SANTA CLARA UNIVERSITY
Computer Engineering Department
Software Engineering

Doorbell for the Hearing Impaired

by

Dominic Magdaluyo
Shannen Edwin

Santa Clara, California
November 16, 2018

Table of Contents

List of Figures

List of Tables

Doorbell
for the
Hearing Impaired

Dominic Magdaluyo
Shannen Edwin

Computer Engineering Department
Santa Clara University
November 16, 2018

Abstract

Doorbell solutions for hearing impaired individuals is seriously limited. Affordable solutions are not effective and higher end solutions are not affordable for most hearing impaired individuals and are not designed primarily with them in mind. In this report, we propose our design for a cheaper doorbell solution that is just as effective as the expensive solutions.

Introduction

Doorbells and knocking are often useless for those that are audibly impaired. Someone with hearing impairments needs to depend on other residents to answer the door for them or purchase expensive equipment to send them phone notifications. Depending on the neighborhood, the ability to know there's a visitor at the door is essential for quality of life including package delivery and community involvement.

Current solutions are limited in supporting these individuals. Louder doorbells are useless to those that are completely deaf and can be annoying for housemates that are not hearing impaired. Current light systems are not integrated into the house and must be carried around the house (or multiple installments are needed, thereby increasing the cost). Furthermore, the light systems may be easily missed if not facing the device or if in a well-lit room. Video doorbells are expensive and require a smartphone to interface with it. The notifications may be missed during the time frame where a person is at the door. A useful alternative is smart wearables but those are limited in battery life. For example, the Apple Watch Series 4 is advertised to last 18 hours on average use which is not useful for a person when sleeping. Designing a low-cost solution was important to the team, as unemployment among deaf individuals is high at 47%.[1] Generally, hard of hearing individuals are senior citizens who do not have much money. A simple and affordable solution would improve their lives greatly. The hearing impaired community already struggle enough and providing an affordable option to doorbells is essential in keeping them aware of their surroundings and being part of their communities.

To tackle these issues, our solution would aim at integrating a system into the entire household by using Philips Hue light fixtures or, as a cheaper alternative, providing LED strips, removing the need for a portable device that may be easily forgotten. Furthermore, the system will give the user options on a configuration that works best for their home. On top of that we will provide a low power vibration bracelet

[1] Garberoglio, C.L., Cawthon, S., & Bond, M. (2016). "Deaf People and Employment in the United States: 2016." Washington, DC: U.S. Department of Education, Office of Special Education Programs, National Deaf Center on Postsecondary Outcomes.

solution. This would be sleek so that it can be worn while sleeping while providing a gentle but noticeable vibration. The bracelet could be charged while away from the house, so a user will always be able to use it once in the house. Together, our system will be low cost while providing more functionality towards the hearing impaired.

Requirements

Functional:

- Critical
    - Doorbell will activate light system
- Recommended
    - Doorbell will activate vibration system
    - Doorbell is integratable with other light/vibration devices available
    - The System will be low power
- Suggested
    - Doorbell will not communicate over WiFi

Non-Functional:

- Critical
    - The system will be cheap, ideally $50 or less for a full house integration
- Recommended
    -
- Suggested
    - The system has an appealing aesthetic

