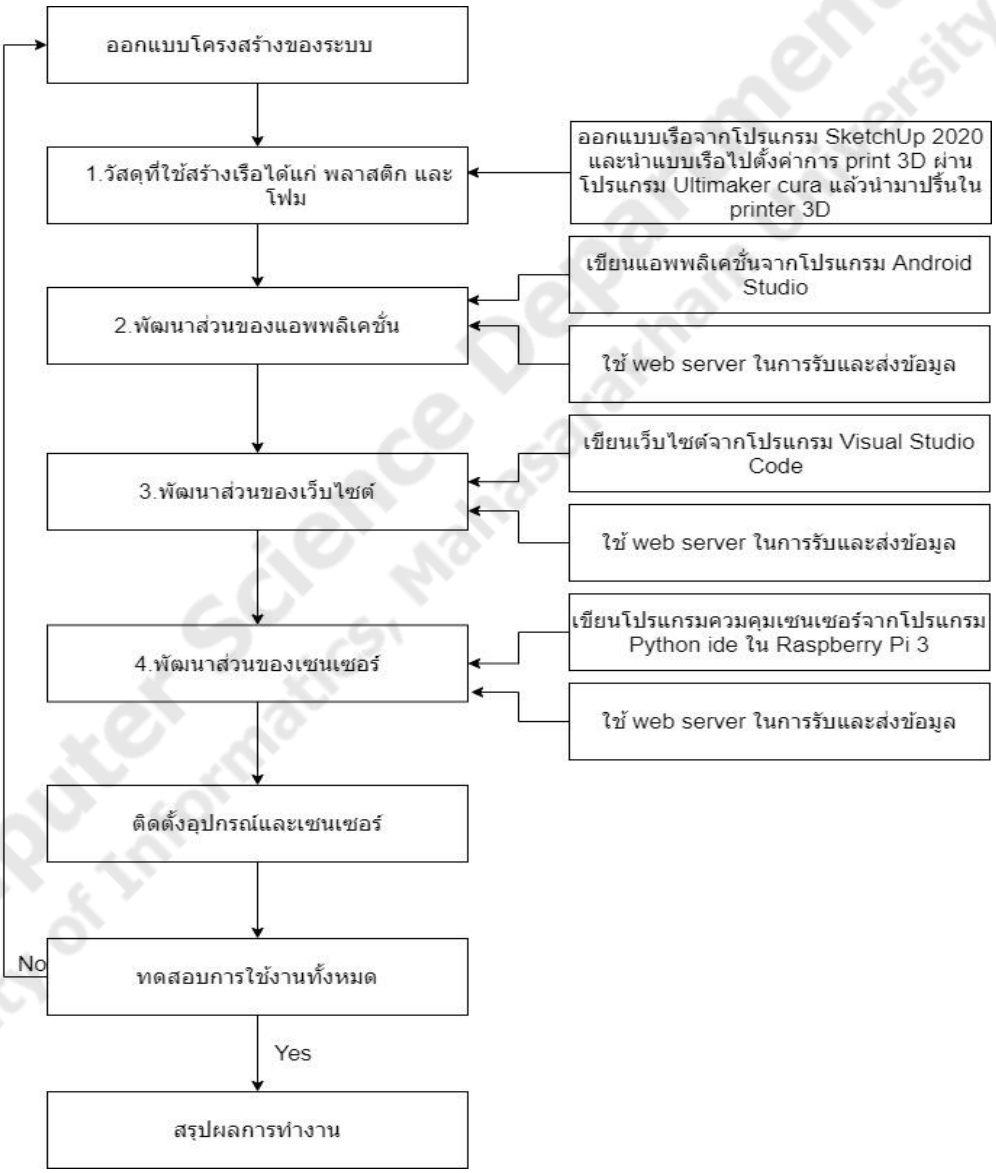
บทที่ 3

วิธีดำเนินงานวิจัย

3.1 กรอบการดำเนินงาน



ภาพประกอบที่ 3.1 กรอบการดำเนินงาน

Computer Science Department
Faculty of Informatics, Mahasarakham University

3.1.1 คำอธิบายการทำงานของระบบตรวจวัดระดับน้ำ

3.1.1.1 ออกแบบโครงสร้างของระบบ จะมีองค์ประกอบทั้งหมด 4 องค์ประกอบ คือ แอปพลิเคชัน เรือ Server และเว็บไซต์ โดยจะมี แอปพลิเคชันเป็นตัวการในการควบคุมเรือ ที่ใช้ในการเคลื่อนที่และวัดค่าข้อมูลต่างๆ ข้อมูลการควบคุมเรือและข้อมูลการวัดค่าต่างๆจะถูกนำขึ้นไปเก็บยังฐานข้อมูล ข้อมูลการวัดค่าต่างๆสามารถแสดงในแอปพลิเคชันและเว็บไซต์

3.1.1.2 การทำโมเดล ออกแบบโครงสร้างของชิ้นส่วนโมเดล ด้วยโปรแกรม SketchUp 2020 หลังจากนั้นทำการ export ไฟล์ออกมา เพื่อไปใช้งานที่โปรแกรม Ultimaker Cura และทำการเซฟไฟล์แล้วนำไปปริ้นที่เครื่อง Printer 3D ต่อไป และนำโฟมมาเป็นฐาน เพื่อช่วยในการลอยน้ำ

