2017

IoT Project Proposal

fit5140 advanced mobile systems

IoT pROJECT PART A.

QIUXIAN CAI 27010767

YINGCHEN LIU 26981068

Table of Contents

[1 Project Description 2](#_Toc494544690)

[2 The problem solved 2](#_Toc494544691)

[3 System Architecture 3](#_Toc494544692)

[4 Functionality of the IOS client application 4](#_Toc494544693)

[5 User Interface Design mockups 4](#_Toc494544694)

[6 Circuit diagrams 6](#_Toc494544695)

# 1 Project Description

Lazy Mails is an IoT system aimed at intelligently differentiating variety of mails in a physical mailbox. The system distinguishes useful letters from junk mails and send notifications to the owner’s mobile phone, which saves the users’ time in checking mails everyday and avoid them from missing important letters.

# 2 The problem solved

Currently, people are getting more junk mails than ever. This leads to the increased difficulty in collecting important letters from a stack of junk mails. People need to check their mailbox frequently in order to make sure that they do not miss any important letter. The system can solve this problem via intelligently categorizing different kinds of mails. It can monitor incoming mails and automatically separate them into different categories. The only thing people need to do is to wait notifications on their mobile phones and decide whether to pick up them from their mailboxes.

Another issue is, for tenants, they often receive letters that belong to previous renters. It is very annoyed but they have to check whether the letters belong to them or not. The system also can address this problem by filtering letters based on the receiver’s name and address.

When a postman sends a letter to a Lazy Mails box, the system will process the letter, put it into a specific category on the app and decide whether to notify the user or not. The system can distinguish different categories of mails, such as normal letters, parcel collection cards, bank statements, utility bills, and even advertisements.

