# Project Plan

# <Team : HoyKhom (KMITL)>

Member : Isara Naranirattisai

Peerawat Pipattanakulchai

Siridej Phanathanate

Patcharapon Jantana

Thanatcha Sangpetch

# Project Objectives

พัฒนาและรวบรวมระบบเพื่อจัดการและสนับสนุนการพัฒนาผลผลิตทางการเกษตรด้วยเทคโนโลยี เพื่อติดตามผลต่างๆ เช่น การเจริญเติบโตของต้นไม้และผล สภาพอากาศ และ ความชื้น เป็นต้น เพื่อให้สามารถวางแผนแก้ปัญหา และคาดการณ์ปัญหาล่วงหน้าได้

# Project Scope



# **ภาพรวม**

Raspberry Pi จะทำงานเป็นคาบเวลา เช่น ทุกๆ 1 วัน โดยค่าดังกล่าวจะตั้งค่าในตัว Pi เอง เมื่อถึงคาบเวลาที่ต้องทำจะทำการร้องขอค่าเซนเซอร์ความชื้นจาก STM32 และ ถ่ายรูปผลไม้(ในที่นี้คือทุเรียน) และส่งต่อไปยัง server

Server จะทำการรับค่าความชื้นดิน และ รูปภาพจาก Raspberry Pi และ อัพโหลดขึ้น Cloud Database อัตโนมัติ

Matlab จะทำการวิเคราะห์รูปภาพและสภาพอากาศต่างๆ เมื่อมีการกดปุ่มที่โปรแกรมที่มีหน้า UI ที่เขียนจาก Matlab เมื่อวิเคราะห์เสร็จก็จะนำข้อมูลไปเก็บใน Cloud Database(คนละชุดกับที่ไปดึงมาวิเคราะห์)

Line bot จะทำการส่งคำสั่งที่ใช้ผู้ใช้สั่งผ่านทาง Line ไปยัง Line API และ Line API จะส่งต่อไปยัง Server เพื่อให้ Server ดึงข้อมูลการวิเคราะห์ครั้งล่าสุดไปยัง Line API แล้วส่งต่อไปยัง Line ผู้ใช้ต่อไป

**Module Details**

**STM32**

ทำการรอสัญญาณเพื่อขอข้อมูลความชื้นจาก Raspberry Pi 3 หากมีสัญญาณเข้ามาจะส่งค่าความชื้นไปให้ Raspberry Pi ผ่านทาง UART

**Input :**

1. สัญญาณเพื่อขอข้อมูลความชื้น จาก Raspberry Pi 3
2. อ่านค่าความชื่นของดินจาก Soil Moisture Sensor เป็น analog ค่าตั้งแต่ 0-4095

**Output:**

1. ค่าความชื่นของดิน มีค่า 0-100 (% ความชื้น) ส่งผ่าน UART ไปยัง Raspberry Pi 3

**Raspberry Pi 3**

จะทำงานก็ต่อเมื่อถึงเวลาครบคาบเวลาทำงาน ซึ่งค่าดังกล่าวกำหนดใน Raspberry Pi

**Input :**

1. ภาพถ่ายผลทุเรียน จาก Raspberry Pi Camera V2
2. ค่าความชื้นของดิน จาก STM32

**Output:**

1. ส่งสัญญาญขอค่าความชื้นไปยัง STM32 ผ่านทาง UART
2. ส่งรูปถ่าย และ ความชื้นของดินไปยัง Server ด้วย Restful API

**Server**

กรณีที่ 1 เมื่อครบรอบการทำงานของ Raspberry Pi 3

**Input :**

1. ค่าอุณหภูมิ ความชื้น สภาพอากาศของวันนี้ จาก Weather API ด้วย Restful
2. รูปถ่าย และ ค่าความชื้นที่ได้รับจาก Raspberry Pi 3