3.1.1.3 พัฒนาส่วนแอปพลิเคชัน จะใช้ ภาษา Java ซึ่งเขียนด้วยโปรแกรม Android studio ในการออกแบบแอปพลิเคชัน และหน้าที่ในส่วนของแอปพลิเคชัน คือ การควบคุมเรือ การวัดค่าต่างๆ การแสดงข้อมูลที่วัดได้ โดย การควบคุมเรือและการวัดค่าต่างๆ จะส่งข้อมูลคำสั่งไปที่ฐานข้อมูล แล้วฐานข้อมูลจะส่งข้อมูลคำสั่งไปที่เรือเพื่อทำการควบคุม เมื่อทำการวัดค่าต่างๆเสร็จแล้วจะนำข้อมูลไปเก็บไว้บนฐานข้อมูล และสามารถนำข้อมูลที่วัดได้มาแสดงที่แอปพลิเคชัน

3.1.1.4 พัฒนาเว็บไซต์ จะใช้ภาษา PHP ซึ่งเขียนด้วยโปรแกรม visual studio code ในการออกแบบหน้าเว็บไซต์ และหน้าที่ในส่วนของเว็บไซต์ คือ การแสดงข้อมูลวัดค่าต่างๆ โดยจะดึงข้อมูลที่เก็บไว้ในฐานข้อมูลมาแสดง

3.1.1.5 พัฒนาส่วนของเซนเซอร์ จะใช้ภาษา python ซึ่งเขียนด้วยโปรแกรม python ide และนำโค้ดของเซนเซอร์ต่างๆลงใน Raspberry Pi

3.1.1.6 การติดตั้งอุปกรณ์และเซนเซอร์ ลงในเรือทำการตรวจสอบตัวเรือ เพื่อดูว่าน้ำสามารถเข้ามาได้หรือไม่ และทำการติดตั้งอุปกรณ์และเซนเซอร์ให้เรียบร้อย

3.2 แผนภาพบริบท (Context Diagram)

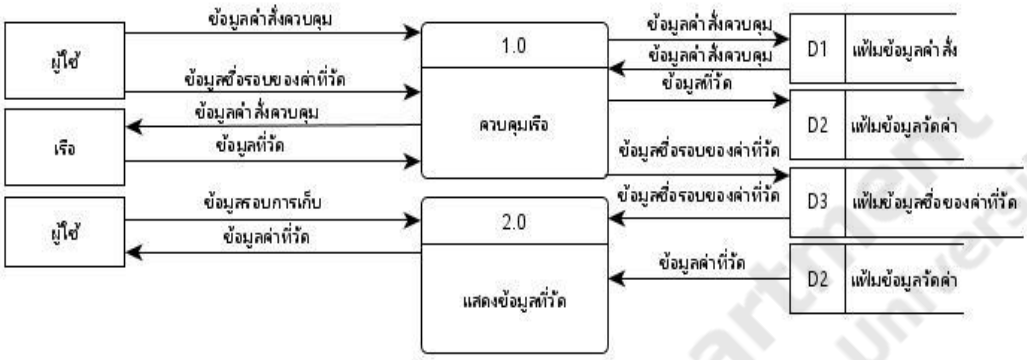
Context Diagram

ภาพประกอบที่ 3.2 Context Diagram

Computer Science Department
Faculty of Informatics, Mahasarakham University

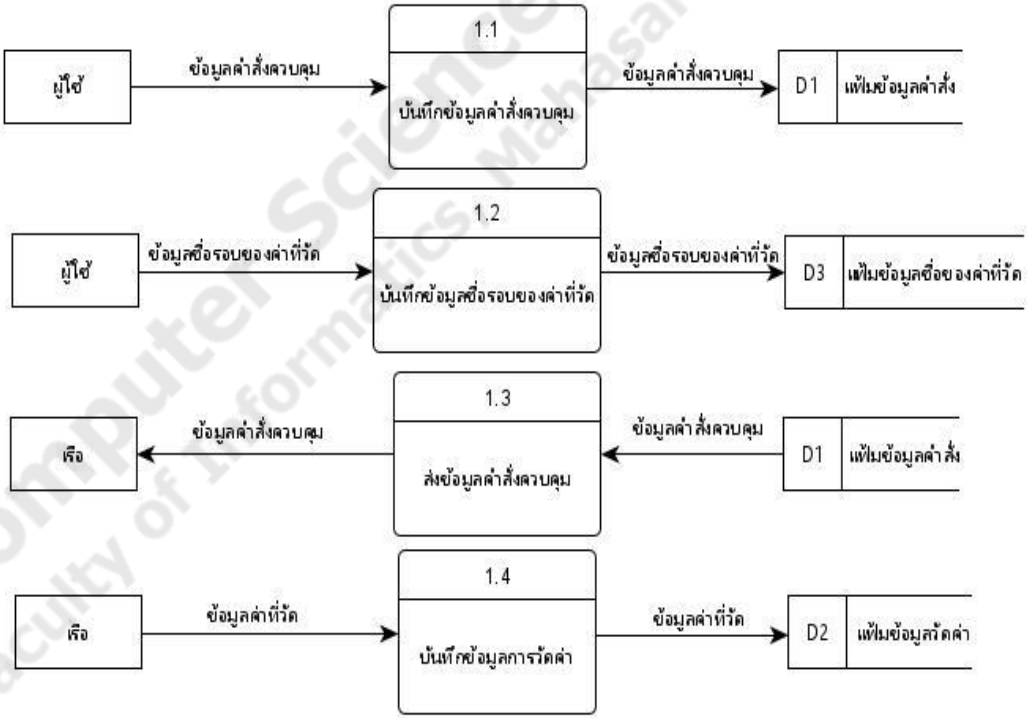
3.3 แผนภาพการไหลของข้อมูล (Data Flow Diagram)

