

Holiday Arboreal Light Project

Rajiv Bhoopala Michael Scholl Justin Falat Robert Tyynismaa Aaron Hudson

Advisor/Client: Dr. Thomas Daniels

Introduction

Problem Statement:

Many people decorate their homes and other objects like Christmas trees with a set of lights. However, current holiday lights are limited by not being customizable and are usually only one design. In order to decorate an arboreal display we must first visualize the display and lay the lights accordingly. Thus, our team has decided to tackle this problem of being able to create complicated light displays in a simple manner.

Solution:

Our solution to this problem combines both hardware and software. The idea is for users to set up RGB LED lights on a tree and then upload patterns to the string of lights. We will use an Android application to send pictures of the RGB LED lights to the web server, from which a model of the LEDs will be created. The web server will send the pattern to the Raspberry Pi which will then power the lights. The user will be able to select many different types of colors/patterns.

Requirements and Constraints

Functional Requirements:

- RPi PWM controller sends RGB values to LED
- Android App takes pictures for calibration process and sends them to the Web App
- Web App selects what patterns to display on the tree
- Web App controls state of LED Manager by creating and deleting .lck files
- LED Manager drives lights

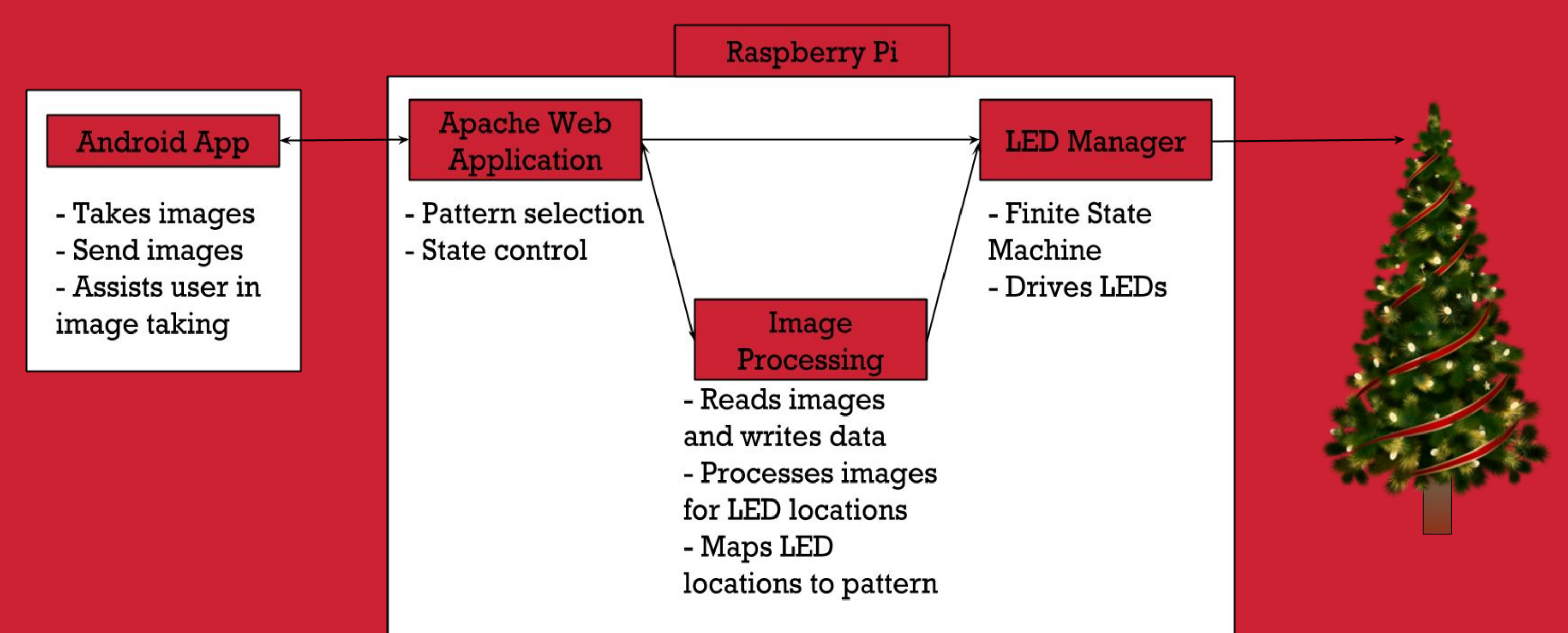
Non-Functional Requirements:

- Android App must be responsive and easy to use
- Calibration process must be energy efficient to limit battery use
- Web App must control state of LED Manager while maintaining mutual exclusion
- System must be able to run for long periods of time without fail

Constraints

- Raspberry Pi 3B processor and storage limitations
- Android only mobile application
- Android App and Web App must communicate via WiFi

Block Diagram



Technology

Hardware:

- Raspberry Pi 3 Model B
- 3x WS2811 LED String of 100 LEDs
- 3.3V to 5V Level Shifter (SN74AHCT125N)
- 12V30A Power Supply

Software:

- Java
- Android Studio
- Python
- RPi WS281x Python Library
- PHP
- Apache2 Web Server

Intended Users & Uses

The intended users for our holiday lights are people who are interested in programmable LED lights, as well as anyone interested in using technology around their homes.

The intended use for our holiday lights is for display on a tree displayed in a home and/or indoor environment. It is not suitable for outdoor display.

Testing Methodology

Testing Environment

- Different background lighting settings
- Changing background objects
 - Windows, Mirrors, Picture frames

Testing Strategy

- Image quality on different devices
- Android application unit testing
- Multiple iterations of image calibration
- LED update frequency

Testing Metrics

- Image processing computation time
- LED power draw over level shifter
- LED/PI Temperature over time
- LED brightness -> Image blur

SubSystem Descriptions

Mobile App

- Start calibration process with Server
- Image capture
- Upload images to the Web Server

RPi, Level Shifter, Lights

- Drive WS2811 LEDs via GPIO PWM
- 3.3V output to 5V input logic

Image Processing

- Analyzes images for LED locations
- Saves LED locations for pattern mapping

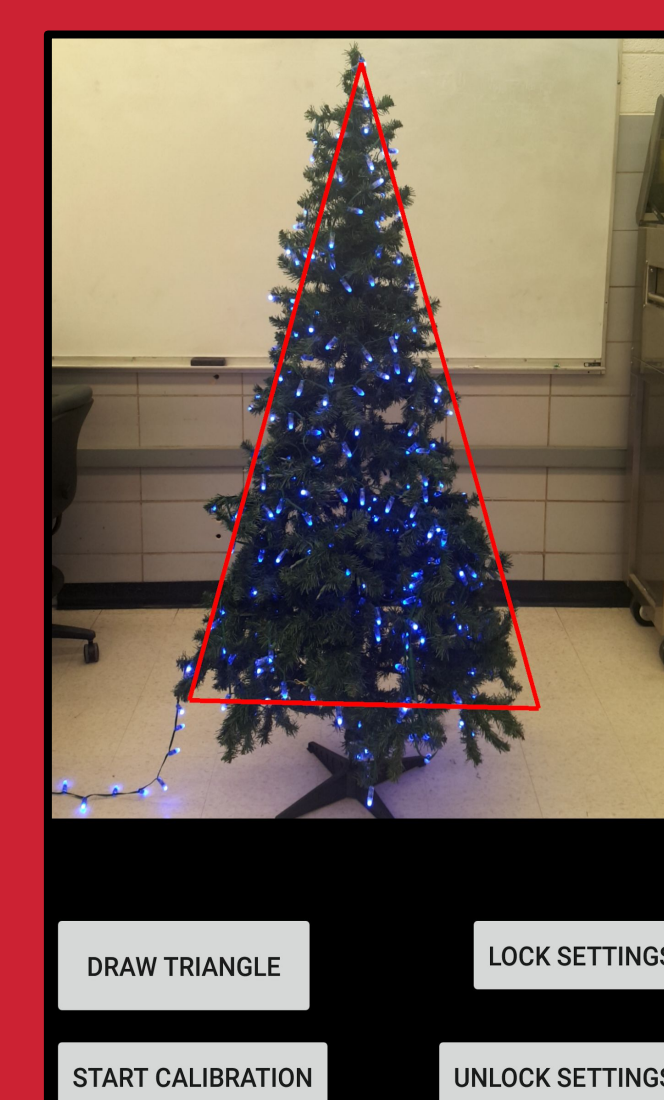
LED Manager

- Drives WS2811 LED patterns
- Polls for state .lck files

Web App

- Calibrates LED positions
- Sets color of LEDs/patterns

SubSystem Diagrams



Mobile App (1)