Constraints

- Not building from the ground up

Use Cases

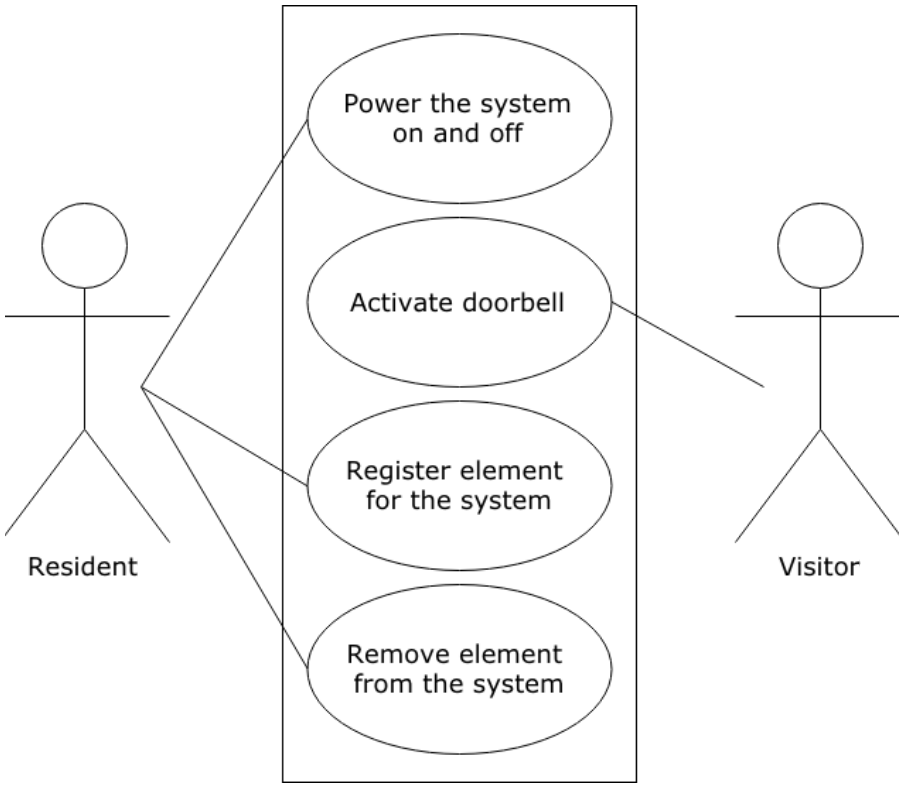

Figure 1: Use Cases

Use Case Descriptions

- Power the System on and off
    - **Actors**: Resident
    - **Goal:** Setting up system and personalizing it
    - **Postcondition:** System should safely shut off or turn on with previous settings
- Activate Doorbell
    - **Actors:** Visitor
    - **Goal:** Alert the Resident, that someone is at the door
    - **Precondition:** System is successfully activated with a device paired to it
    - **Postcondition:** Alert the Resident through their connected device(s)

    - **Exceptions:** Nothing happens if System is not active or device is disconnected
- Register Element for the System
    - **Actor:** Resident
    - **Goal:** Connect a device so Resident may be alerted of Visitor at door
    - **Precondition:** System is powered on successfully
    - **Postcondition:** Device is registered successfully to system and will be activated once prompted
    - **Exception:** Unsuccessful device registration → prompt user of error in pairing
- Remove element from the System
    - **Actor:** Resident
    - **Goal:** Unregister device from the system when no longer using it
    - **Precondition:** System should be on
    - **Postcondition:** Device successfully unregistered