Data Flow Diagram Level 1



ภาพประกอบที่ 3.3 Data Flow Diagram Level 1

Data Flow Diagram Level 2



ภาพประกอบที่ 3.4 Data Flow Diagram Level 2

Computer Science Department
Faculty of Informatics Mahasarakham University

3.4 External Entity Description

ตารางที่ 3.1 External Entity Descriptions

Name	Descriptions	Input Data Flow	Output Data Flow
ผู้ใช้	ผู้ใช้ควบคุมเรือผ่านแอพพลิเคชั่นและดูข้อมูลที่วัดค่าได้	-ข้อมูลค่าที่วัด	-ข้อมูลคำสั่งควบคุม
เรือ	เป็นเรือ ที่ถูกควบคุมผ่านแอพพลิเคชั่น	ข้อมูลคำสั่งควบคุม	-ข้อมูลค่าที่วัด

ตารางที่ 3.2 Data Flow (Data Flow Description and Data Structure of Data Flow)

Name	**Description**	**Source**	**Description**	**Data Structure**
ข้อมูลคำสั่งควบคุม	เป็นข้อมูลคำสั่งการเคลื่อนที่	ผู้ใช้	1.1 บันทึกข้อมูลคำสั่งควบคุม	คำสั่งควบคุม+ระยะเวลาการเคลื่อนที่
		1.1 บันทึกข้อมูลคำสั่งควบคุม	D1 แฟ้มข้อมูลคำสั่ง	
		D1 แฟ้มข้อมูลคำสั่ง	1.2 ส่งข้อมูลคำสั่งควบคุม	
		1.2 ส่งข้อมูลคำสั่งควบคุม	เรือ	
ข้อมูลชื่อรอบของค่าที่วัด	เป็นชื่อรอบของการวัด	ผู้ใช้	1.2 บันทึกข้อมูลชื่อรอบของค่าที่วัด	ชื่อรอบของการวัด
		1.2 บันทึกข้อมูลชื่อของค่าที่วัด	D3 แฟ้มข้อมูลชื่อของค่าที่วัด	
		D3 แฟ้มข้อมูลชื่อของค่าที่วัด	2.0 แสดงข้อมูลที่วัด	

Computer Science Department
Faculty of Informatics, Mahasarakham University

ตารางที่ 3.2 Data Flow (Data Flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Description	Data Structure
ข้อมูลค่าที่วัด	เป็นข้อมูลค่าที่วัด	เรือ	1.4 บันทึกข้อมูลการวัดค่า	ความลึก+PH+ความขุ่น+ละติจูด+ลองจิจูด
		1.4 บันทึกข้อมูลการวัดค่า	D2 แฟ้มข้อมูลวัดค่า	
		D2 แฟ้มข้อมูลวัดค่า	2.0 แสดงข้อมูลที่วัด	
		2.0 แสดงข้อมูลที่วัด	ผู้ใช้	
ข้อมูลรอบการเก็บ	เป็นชื่อข้อมูลรอบการเก็บค่า	ผู้ใช้	2.0 แสดงข้อมูลที่วัด	ชื่อรอบของการวัด
		2.0 แสดงข้อมูลที่วัด	D3 แฟ้มข้อมูลชื่อของค่าที่วัด	

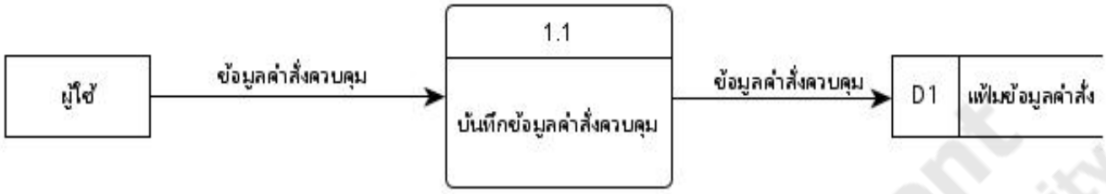
3.5 Data Store

ตารางที่ 3.3 Data Flow (Data Flow Description and Data Structure of Data store)