# 3 System Architecture

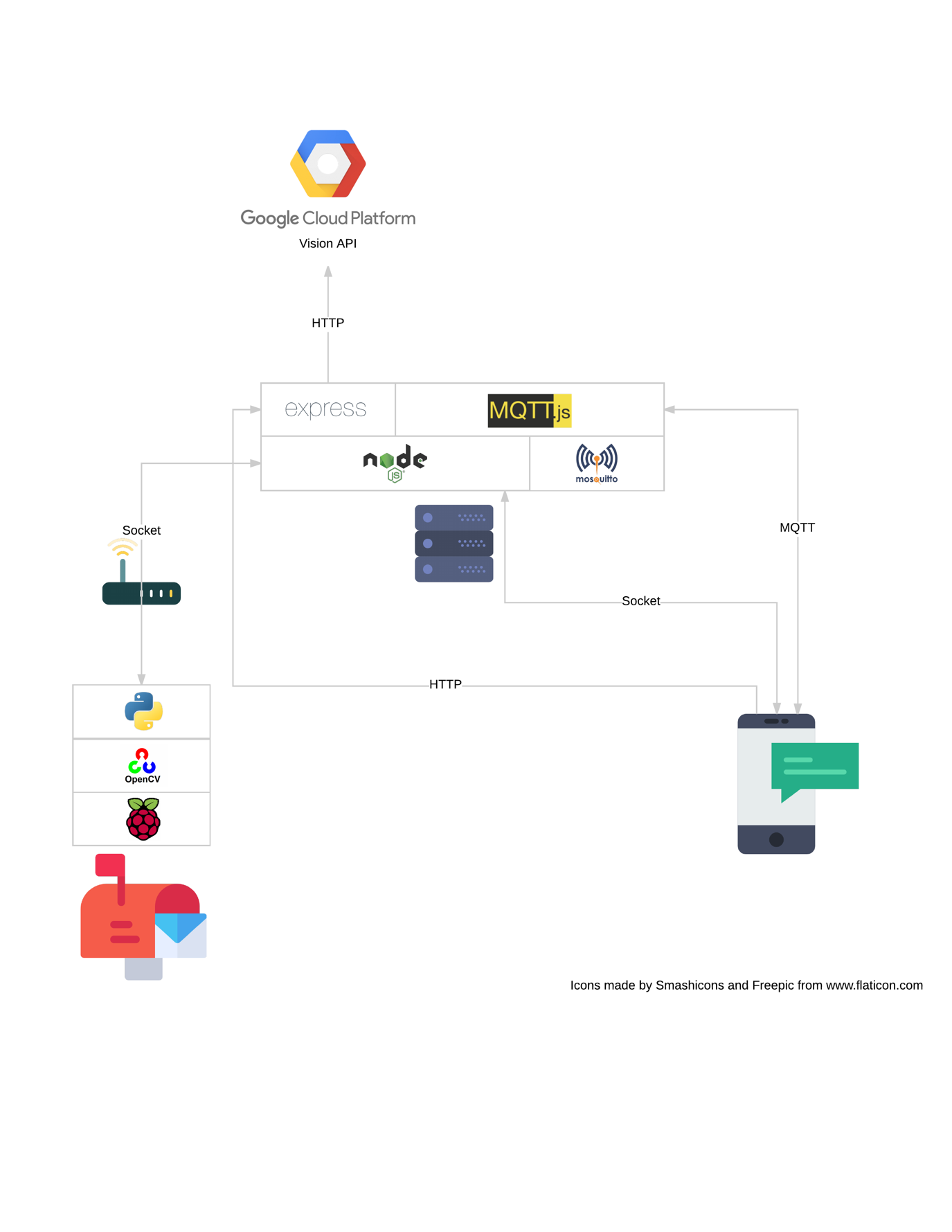


Figure 3.1. System Architecture

A Raspberry Pi is installed in the physical mailbox with a camera and other sensors (see 6 Circuit Diagram) using Python and OpenCV to detect a mail is arrived. The Raspberry Pi sends a message containing the photo of the received mail to the server via socket connection. The server will send a request to Google Vision API via HTTP protocol to analyse the photo. Then, some post-processes will be done before a notification can be send to the mobile phone via MQTT. ExpressJS is used to handle user setting changes from the mobile phone and requests for history mails.

# 4 Functionality of the iOS client application

The iOS client application for Lazy Mails has a bunch of functionalities which are very handy to use.

Users can check their mails by categories on mobile phones without going outside to check their physical mailboxes.