Activity Diagrams

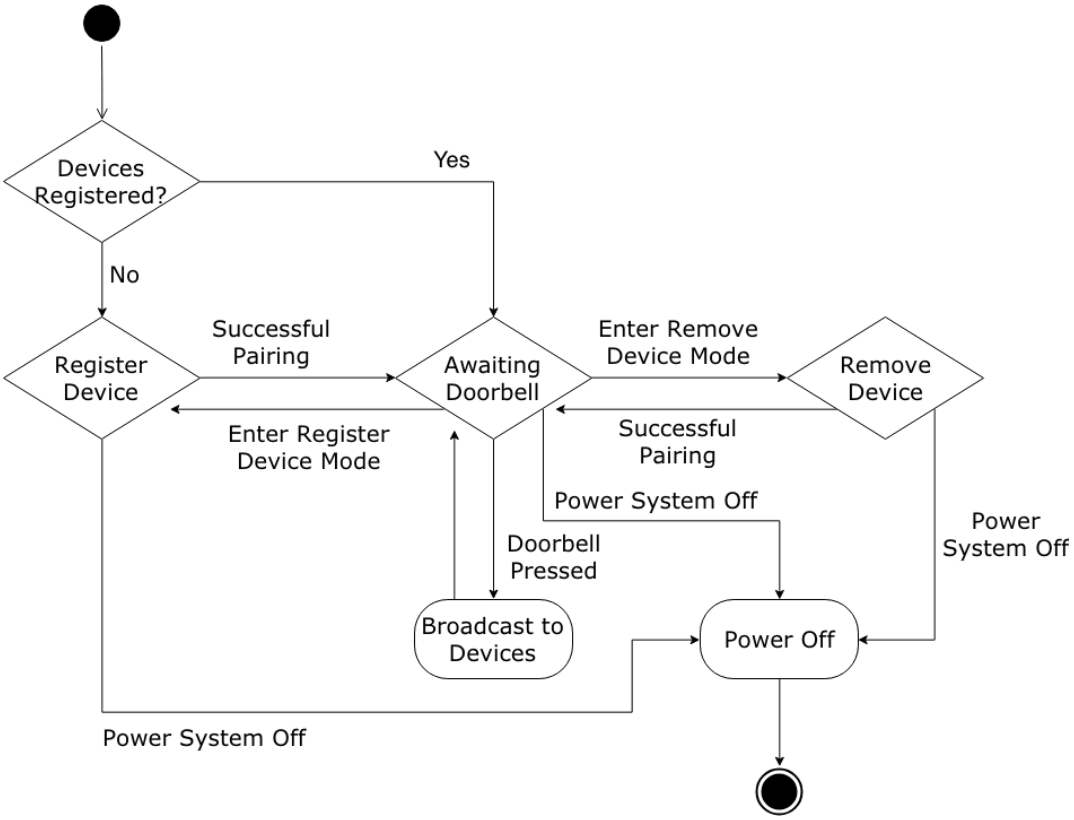

Figure 2: Resident Activity Diagram

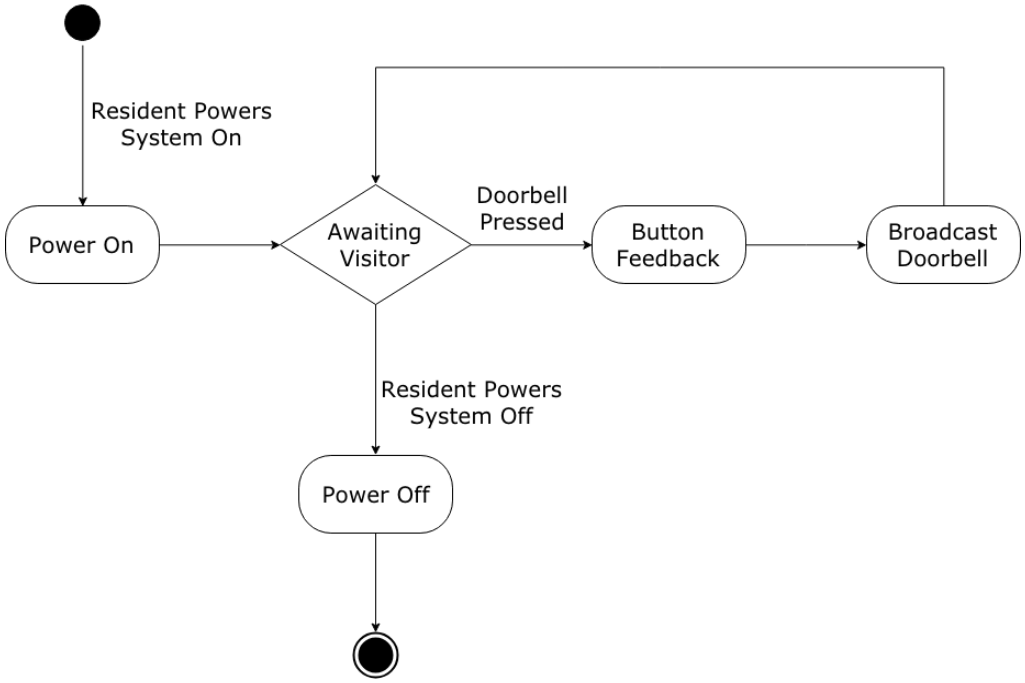

Figure 3: Visitor Activity Diagram

Technologies Used

- Raspberry Pi Zero W
- Philips Hue Lights
- LED Strips with RF capabilities
- Bone conduction ear buds
- FitBit
- Bluetooth
- WiFi

Design Rationale:

The main computer of this System will be the Raspberry Pi Zero W. The Raspberry Pi Zero W is a low cost IoT board with WiFi capabilities and is part of a widely used open source project. This will allow us to find solutions to various problems we may encounter in development. Furthermore, if we come up with unique issues, the community is very responsive and hopefully minimize any delays.

We want this to be an open platform that multiple types of devices can be connected to. As such we will provide high cost and low cost solutions. Philips Hue Light bulbs are popular in IoT solutions for lighting in houses. Furthermore, they have an open API which will help development be smoother. The LED strips are our answer to a low cost light solution. LEDs are cheap and could be spread out across a house with minimal extra costs.

Furthermore, we will provide vibration solutions to the doorbell system. We will use FitBit as our high end solution which has an open API making development easy. We will be developing with a higher cost FitBit so that we can interact with a screen. For a low-cost vibration solution, we will be using bone conduction ear buds which are designed to hang around an individual's head at all times and can be simply interacted with by using bluetooth.

Overall, we will be experimenting with both WiFi and Bluetooth. We would like to possibly stay away from WiFi in case a low income household didn't have a router. Both are standard forms of communication between devices and are available with our chosen devices. We will also be able to set up the Raspberry Pi Zero W with hostapd (host access point daemon) to make the board function like a router. Most likely, this will lead to more power consumption so it is not an ideal solution.