**Output:**

1. อัพโหลดรูปและข้อมูลขึ้นไปยัง Cloud Database

กรณีที่ 2 มีการเรียกดูผลการวิเคราะห์ผ่านทาง Line

**Input :**

1. คำสั่งจากผู้ใช้ ผ่าน Line API

**Output:**

1. Server ส่งข้อมูลการวิเคราะห์ล่าสุดไปยัง Line API

**Line bot API**

เป็นส่วนติดต่อระหว่าง Server กับ Line

**Input :**

1. รับผลการวิเคราะห์ล่าสุดจาก Server

**Output:**

1. ส่งคำสั่งจากผู้ใช้ไปยัง Server

**Matlab**

เป็นส่วนที่เอาไว้วิเคราะห์ข้อมูลต่างๆ จาก Cloud Database และส่งผลการวิเคราะห์กลับไปยัง Cloud Database โดยจะมี Input และ Output จาก matlab คร่าวๆดังนี้

**Input :**

1. รับ input จาก UI (คลิก)

**Output:**

1. ส่งผลการวิเคราะห์ไปยัง Server

โดย Matlab จะมีการวิเคราะห์ข้อมูล 2 ส่วนดังต่อไปนี้

**การวิเคราะห์รูปผลไม้**

รับรูปจาก cloud database แล้วนำมาประมวลผลผ่านโปรแกรม matlab เมื่อประมวลผลเสร็จนำรูปกลับไปเก็บใน cloud database อีกชุดนึง ( ไม่เซฟทับ ) โดย จะมี input และ output ดังนี้

**Input**

รูปผลทุเรียน ที่มีพื้นหลังสีขาว

**Output**

ขนาดของผลทุเรียน

**วิเคราะห์ข้อมูลจากข้อมูลสภาพอากาศ ความชื้น อุณหภูมิ และ อื่นๆ**

รับข้อมูลจาก cloud database แล้วมาประมวลผล เป็นกราฟต่างๆ

**Input**

สภาพอากาศ ความชื้นอากาศ ความชื้นดิน อุณหภูมิ

**Output**

กราฟของการเปลี่ยนแปลง

|  |
| --- |
|  |

|  |
| --- |
|  |

**Roles and Responsibilities**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Core** | **Responsibilities** |
| Peach | Project Member | Image Processing , UI in Matlab | * check size of durian (Analytics) * Check white balance * UI in matlab * other method for analysis |
| Nay  (Peerawat Pipattanakulchai) | Project Member  Contact Customer | Pi and STM32 | * Develop Sensor and STM32 * Implement unit test between rPi STM32 * Find and Analysis requirement from customers |
| Pao  (Siridej Phanathanate) | Project Member | Pi and STM32 | * Develop Raspberry Pi * Take pictures and POST to Restful * Communicate with STM32 * Optimizing Performance |
| Dream  (Isara Naranirattisai) | Project Manager,  Scrum Master | Server, Database | * Plan Project * Monitor Project * Line Bot Api (Server Side) * Manage Database * RESTful API * Tester server part * Manage GIT |
| Eao เอี่ยว  (Thanatcha sangphet) | Project Member | Data analysis  Image Processing | * Develop Monitoring Interface * Data analytics |
| Mr. X | Customer | Farmer | * Supply Project Requirement |

**Necessary Tools**

**Hardware**

**Hardware**: Raspberry Pi 3, STM32 Nucleo , Moisture Sensor

**Software**

**Database**: PostgreSQL (DBMS)

**Image Processing and Statistic**: Matlab(GUI)

**Programming Language** (RPi and Bot): Python 2.7

**Bot**: Line Bot

**Server**: Deploy on heroku (PaaS)

**Collaboration**

**Working Methodology**: Scrums (Trello)

**Version Control (collaboration)**: GIT

**Garden**

**Fruit**: Durian

**4M:**

**Man**:

Stm32 and PI: มีคนที่เคยมีประสบการณ์กับกล้องและส่งคนที่มีทักษะ STM32 และ arm ที่สุดในทีมไปด้านนี้

