2017

IoT Project Document

fit5140 advanced mobile systems

IoT pROJECT PART B.

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 for checking mails everyday and avoid them from missing important letters.

# 2 System Architecture

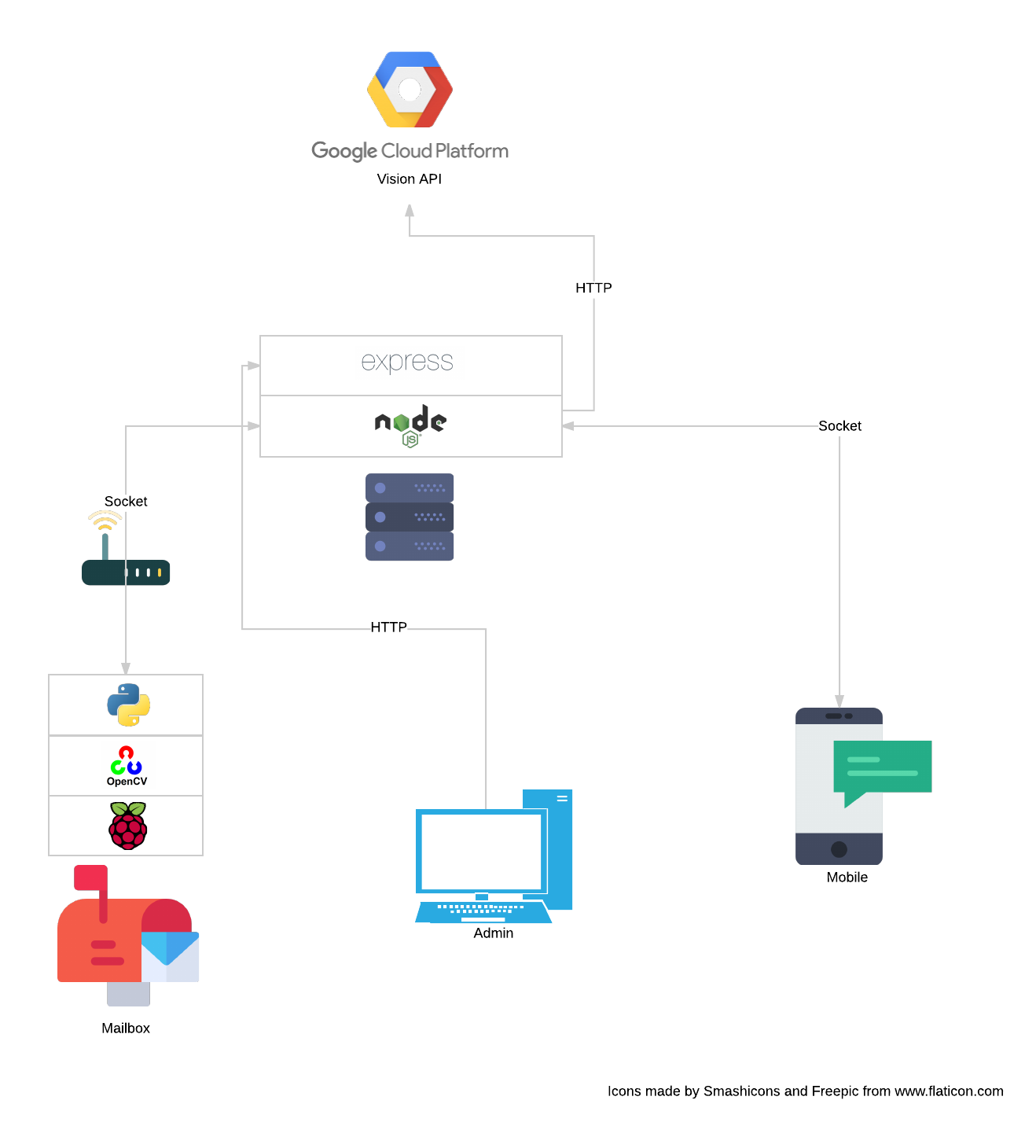


Figure 1. System Architecture

A Raspberry Pi is installed in the physical mailbox with a camera and motion detectors (see 3 Circuit Diagrams). We use three motion detectors because it can increase the success rate in detecting an arriving mail. The camera is used to capture the photo of the mailbox. By using background subtraction in OpenCV with Python, mail can be cropped. The Raspberry Pi sends a message containing the photo of the received mail to the server via TCP/IP protocol. The server then sends a request to Google Vision API via HTTP protocol for logo, text detection and labelling. After that, some post-processes, such as image rotating and mail classification, will be done before a notification can be sent to the mobile phone also via TCP/IP. By using multi-threading, this system can support multiple mailboxes. Also, for one mailbox, it supports multiple users. In addition, a web-based admin system built with ExpressJS enables administrators to manage the system easily using a web browser.