Conceptual Model

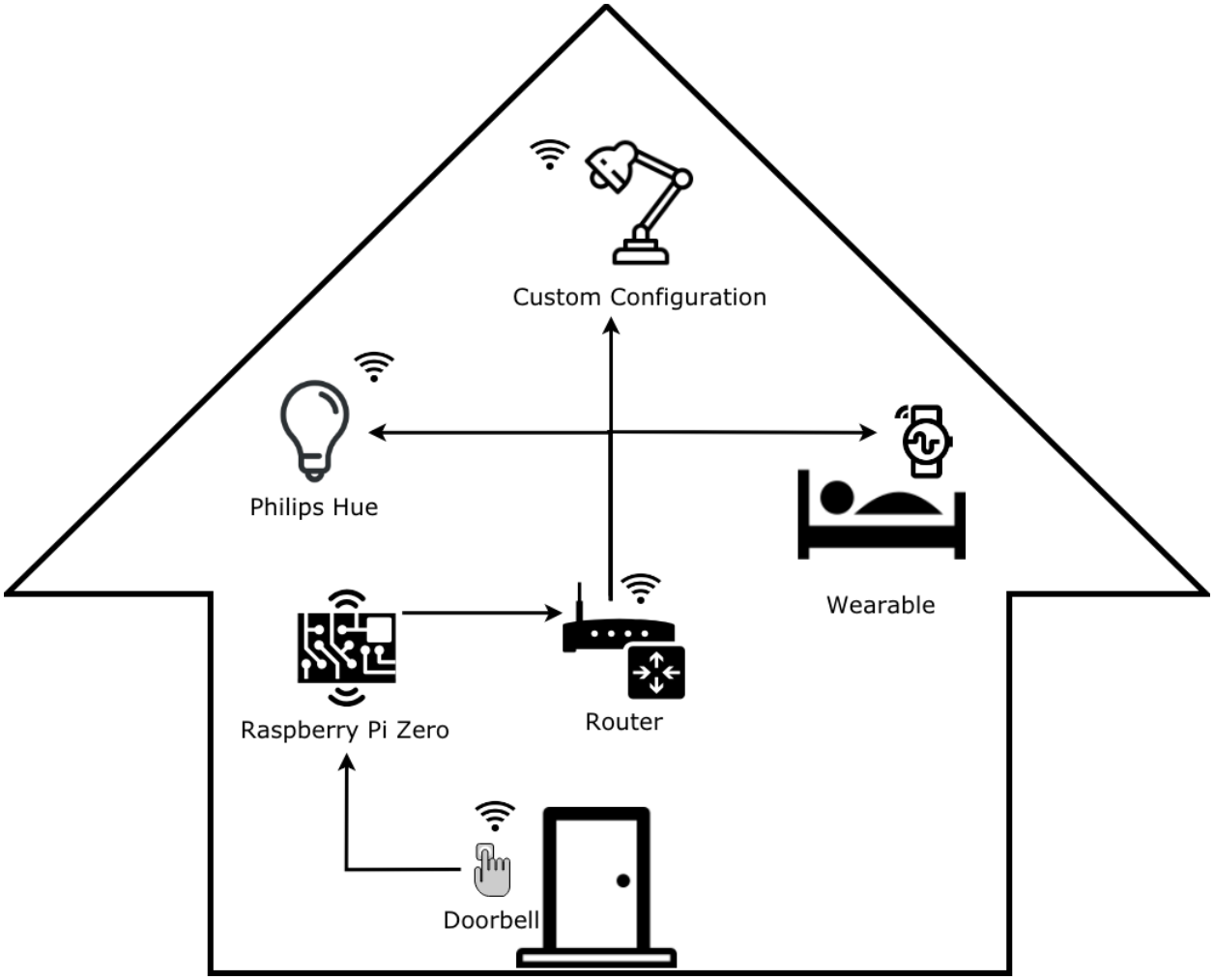

Figure 4: Doorbell Conceptual Model

The system will use the Raspberry Pi Zero as the main computer. A router will be necessary in communicating to other devices in the household. The router can be performed by the Raspberry Pi Zero if necessary. The button of the doorbell will be attached to the Raspberry Pi Zero.