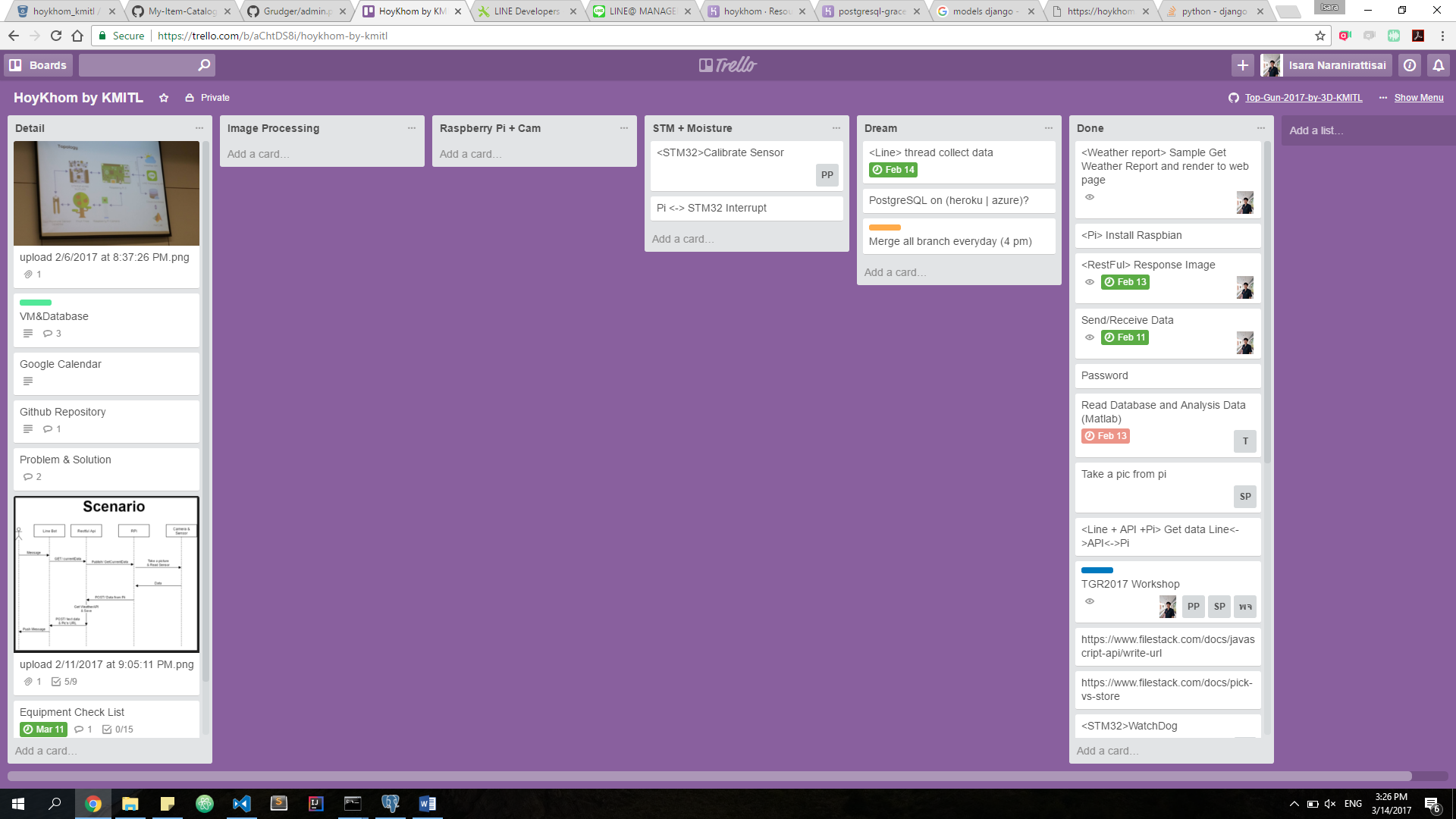
Bot and API: มีทักษะ python และ flask

Image Processing: เนื่องจากไม่เคยทำด้านนี้มาก่อนจึงส่งคนเรียนรู้ไวและมีทักษะโปรแกรมมิ่งมากๆทำด้านนี้

**Machine**:

Raspberry Pi, Heroku Cloud, STM32

**Method**:



* Scrums (Trello Board)
* GIT(Bitbucket)

**Manage**:

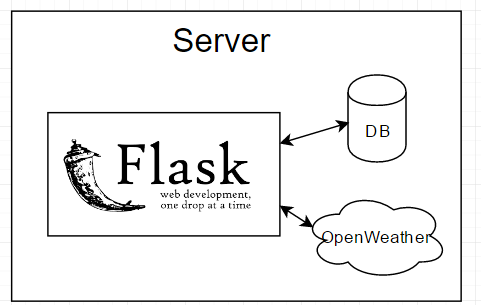
Project Manage(Trello (Scrums)), Google Docs(meeting), Code collaboration (Git), Google Calendar