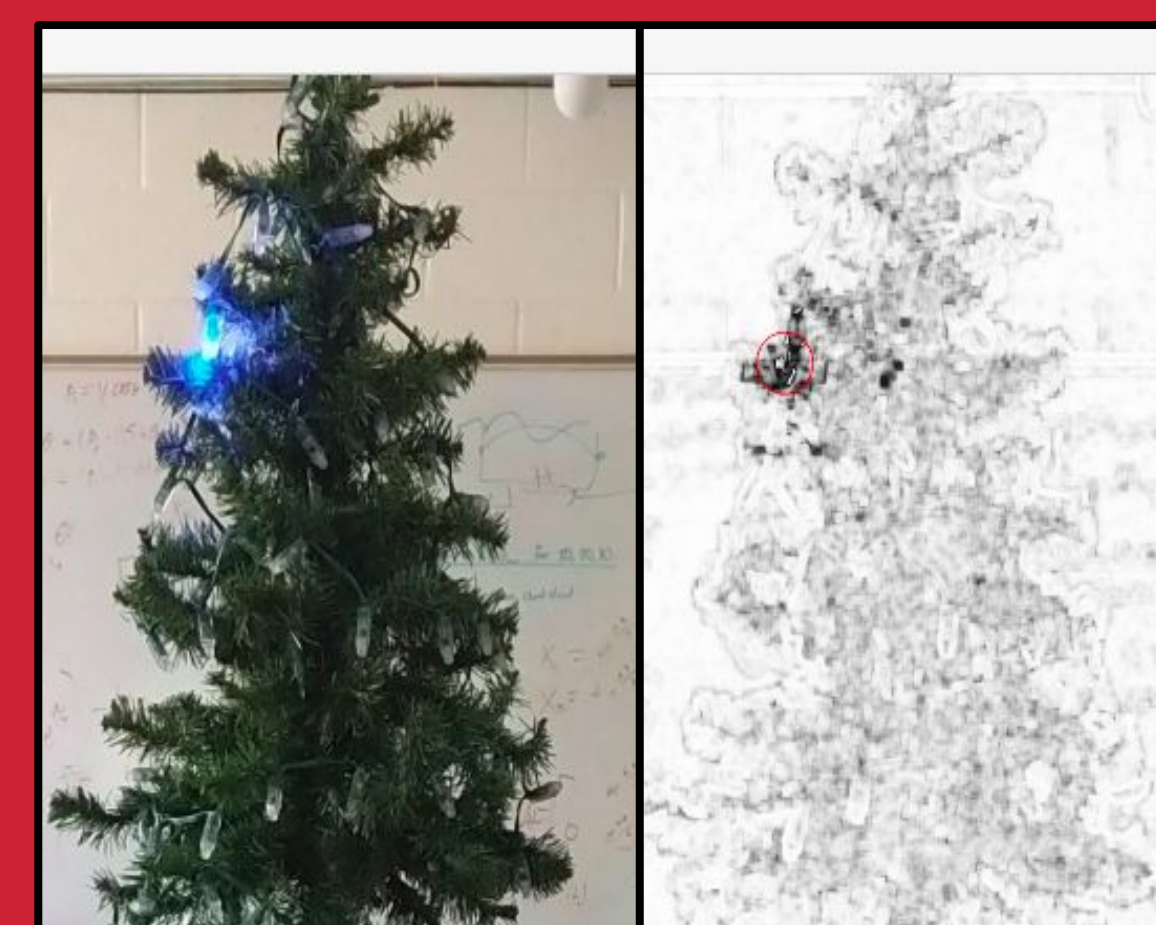
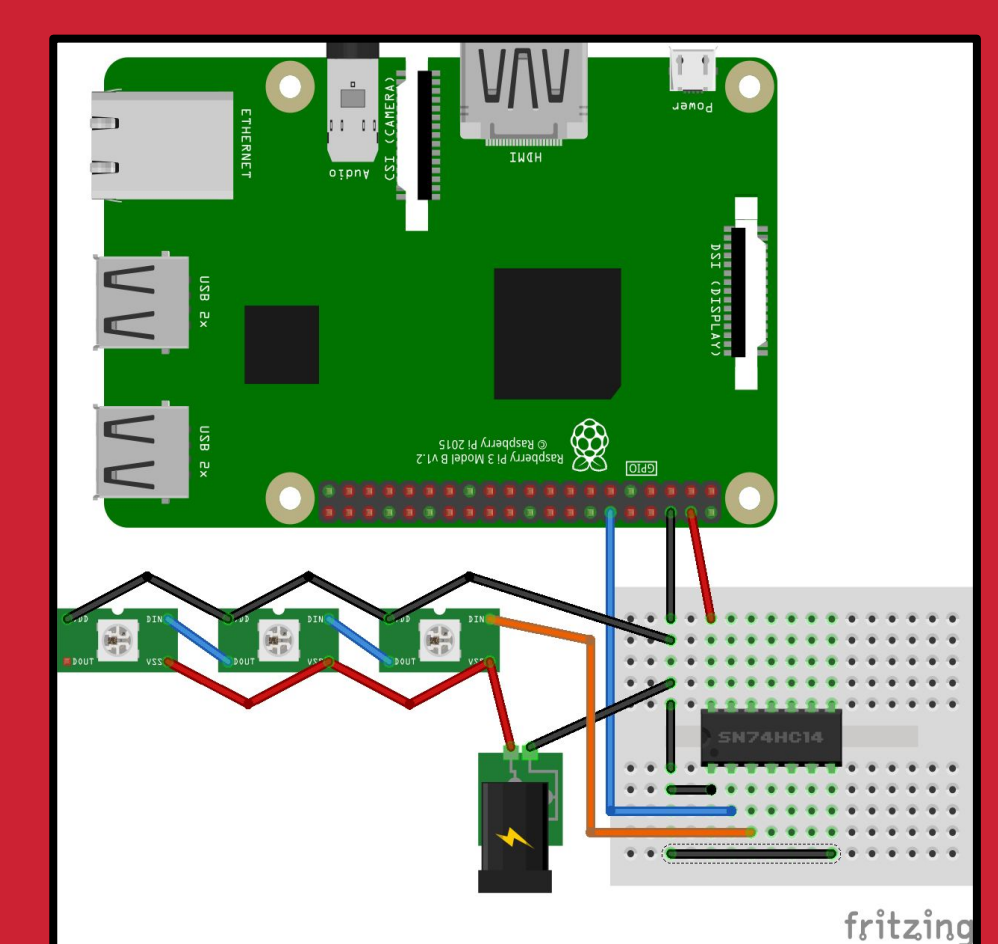