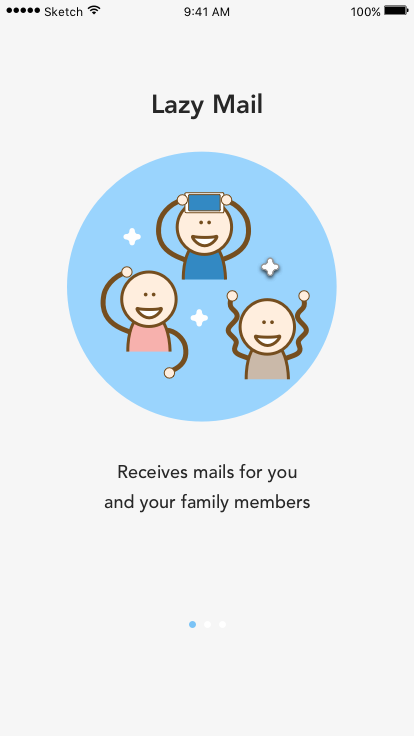
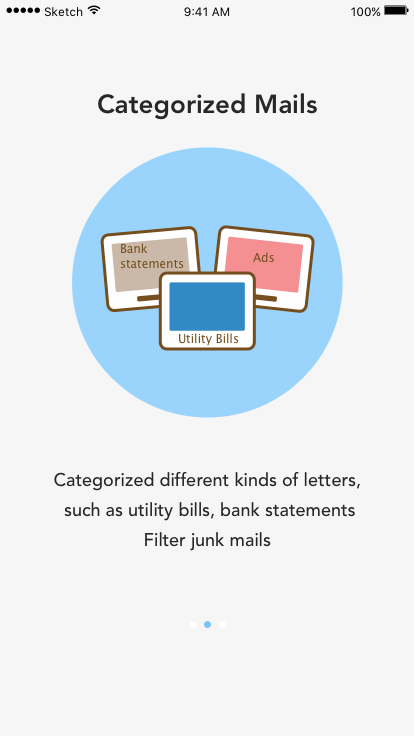
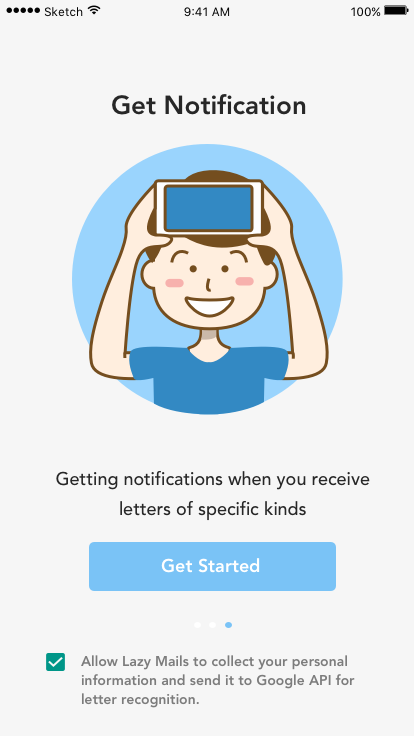
By clicking a mail, they can also view appearance photo and other information, such as sender’s address, receiver’s name, received time, extracted URLs and etc., of the received mail and know whether or not it is important to them. In the mail detail page, they can report issues to correct the information and help developers to improve the app. Issues include incorrect category, photo incorrect display and incorrect recognition. For incorrect category, user can correct the category and the letter can be moved to the right category; for photo incorrect display, user can view the photo which shows the entire view of the physical mailbox; for incorrect recognition, the whole information extracted by the system will be displayed and user can also report that issue to the server.

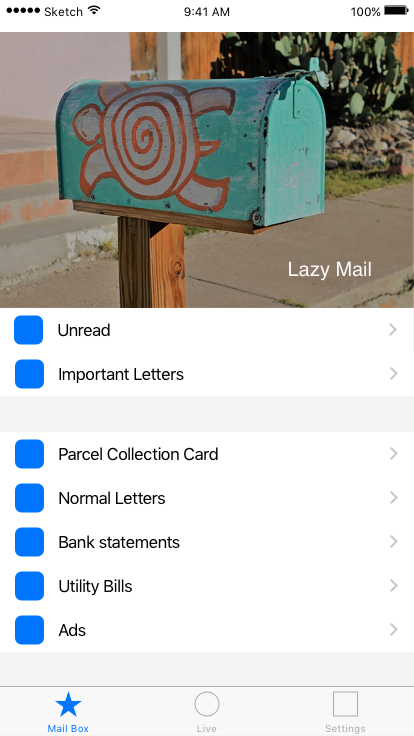
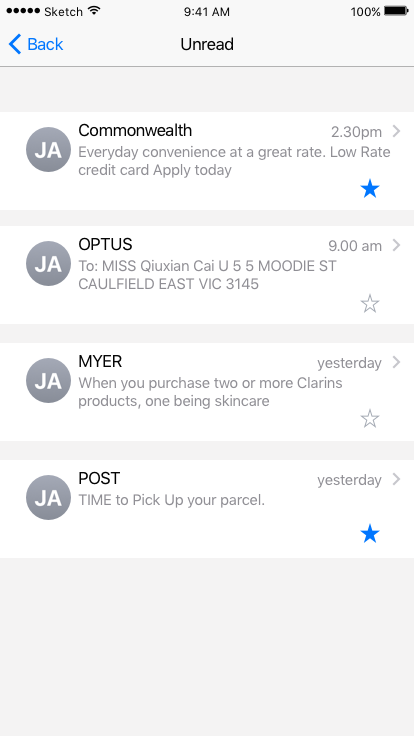
Similar to the Mail app on iOS, users can also mark a letter as important reminding them to handle it later.