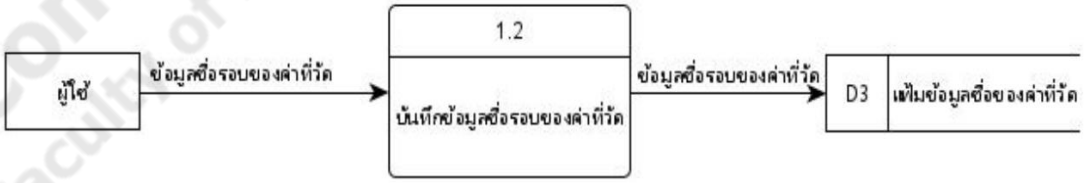
ID	Data Store	Description	Data Structure
D1	แฟ้มข้อมูลคำสั่ง	เก็บข้อมูลคำสั่งควบคุมเรือ	คำสั่งควบคุม+ระยะเวลาการเคลื่อนที่
D2	แฟ้มข้อมูลวัดค่า	เก็บข้อมูลวัดค่าที่ได้จากเซนเซอร์	ค่าระดับความลึก+ค่าPH+ลองจิจูด+ค่าละติจูด
D3	แฟ้มข้อมูลชื่อของค่าที่วัด	เก็บข้อมูลถานที่	ชื่อรอบของการวัด

Computer Science Department
Faculty of Informatics, Mahasarakham University

3.6 Process Description



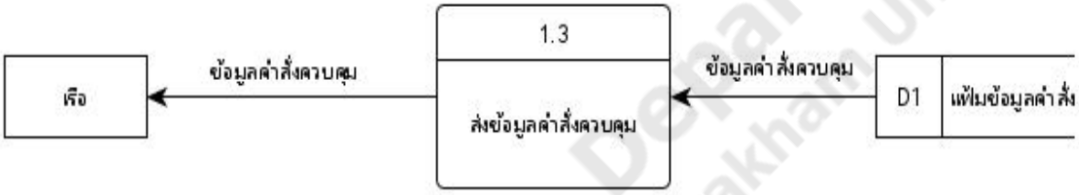
ID	1.1
NAME	บันทึกข้อมูลคำสั่งควบคุม
DESCRIPTION	การบันทึกข้อมูลคำสั่งควบคุม
INPUT DATA FLOW	-ข้อมูลคำสั่งควบคุม
OUTPUT DATA FLOW	-ข้อมูลคำสั่งควบคุม
PROCESS DESCRIPTION	เริ่มต้น 1. ส่งข้อมูลคำสั่งควบคุมจาก ผู้ใช้ 2. รับข้อมูลคำสั่งควบคุมจาก ผู้ใช้ ถ้า (ถูกต้องและครบถ้วน) รับข้อมูลคำสั่งควบคุม 3. บันทึกลงแฟ้มข้อมูลคำสั่ง จบการทำงาน



ID	1.2
NAME	ส่งข้อมูลชื่อรอบของค่าที่วัด
DESCRIPTION	การส่งข้อมูลชื่อรอบของค่าที่วัด
INPUT DATA FLOW	-ข้อมูลชื่อรอบของค่าที่วัด
OUTPUT DATA FLOW	-ข้อมูลชื่อรอบของค่าที่วัด

Computer Science Department
Faculty of Informatics, Mahasarakham University

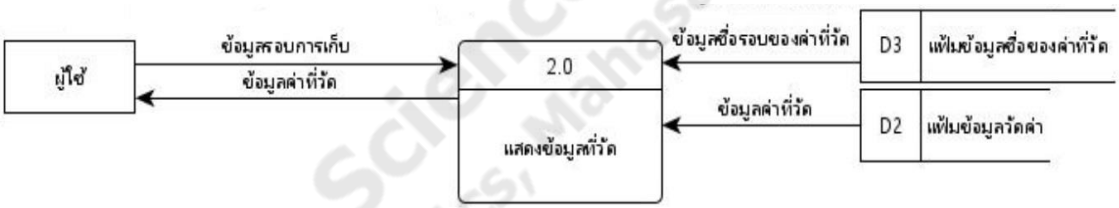
PROCESS DESCRIPTION	เริ่มต้น 1. ส่งข้อมูลชื่อรอบของค่าที่วัดจาก ผู้ใช้ 2. รับข้อมูลชื่อรอบของค่าที่วัดจาก ผู้ใช้ ถ้า (ถูกต้องและครบถ้วน) รับข้อมูลชื่อรอบของค่าที่วัด 3. บันทึกลงแฟ้มข้อมูลชื่อรอบของค่าที่วัด จบการทำงาน



ID	1.3
NAME	ส่งข้อมูลคำสั่งควบคุม
DESCRIPTION	การส่งข้อมูลคำสั่งควบคุม
INPUT DATA FLOW	-ข้อมูลคำสั่งควบคุม
OUTPUT DATA FLOW	-ข้อมูลคำสั่งควบคุม
PROCESS DESCRIPTION	เริ่มต้น 1. ส่งข้อมูลคำสั่งควบคุมจาก แฟ้มข้อมูลคำสั่ง 2. รับข้อมูลคำสั่งควบคุมจาก แฟ้มข้อมูลคำสั่ง ถ้า (ถูกต้องและครบถ้วน) รับข้อมูลคำสั่งควบคุม 3. ส่งไปยังเรือ จบการทำงาน

