CS CAPSTONE REQUIREMENTS DOCUMENT

NOVEMBER 2, 2017

SMART HOME INTERCOM SYSTEM

PREPARED FOR

OREGON STATE EECS

D. KEVIN MCGRATH

PREPARED BY

GROUP 25

TEAM 25

LAZAR SHARIPOFF

JORDAN DAVIS

GLEN ANDERSON

Abstract

The requirements for the Smart Home Intercom system are reviewed and explained in this document. This will include a detailed description of the final system, constraints associated with creating the system, and requirements that it must meet. The document also includes a Gantt chart that shows a timeline for the project, including requirements.

CONTENTS

LIST OF TABLES

1 INTRODUCTION

1.1 Purpose

This document is intended to specify requirements for the Smart Home Intercom project and related software. It will explain the functions and constraints of the finished product and provide a reference for progress on the project for the client and project assignees. This document will be updated as progress is made on the project to reflect more specific requirements.

1.2 Scope

The Smart Home Intercom system uses Raspberry Pi 3 devices as nodes for a method of communicating within a home between rooms. The system should run out-of-box and require little configuration by the end user. Users can place calls to specific rooms from one node to another, selecting between audio and video communication. Each node will run software required to talk to every other, and have the option to send a call to all nodes at once.

1.3 Definitions, acronyms, and abbreviations

TABLE 1

Term	Description
Node	A single unit of the Smart Home Intercom System
User	An individual using the system
UI	User interface for each node
Client	Individual requesting the product
Project assignees	Individuals responsible for completing the requirements of the project
rPi3	Raspberry Pi 3

Definitions

1.4 References

[1] IEEE Software Engineering Standards Committee, IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications, October 20, 1998. Source: http://www.math.uaa.alaska.edu/ afkjm/cs401/IEEE830.pdf
[2] Software Requirements Specification, 2010.
Source: http://www.cse.chalmers.se/f̃eldt/courses/reqeng/examples/srs_example_2010_group2.pdf

1.5 Overview

This document describes the intended function of the finished product and associated requirements. Section 2 describes in detail the finished product, constraints, and assumptions associated with the project. Section 3 discusses specific requirements and explains why they are necessary for the scope of this project.

2 OVERALL DESCRIPTION

2.1 Product perspective

This product will be independent in the sense that it is not a part of a larger existing system, although future projects may incorporate this one. The system will consist of nodes that communicate over a network. Each node will contain

a Raspberry Pi, a camera, as well as I/O for audio. The system will need to incorporate streaming, network, and encryption libraries in order to function.

All sensitive data transmitted between nodes, including video and audio data, must be encrypted so that even if it is intercepted, it is unusable.

Since memory is limited to what one node can hold, the libraries used for network and encryption will have to be considered in terms of overhead as well as functionality. The quality of video streaming will also be a consideration, and will be affected by the method of network connection used.

2.2 Product functions

A user will be able to place a call from one node to any other node in the house. The user can stream audio with the option of streaming video as well. A touch screen interface will allow the user to interact with the system and view video feed from other rooms.

The system will be able to detect when a person is in another room, and keep track of which room this is. If a call is placed from another room in the house, it will inform the caller of another persons last known location so that they can choose to place the call there. Since this feature will have to distinguish a person from other moving objects such as pets, facial or shape recognition may be necessary.

2.3 User characteristics

This system should be able to be used by most people with minimal instruction and set-up. Since the main functionality only involves placing and accepting calls with simple buttons, no technical experience should be needed to use this system.

There will be three types of user for the system, which are set by the user: admin, standard, and guest. A node set to guest mode will only be able to receive calls, and will not receive video calls unless the other node authorizes it. This mode would be useful for a device placed outside, for example as a method of communicating with someone outside the front door.

In standard mode, a user can place and receive calls, including video calls (if accepted from the other node). This mode would be intended for rooms that guests in a house would be likely to frequent, such as a kitchen or living room. This is also the mode that new nodes would default to.

Admin mode allows a user to view video feed from any room and broadcast a message to the entire house (every node). This provides additional functionality for the system to double as a monitoring device for other rooms, for example as a baby monitor. Since this mode would allow complete access to video feeds and tracked locations of other people, a pre-set PIN would be required to ensure that the user is authorized.

2.4 Constraints

The main hardware for this system should be built using a Raspberry Pi 3 for its affordability and processing power required for this project. The system should have its own mesh network separate from the public internet, and allow a new device node to be easily added into the system by most users. Upon starting the device the UI application for the Intercom system

A possible constraint might be that the mesh network might not be able to reliably stream video to a device multiple nodes away, because the more nodes the information has to hop over the more it degrades and the video becomes more stuttered and choppy at the receiving end of the video.

2.5 Assumptions and dependencies

Beginning this project, it is assumed that the provided Raspberry Pi 3s will have the Linux-based operating system Raspbian installed. Furthermore, it is assumed that there is sufficient hardware resources to run a network from the nodes themselves rather than relying on a central device to do so. More generally, it is assumed that the Raspberry Pi 3s will be able to handle the overhead of the required software libraries for this project. The application that will eventually run on this hardware relies on these assumptions, and will have to be altered if any of them are incorrect.