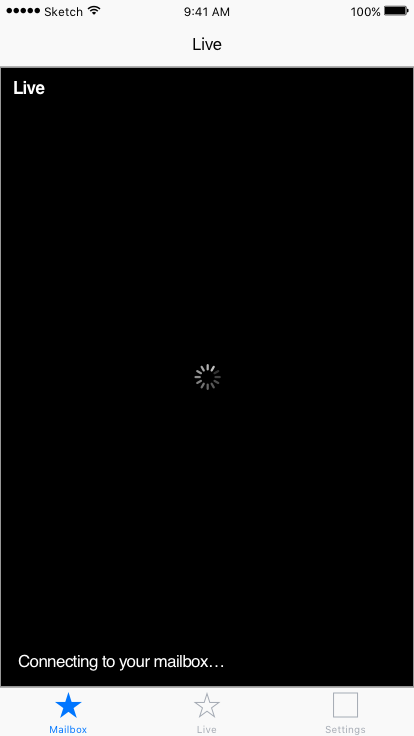
In settings, users can set their names and addresses so that the system can filter out letters that are not belong to them. Users can set whether or not to receive notifications for incoming letters in each category. In addition, energy saving mode enables the system turning off the light and the camera in the mailbox during the night in order to save energy.

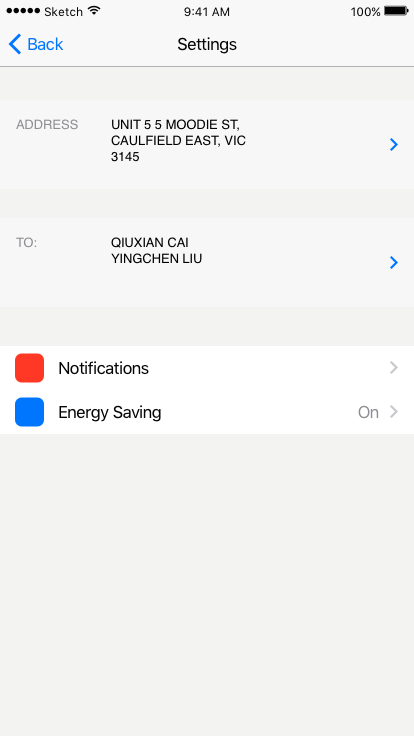
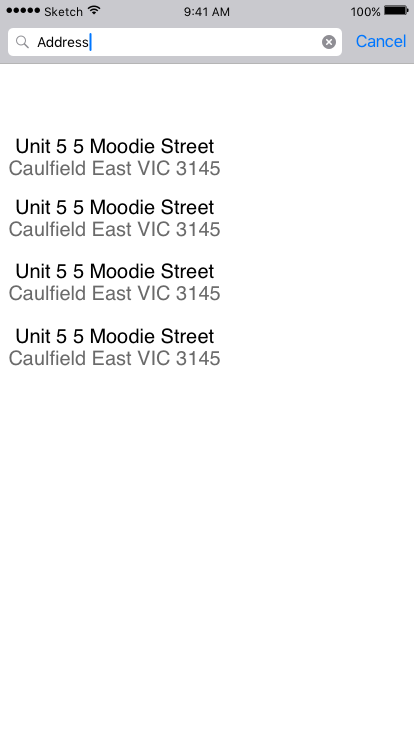
Live video of the mailbox is also supported in case anything goes wrong and they want to have a view of their letterbox at the real time.