Computer Science Department
Faculty of Informatics Mahasarakham University

ID	1.4
NAME	บันทึกบันทึกข้อมูลการวัดค่า
DESCRIPTION	การบันทึกข้อมูลการวัดค่า
INPUT DATA FLOW	-ข้อมูลค่าที่วัด
OUTPUT DATA FLOW	-ข้อมูลค่าที่วัด
PROCESS DESCRIPTION	เริ่มต้น 1. ส่งข้อมูลค่าที่วัดจาก เรือ
PROCESS DESCRIPTION	2. รับข้อมูลค่าที่วัดจาก เรือ ถ้า (ถูกต้องและครบถ้วน) รับข้อมูลค่าที่วัด 4. บันทึกลงแฟ้มข้อมูลวัดค่า จบการทำงาน

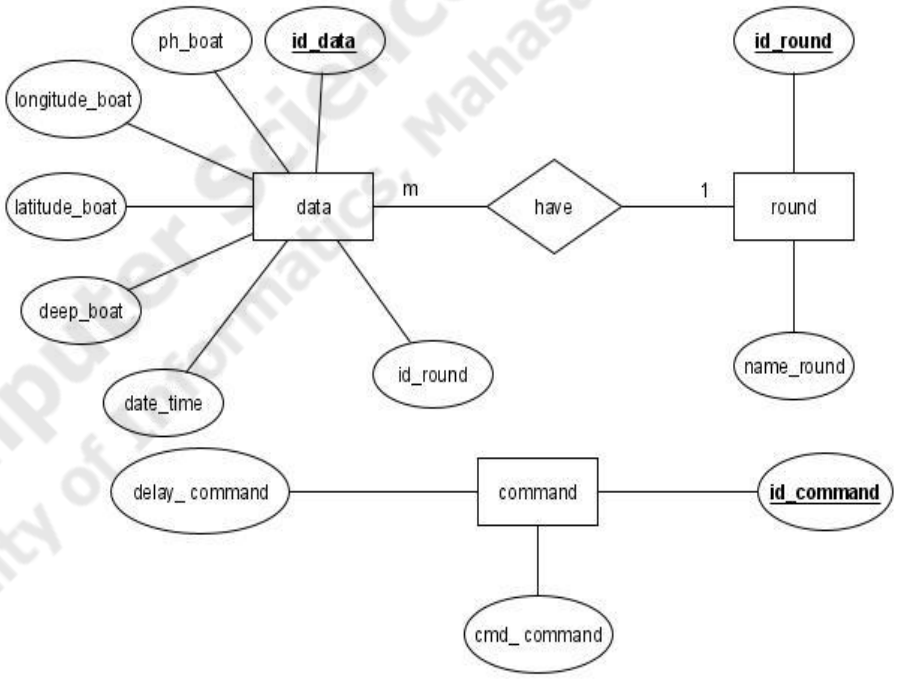


ID	2.0
NAME	แสดงข้อมูลที่วัดค่า
DESCRIPTION	การแสดงข้อมูลที่วัดค่า
INPUT DATA FLOW	-ข้อมูลรอบการเก็บ -ข้อมูลค่าที่วัดค่า -ข้อมูลชื่อรอบของค่าที่วัดค่า
OUTPUT DATA FLOW	-ข้อมูลคำสั่งวัดค่า

Computer Science Department
Faculty of Informatics, Mahasarakham University

PROCESS DESCRIPTION	เริ่มต้น 1. รับข้อมูลรอบการเก็บ จาก ผู้ใช้ 2. รับข้อมูลค่าที่วัดค่าจาก แฟ้มข้อมูลวัดค่า 3. รับข้อมูลชื่อรอบของค่าที่วัดจาก แฟ้มข้อมูลชื่อของค่าที่วัดค่า ถ้า (ถูกต้องและครบถ้วน) รับข้อมูลค่าที่วัดค่าค่า ถ้า (ถูกต้องและครบถ้วน) รับข้อมูลชื่อรอบของค่าที่วัดค่า 4. ส่งข้อมูลค่าที่วัดให้ ผู้ใช้ จบการทำงาน

3.7 External Relationship Diagram (ER-Diagram)



ภาพประกอบที่ 3.5 External Relationship Diagram (ER-Diagram)

Computer Science Department
Faculty of Informatics, Mahasarakham University

3.8 การออกแบบฐานข้อมูล (Database Design)

ตารางที่ 3.4 data ข้อมูลวัดค่าความลึกของน้ำ ค่า PH ค่าความขุ่น ค่าลองจิจูดและค่าละติจูด

id	Column	Type	Description	Example Data	Constraints
1	id_data	Int(5)	รหัสเรือ	1	PK
2	ph_boat	Varchar(20)	ค่า PH	7.0	-
3	longitude_boat	Varchar(20)	ลองจิจูด	185.2	-
4	latitude_boat	Varchar(5)	ละติจูด	158.3	-
5	turbidity_boat	Varchar(5)	ความขุ่น	300	-
6	deep_boat	Varchar(20)	ระดับความลึก	11	-
7	id_ round	Int(5)	รหัสสถานที่	1	FK: reference from round (id_ round) on delete no action on update cascade
8	date_ time	DATETIME	วันที่และเวลา	2020-10-10 10:10:10	-