**Work Breakdown Structure and Schedule**

Gantt Chart(Work Breakdown Structure):(AN: Afternoon, NI: Night)

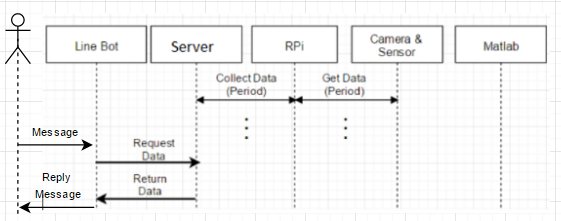
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **WBS** | | 02/1 - 03/12 | 03/13 | | 03/14 | | 03/15 | | 03/166 | | 03/17 | |
| **Tasks** | PIC  (name) |  | AN | NI | AN | NI | AN | NI | AN | NI | AN | Ni |
| **Pi & STM** | N, PA |  |  |  |  |  |  |  |  |  |  |  |
| - Learning | N, PA |  |  |  |  |  |  |  |  |  |  |  |
| -Development | N, PA |  |  |  |  |  |  |  |  |  |  |  |
| -Pi&STM Integration | N, PA |  |  |  |  |  |  |  |  |  |  |  |
| -Testing | N, PA |  |  |  |  |  |  |  |  |  |  |  |
| **SERVER** | D |  |  |  |  |  |  |  |  |  |  |  |
| Learning Linebot |  |  |  |  |  |  |  |  |  |  |  |  |
| - Developing Line Bot | D |  |  |  |  |  |  |  |  |  |  |  |
| -Testing | D |  |  |  |  |  |  |  |  |  |  |  |
| -Deploy to server | D |  |  |  |  |  |  |  |  |  |  |  |
| -Design&Create DB | D |  |  |  |  |  |  |  |  |  |  |  |
| -RESTful API | D |  |  |  |  |  |  |  |  |  |  |  |
| **Matlab** |  |  |  |  |  |  |  |  |  |  |  |  |
| - develop check size durian & Testing | P |  |  |  |  |  |  |  |  |  |  |  |
| - GUI in matlab & Testing | P |  |  |  |  |  |  |  |  |  |  |  |
| -check count and desease durian | P |  |  |  |  |  |  |  |  |  |  |  |
| -integration matlab program and fix bug | P |  |  |  |  |  |  |  |  |  |  |  |
| - kmean clustering | E |  |  |  |  |  |  |  |  |  |  |  |
| -import and export database | EAO |  |  |  |  |  |  |  |  |  |  |  |
| -Statictical visualization | EAO |  |  |  |  |  |  |  |  |  |  |  |
| -Hypothesis Testing  -box plot |  |  |  |  |  |  |  |  |  |  |  |  |
| **INTEGRATE** | \* |  |  |  |  |  |  |  |  |  |  |  |
| Integrate | \* |  |  |  |  |  |  |  |  |  |  |  |
| Testing | \* |  |  |  |  |  |  |  |  |  |  |  |
| **MONITORING** | D |  |  |  |  |  |  |  |  |  |  |  |
| -Daily Meeting | D |  |  | \_ |  | \_ |  | \_ |  | \_ |  | \_ |
| -Team planing | \* |  |  |  |  |  |  |  |  |  |  |  |

System Scenario



\* Server จะเป็นตัวจัดการข้อมูลต่างๆ เช่น Database และ รับข้อมูลจาก Openweather Api, รวมถึงติดต่อกับอุปกรณ์ต่างๆ

