system for a home user, meaning the system can help with managing room temperature, adjust light levels and lastly helps the user to adjust the curtains in the specified room. The system will have 3 different sensors that is connected to a central hub. This hub will communicate with a personalized cloud server using the MQTT communication protocol. The end-user will be able to see the gathered data and control the light level and curtain positions from the local web dashboard panel.

## Document Overview

This document will specify, project requirements, work flow, system capabilities, system use environment, resources, and constraints of the SLS.

SLS does not collect any personal data so there are no privacy concerns regarding its use.

## Literature and Referenced Documents

There are no open-source or closed-sourced Home automation system that have the same functionally as SLS, but there are lots of different competitors in regards to its specified and different abilities some are listed below.

* **Smart Curtains:**
* Turns existing curtains to a Smart Curtain - <https://slide.store/>
* Smart Home Blinds <https://getkeego.com/>
* Spikebot Curtain Device - <https://spikebot.io/spikebot-curtain-device/>
* **Smart Home Light:** Philips Hue lights and Xiaomi Mi Smart lights.
* Smart Motion Detectors: Bosch Motion Detectors and SONOF Motion Sensors.
* **Software:** This software is used to combine different smart home automation vendors.
* OpenHAB, Open-Source Home Automating Software - <http://www.openhab.org/>
* PiDome, - <https://pidome.org/>

# Requirements

## Functional Requirements

* All sensors shall be able to read its intended values.
* The systems hardware shall be able to communicate with the systems server.
* The system shall be able communicate over MQTT protocol.
* Systems server shall be able to send instructions the systems hardware.
* The System shall provide the user with a visual output.
* The System shall be able to communicate over a local network.

## Non-Functional Requirements

### Performance

The system shall display the visualized output to the user in less than 3 seconds.

### Reliability

The devices within the MQTT network shall have the fault tolerance mechanisms to restart itself

### Reliability

The system shall have an availability rate of 99.99%.

## Other Requirements

### System Environment Requirements.

* The system shall be able to work in an average room environment
* The systems sensors shall be able to work between 0⁰C - 50⁰C
* The system shall be able to operate in side of a LAN

### System Quality Factors

xx

### Computer Resource and Hardware Requirements

* + System shall run on tablet pc and mobile phone.

### Interfaces

xx

## Constraints

* All updates to the codebase will be done using Git in order not to lose data and manage versions more efficiently.
* In order to maintain a persisting code style and standards will be followed.
* Only analog signal sensors and card drivers can be used.
* Hardware specified in the Proposal Plan must be used.

# Qualification Provision

## Demonstration

* Using the dashboard, the tester should open and close the lights and motor(curtains)

## Test

* Tester should verify that the program is using MQTT with a network analyser like Wireshark
* Tester should verify that the system can work in an LAN.

## Inspection

* Tester should verify that the dashboard is seen.
* Tester should verify that the dashboard visualization is working.
* Tester should verify that the dashboard buttons can be pressed.

# System Models

## Component

* xx

## Requirements

* xx

## Use Case

* xx

## Activity

* xx

# Requirement Traceability

Xx

# References