ตารางที่ 3.5 command ข้อมูลคำสั่งควบคุมระบบตรวจวัดระดับน้ำ

id	Column	Type	Description	Example Data	Constraints
1	id_command	Int(5)	รหัสควบคุม	1	PK
2	cmd_ command	Varchar(10)	การเคลื่อนที่	go	-
3	delay_ command	Varchar(10)	กำหนดเวลาการเคลื่อนที่(s)	5	-

ตารางที่ 3.6 round ข้อมูลชื่อสถานที่

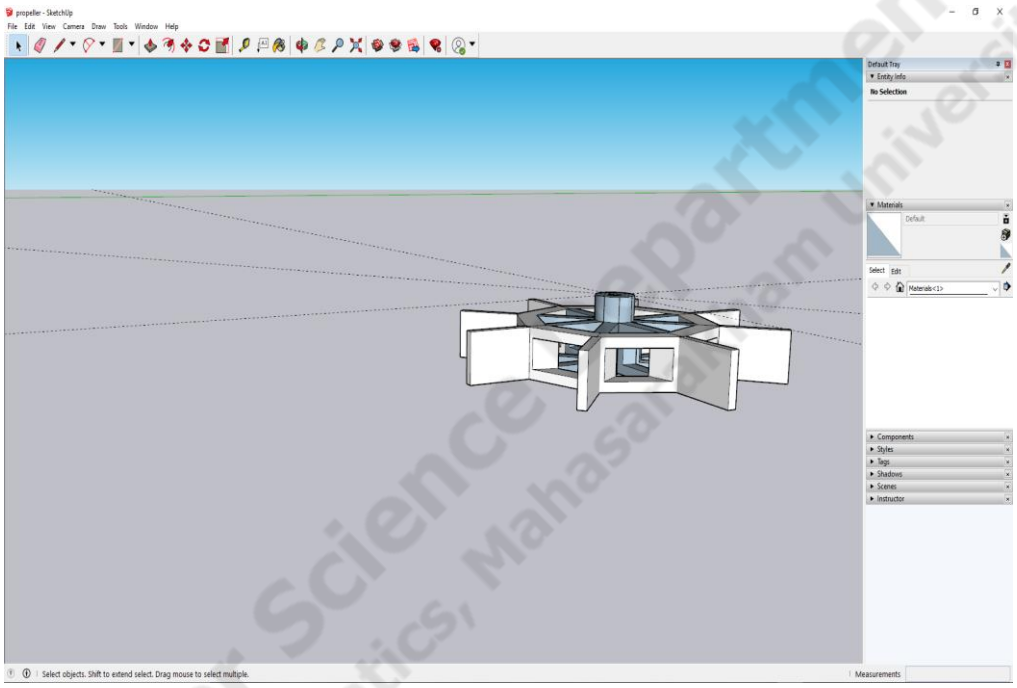
id	Column	Type	Description	Example Data	Constraints
1	id_ round	Int(5)	รหัสชื่อรอบ	1	PK
2	name_ round	Varchar(30)	ชื่อรอบที่วัด	สระหน้าคณะ	-

Computer Science Department
Faculty of Informatics, Mahasarakham University

3.9 การออกแบบโมเดล

การออกแบบชิ้นส่วนระบบตรวจวัดระดับน้ำ จะใช้โปรแกรม SketchUp 2020 และ โปรแกรม Ultimate Cura ซึ่งมีรายละเอียด ดังนี้