# 3 Circuit Diagrams

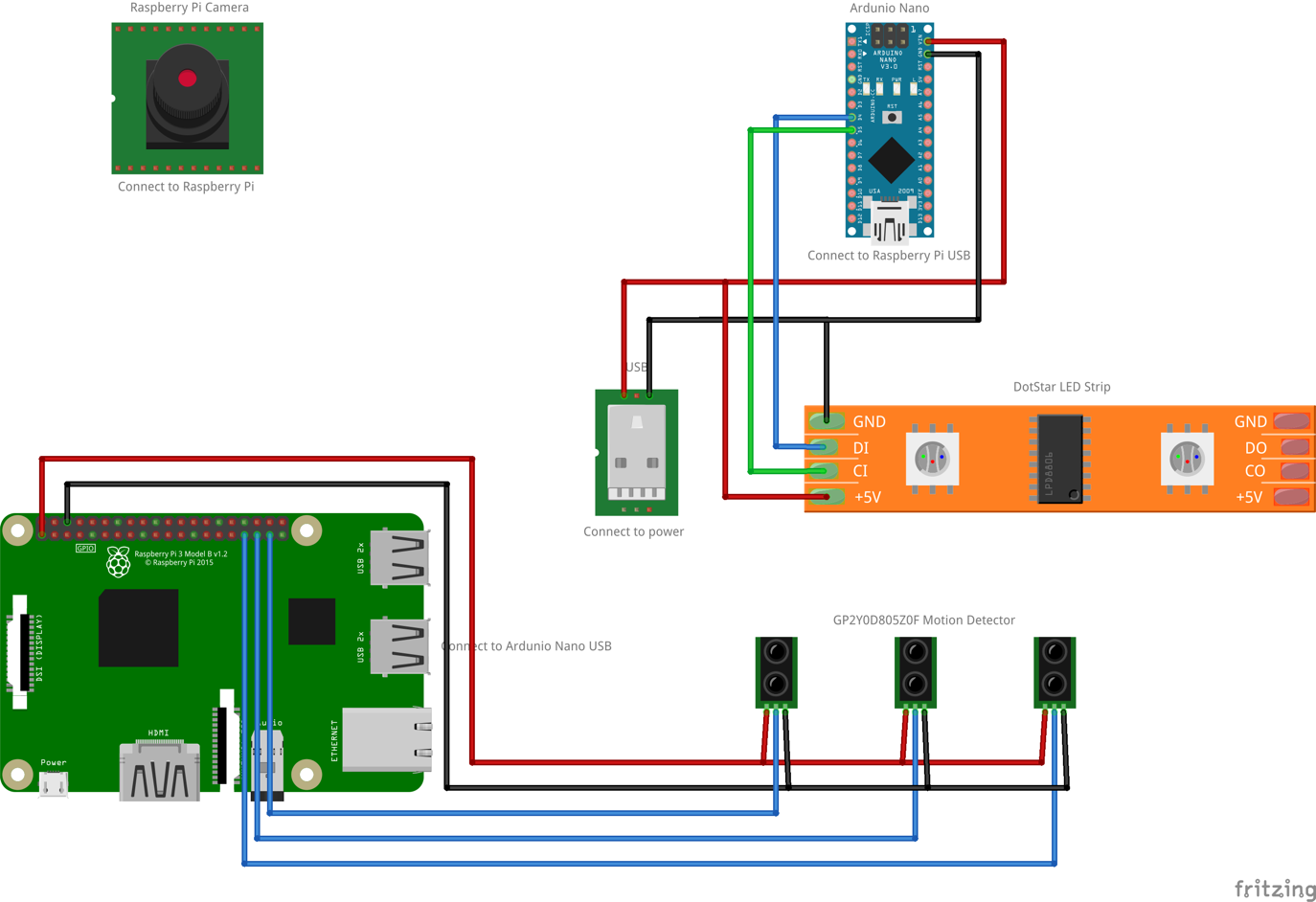


Figure 2. Circuit Diagrams

In a LazyMails box, to ensure that the camera works properly, an Arduino Nano is also installed to control a LED strip (5V-based). The Raspberry Pi uses serial to control the Arduino, so that the LED can be turned on or off. Another USB cable is used for power the LED strip as it requires high current.

# 4 Libraries/APIs Used

* iOS Application
  + QRCodeReader

As this project supports multiple users for any single mailbox, user register is required. QRCodeReader makes it easier for users in a family to link their accounts to a mailbox by scanning the QRCode provided by us. It is extremely hard to develop a QRCode scanner by ourselves, so we decided to use this library.

* + SwiftValidator

It is convenient to use this library to do validation as it has some useful and good built-in validators such as email and length.

* + Whisper
  + Toast-Swift
* Server
  + ExpressJS