Architectural Diagram

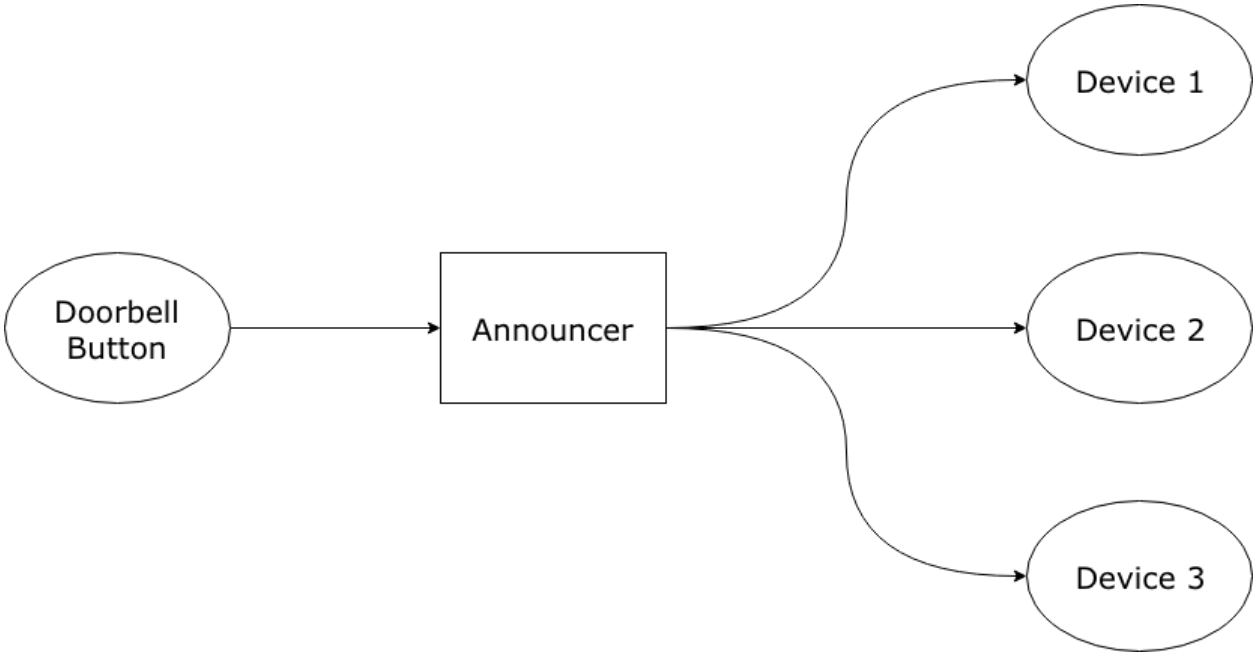

Figure 5: Doorbell Architecture

The system will use an event-based architecture. In Figure 4, the doorbell button triggers an event that goes to the Raspberry Pi Zero which acts as an announcer in this architecture. This event tells the announcer to broadcast the appropriate doorbell event to any device currently registered in the system.

Test Plan

We will spend most of our time testing the system in small pieces. We will have to make sure that each type of device can be registered with the system and function. Our tests will mainly consist of us pressing a button and looking for a registered device to activate. We will set up scripts that will give us messages that show us each step the system is taking. These tests would have to be continuously done from the start of development.

Major testing performed before deadlines would consist of us running through each use case. We will limit testing to just one device and once all of those are successful, move on to testing multiple devices at one time.

Risk Analysis

Risk Name	Consequences	Probability	Severity (0-10)	Impact	Mitigation Strategies
Time	• Behind on deadlines	0.7	9	6.3	1. Follow the development tImeline
Compatibility Issues	• Will delay the development of the system • Spend unnecessary money	0.6	7	4.2	1. Research technologies and their compatibility in full 2. Discuss with advisor in approaches to resolving issues
Network Issues	• Will delay the development of the system	0.35	9	3.15	1. Work with the IT department to resolve issues 2. Might just have to focus on bluetooth

Table 1: Risk Analysis

Development Timeline

Task Name	Duration	Start	ETA	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Quarter Break
Fall Quarter														
Problem Statement	14 days	18.09.22	18.10.05											
Funds Proposal	14 days	18.10.06	18.10.19											
Design Report (Rough Draft)	28 days	18.10.20	18.11.16											
Design Report (Final)	5 days	18.11.26	18.11.30											
Minor Development (Base System)	50 days	18.11.17	19.01.06											
Winter Quarter														
Initial Operating System	67 days	19.01.07	19.03.15											
Design Review	18 days	19.01.07	19.01.25											
Design Report (Revised)	7 days	19.01.26	19.02.01											
Testing	21 days	19.02.23	19.03.15											
Spring Quarter														
Recommended Requirements	28 days	19.04.01	19.04.28											
Demo System	11 days	19.04.29	19.05.09											
Testing	18 days	19.04.15	19.05.02											
Design Presentation	18 days	19.04.22	19.05.09											
Project Report	28 days	19.05.10	19.06.07											
Final Implementation	28 days	19.05.10	19.06.07											

Table 2: Development Timeline