2.6 Apportioning of requirements

Ideally, the Home Intercom System would incorporate additional features to make the product more compelling to a greater range of users. For example, the ability to display calendar appointments and weather information by connecting securely to a device with internet connection. Another future requirement could enable the system to detect pets apart from people, allowing the user to view video feeds of rooms with pets in them.

3 SPECIFIC REQUIREMENTS

3.1 External interface requirements

3.1.1 User interfaces

The user interface will run on a 7 touch screen and will allow a user to operate the features of the system. Commands to the system will be sent by the user pressing buttons on the screen.

3.1.2 Hardware interfaces

Each node will require a camera, microphone, speaker, touch screen, power supply, a (built in) network card, and potentially other sensors. The software will interface with this hardware using USB and other I/O ports on each node.

3.1.3 Software interfaces

The application for this system will run on the Raspbian operating system. Since this OS is based on Linux, compatible software systems necessary for encryption, video streaming, and networking should be able to run in the application.

3.1.4 Communications interfaces

This system will rely on a mesh network to connect nodes to each other, and will not use any outside network. Data transmitted between nodes will be encrypted to ensure the privacy of users data.

3.2 System Features

3.2.1 Ability to stream video and audio between nodes

3.2.1.1 Introduction/Purpose of feature

Streaming video and audio between nodes is the core of this project. It will allow users to communicate between rooms in a house.

3.2.1.2 Stimulus/Response sequence

User prompts system to stream video/audio to another node.

3.2.1.3 Associated functional requirements

3.2.1.3.1 Functioning mesh network

3.2.1.3.2 Method of encrypting video and audio

3.2.1.3.3 Hardware configured for appropriate I/O

3.2.2 Functional user interface

3.2.2.1 Introduction/Purpose of feature

An interface will allow users to select which rooms to place calls to, choose between audio and video, and operate the other features of the system.

3.2.2.2 Stimulus/Response sequence

User selects room to place call to, specifications for the call, and the node makes the call accordingly.

3.2.2.3 Associated functional requirements

3.2.2.3.1 Application that can run on Raspbian

3.2.2.3.2 Back-end for interface

3.2.3 Ability to detect people in a room

3.2.3.1 Introduction/Purpose of feature

Recognition of a person in the room will allow a node to automatically receive a call upon detection.

3.2.3.2 Stimulus/Response sequence

Node recognizes person in the same room, logs their presence so that another user can place a call to that location.

3.2.3.3 Associated functional requirements

3.2.3.3.1 Identify a person from another moving object

3.2.3.3.1.1 Facial recognition

3.2.3.3.1.2 Sensors (temperature, infrared)

3.2.3.3.1.3 Image analysis

3.2.3.3.2 Store last known location of a person

3.2.4 Minimal first-time setup

3.2.4.1 Introduction/Purpose of feature

Minimizing the difficulty of first-time setup will allow the system to be more usable. It will also prevent setup related errors that could be caused by a user incorrectly following directions.

3.2.4.2 Stimulus/Response sequence

System is powered on, and proceeds to automatically set up as much as possible without user interference.

3.2.4.3 Associated functional requirements

3.2.4.3.1 Startup/power-on script to initialize application

3.2.4.3.2 Application auto-launches, locks in

3.2.5 Installation

3.2.5.1 Introduction/Purpose of feature

Each node will need to be wall mountable so that it is easily accessible. Each node will be powered through a wall socket as it is a convenient source that doesnt require more complex installation.

3.2.5.2 Stimulus/Response sequence

Not applicable

3.2.5.3 Associated functional requirements

3.2.5.3.1 Wall mountable

3.2.5.3.2 Powered through wall socket

3.3 Performance requirements

1. Ability to broadcast video and audio between nodes with minimal delay and latency (at most 10ms).
2. Video at a minimum of 20 fps (verify against hardware.)
3. System responds to user input within 1 second

3.4 Design constraints

All software for this project will need to run on a small computer on each node. The software must run as an application on Raspbian (similar to Linux). Each node must be self-contained, that is the same wall-mounted object will hold the camera, microphone, speaker, touch screen, and any additional sensors.

3.5 Software system attributes

3.5.1 Reliability and availability

After release, the system should be able to run with minimal setup and be able to perform necessary functions with minimal bugs. However, since this is a first prototype, the system does not need to be perfectly reliable.

3.5.2 Security

The following security requirements must be followed to protect users personal communications and data:

1. Video and audio cannot be stored, only streamed
2. Data transmitted between nodes must be encrypted so that if intercepted it could not be used
3. The system cannot connect to an external network
4. Admin/standard/guest modes described above must be enforced

3.5.3 Portability

Since each node will contain the software necessary to communicate with another, they should be completely independent systems. The system should be expandable simply by adding additional nodes with the same software.

4 GANTT CHART