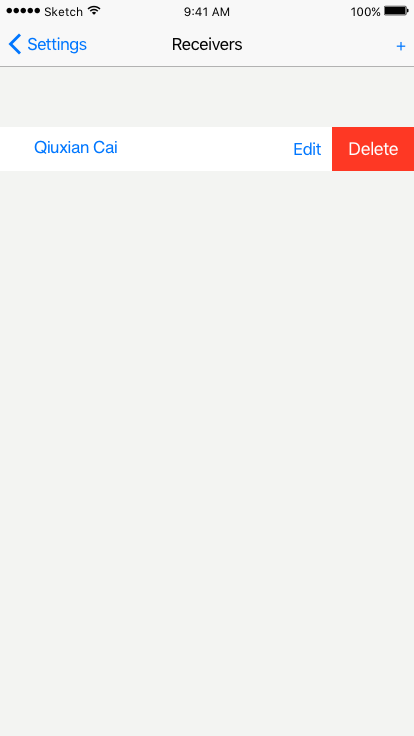
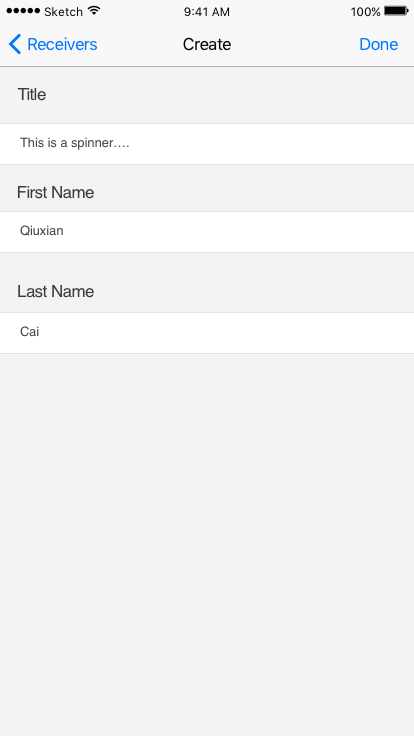
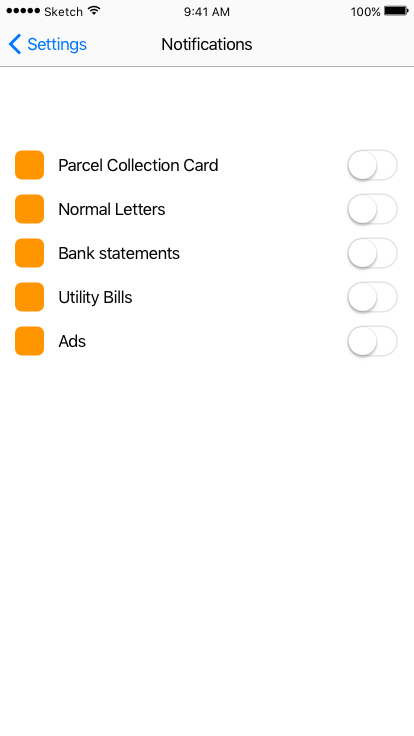
# 5 User Interface Design mockups