3.9.1 ออกแบบใบพัดเรือ

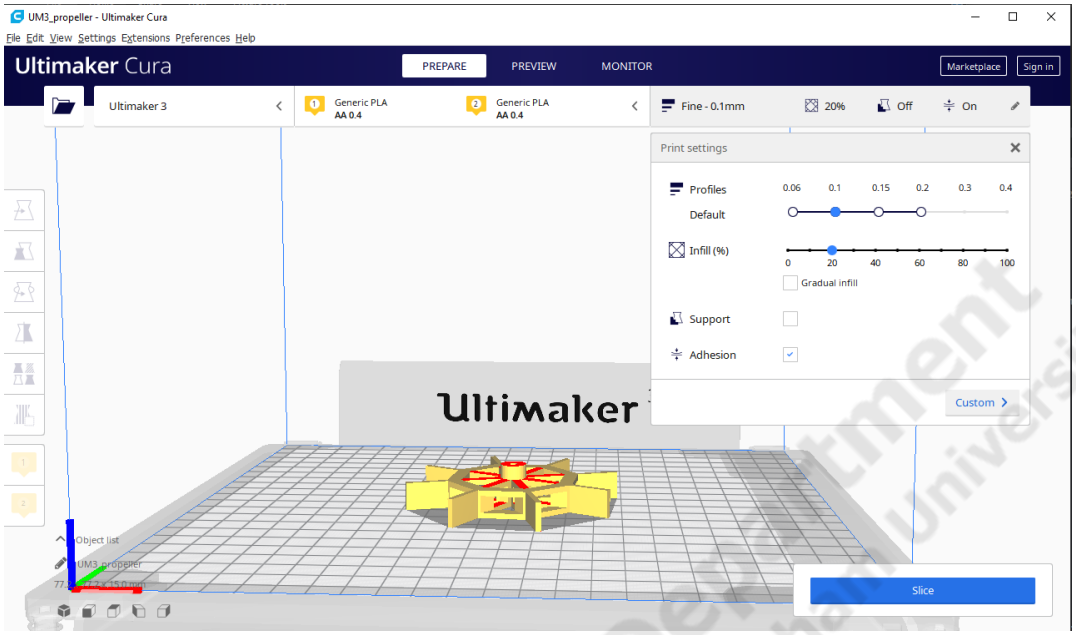


ภาพประกอบที่ 3.6 ออกแบบใบพัดระบบตรวจวัดระดับน้ำ

ที่มา : [7] เดชาธร พองาม, “sketchup วิธีใช้, ” . [Online]. Available: https://bit.ly/2U79PC3. [Accessed 26 August 2020].

จากภาพประกอบที่ 3.6 เป็นการออกแบบใบพัดเรือ ซึ่งมีความสูง 17 มิลลิเมตร ความกว้าง 75 มิลลิเมตร ความยาว 75 มิลลิเมตร โดยประมาณ

Computer Science Department Faculty of Informatics, Mahasarakham University

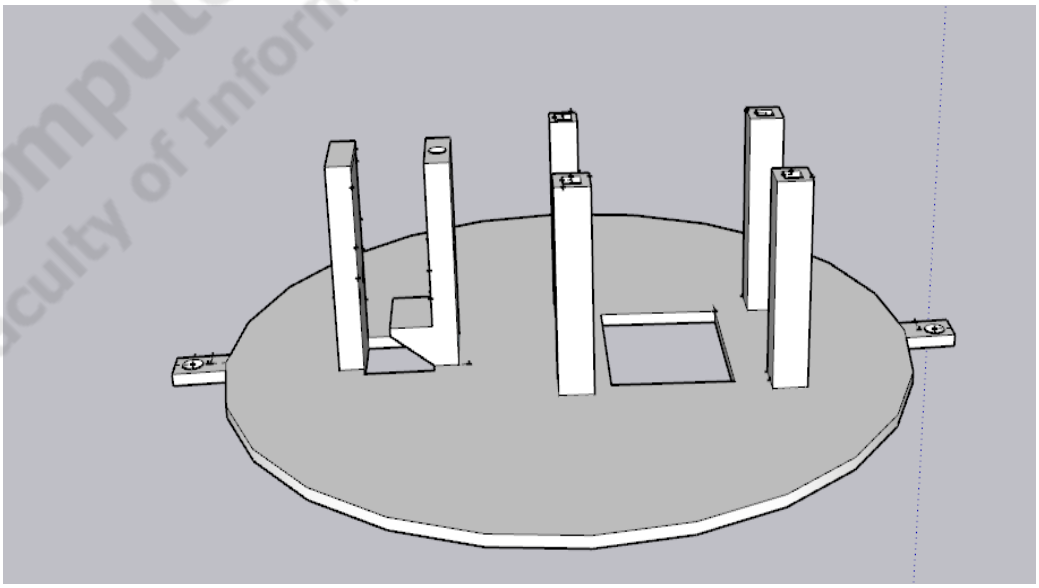


ภาพประกอบที่ 3.7 จำลองการปริ้นใบพัดระบบตรวจวัดระดับน้ำ

ที่มา: [6] Nattawat. “ultimaker cura วิธีใช้, ” 19 January 1016. [Online]. Available: https://bit.ly/3xksKYh. [Accessed 26 August 2019].

จากภาพประกอบที่ 3.7 เป็นการจำลองการปริ้น โดยการกำหนดค่าต่างๆ ที่จะใช้ในการปริ้น เช่น เลือกหัวปริ้น ความแข็งแรงของวัตถุ เส้นใยที่ใช้ในการปริ้น ความเร็วในการปริ้น เป็นต้น