\*\* Line Bot ตัวที่ทำหน้าที่ Callback มายัง Server เมื่อมี message มาจาก user



Request Data

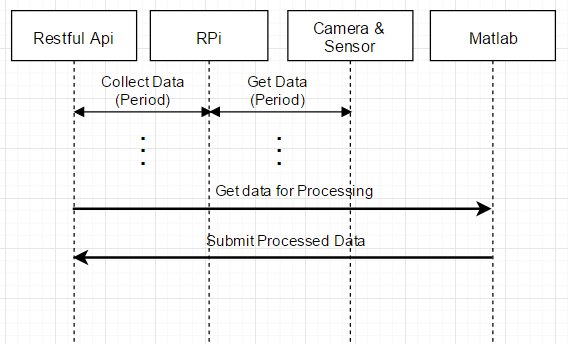
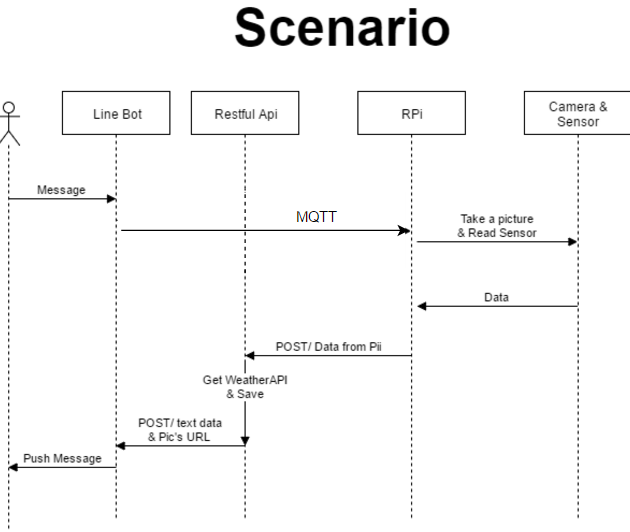


Image Processing



Real-Time (Optional)

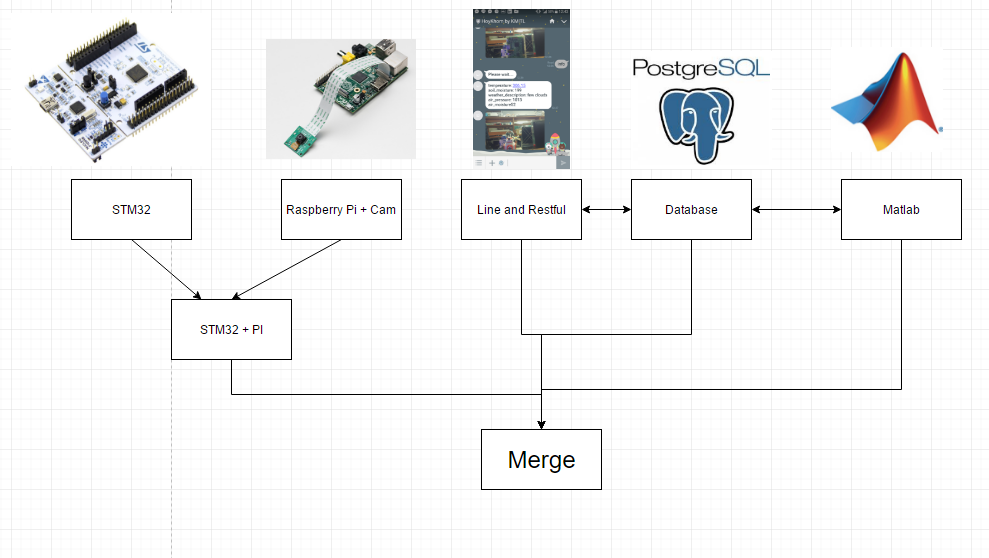
System Integration Strategy

แบ่งส่วนต่างๆเป็น Modules ได้แก่ STM32, Raspberry Pi3, Matlab, Server, Database และนำมาเชื่อมต่อกันดังรูป โดยจะมีการ develop และ test (Unit Test) ในแต่ละ modules จากนั้นจะเชื่อมต่อกันทำการ Integration Test โดยที่ระหว่าง Development จะต้องทำ Database มาก่อนเพื่อให้ “Line and Restful” และ “Matlab” เรียกใช้

STM 32 และ Raspberry Pi 3 จะใช้การเชื่อมต่อโดย UART

Matlab และ Database จะดึงโดยใช้ host, user, password ของ database ตรงๆ

Line และ Respberry Pi จะใช้ Restful และ MQTT



Integration Update (3/15/2017):

- Integrate STM32 and RPi (Communicate) using UART (protocal: mod bus).

- Line get data from ‘Wunderground’ API and commit to PostgreSQL database.

- Restful + Line

Sub module:

Pi + Camera + local database