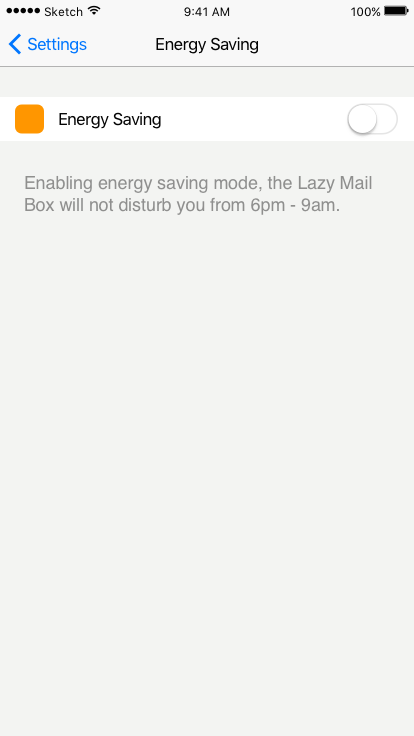
  



# 6 Circuit diagrams

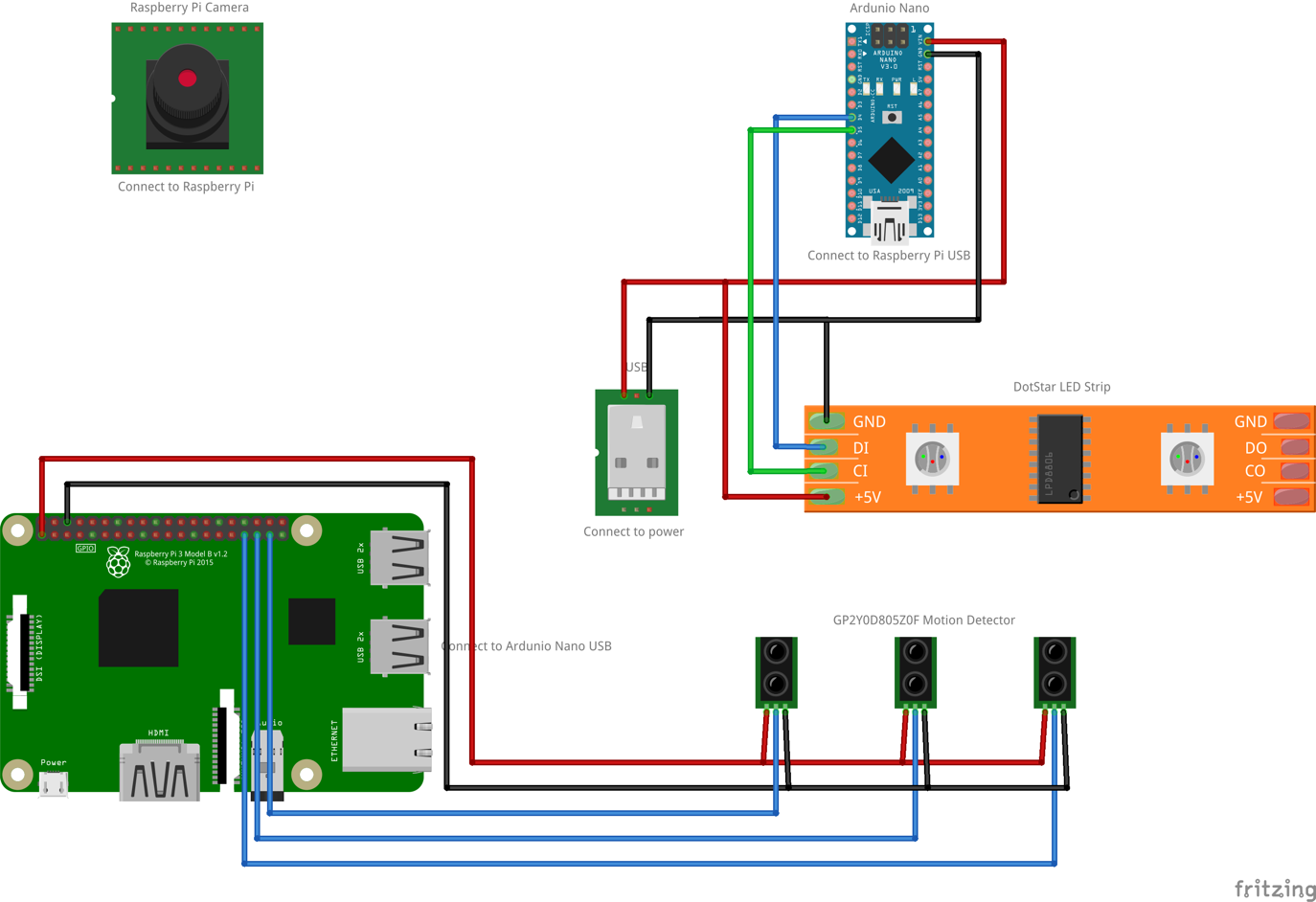


Figure 6.1 Circuit Diagrams

In the Lazy Mails box, three motion detectors are used to detect a mail is inserted into the Lazy Mails box. The camera is used for taking photo of a mail and capturing live video of the mailbox. To ensure that the camera works properly, an Arduino Nano is installed to control a LED strip (5V-based). A separate USB cable is used for power the LED strip as it requires high current. All the three motion detectors, Raspberry Pi camera and Arduino Nano connect to and controlled by a Raspberry Pi.