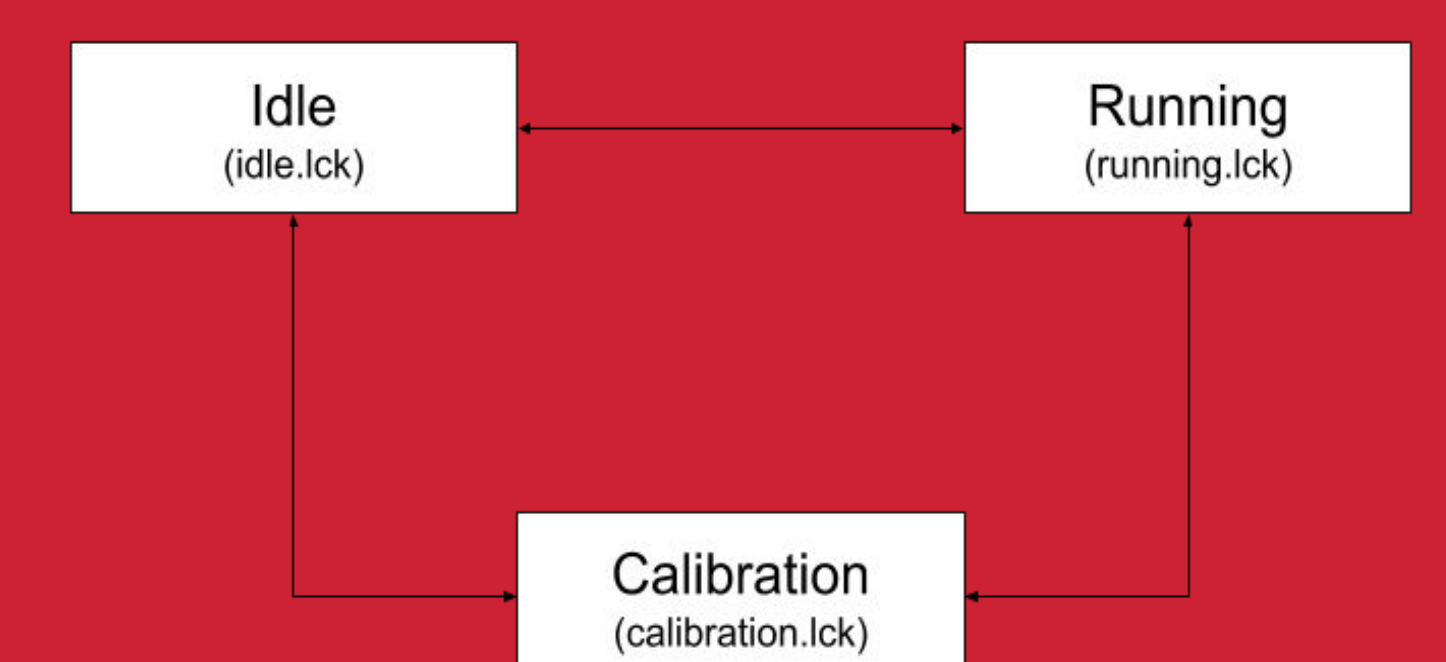


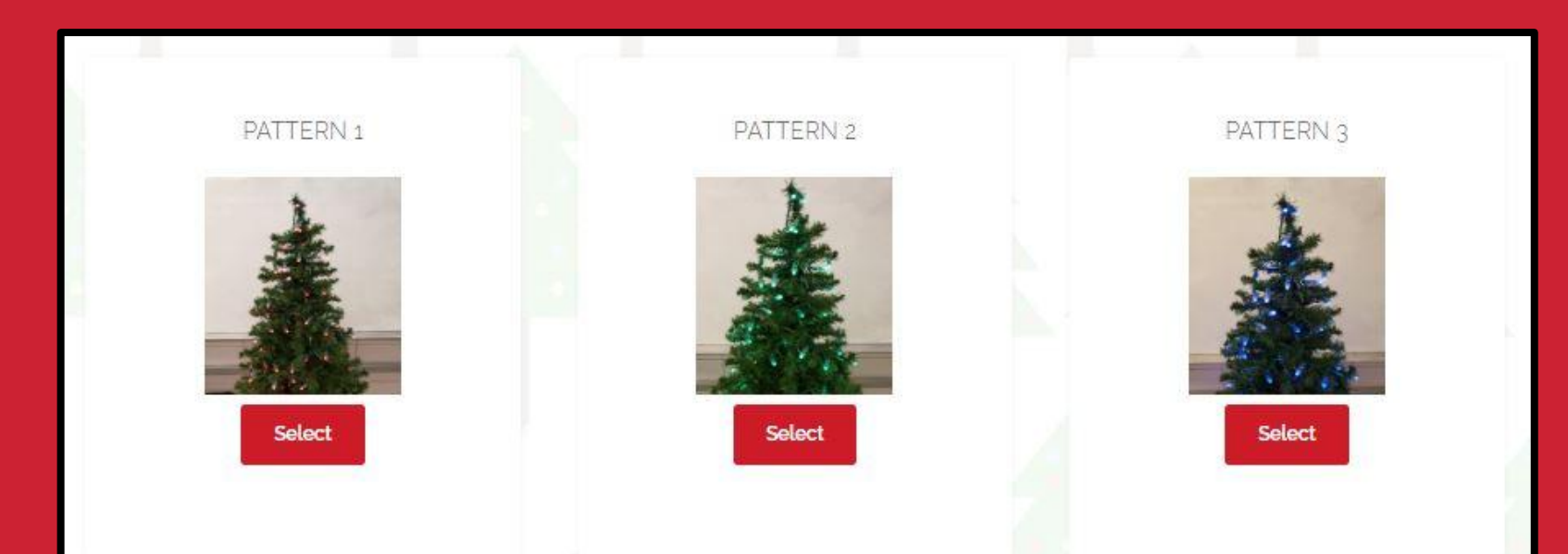
Image Processing (3)



Raspberry Pi, Level Shifter, & Lights (2)



LED Manager State Diagram (4)



Web App (5)

Standards

- 1789-2015 - IEEE Recommended Practices for Modulating Current in High-Brightness LEDs for Mitigating Health Risks to Viewers
- WiFi 802.11b/g/n - Wireless network bearer operating in the 2.4 and 5 GHz ISM bands with data rates up to 600 Mbps.