3.9.2 ออกแบบฐานวัดระดับความลึก

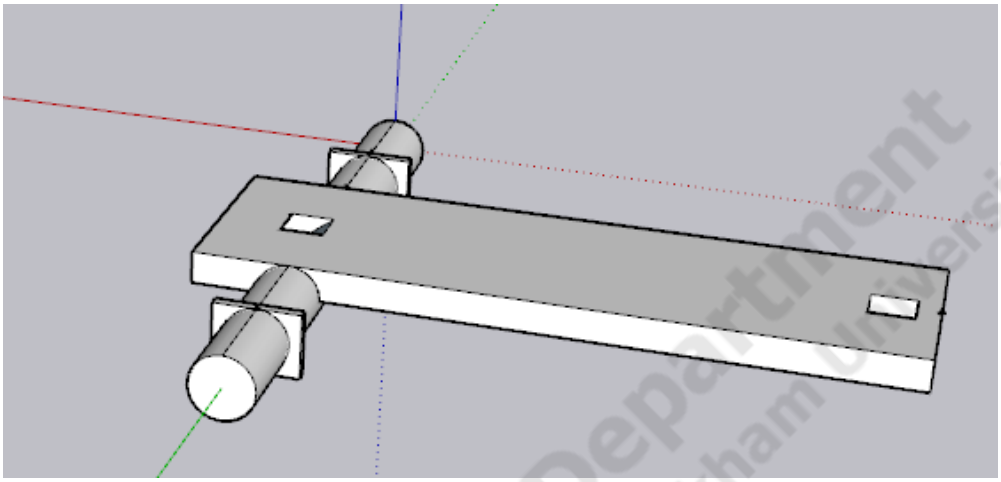


ภาพประกอบที่ 3.8 ฐานวัดระดับความลึก

Computer Science Department
Faculty of Informatics, Mahasarakham University

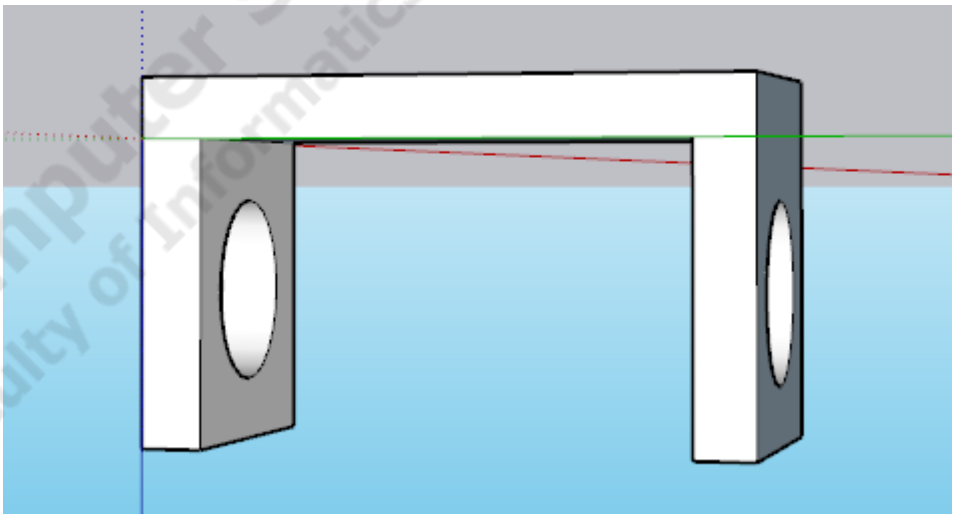
จากภาพประกอบที่ 3.8 เป็นการออกแบบฐานวัดระดับความลึก ซึ่งมีความสูง 50 มิลลิเมตร กว้าง 229 มิลลิเมตร ความยาว 229 มิลลิเมตร โดยประมาณ

ออกแบบคาน



ภาพประกอบที่ 3.9 ชิ้นส่วนคาน 1

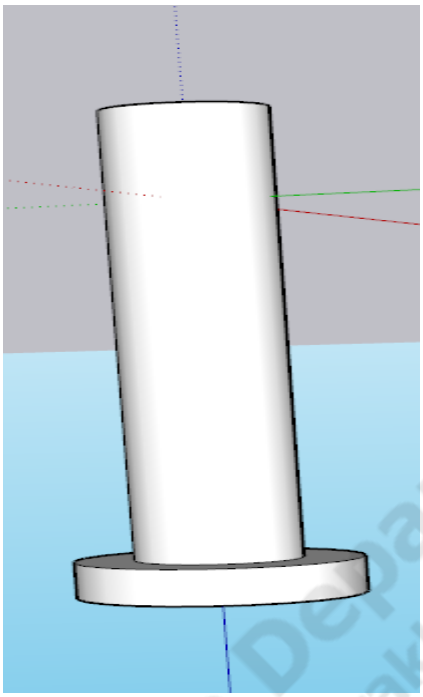
จากภาพประกอบที่ 3.9 เป็นการออกแบบคาน ซึ่งมี ความสูง 10 มิลลิเมตร กว้าง 40 มิลลิเมตร ความยาว 40 มิลลิเมตร โดยประมาณ ทำหน้าที่ยกลง เมื่อปล่อยลูกตุ้มลงน้ำ และจะยกขึ้น เมื่อลูกตุ้มถึงพื้นใต้น้ำ โดยมีเซนเซอร์กันชนเป็นตัวแปรในการให้มอเตอร์ที่อยู่ตรงที่เก็บเชือกว่าจะทำงานหรือไม่



ภาพประกอบที่ 3.10 ชิ้นส่วนคาน 2

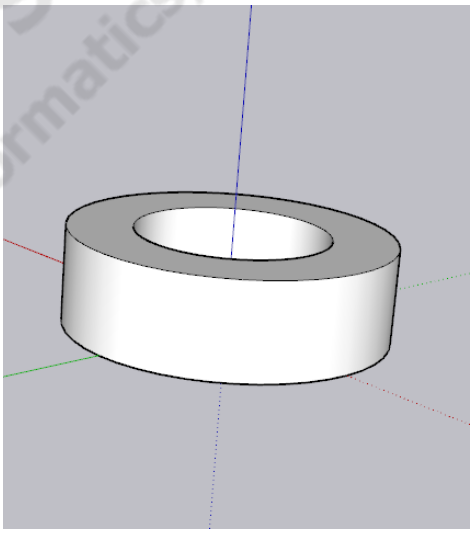
จากภาพประกอบที่ 3.10 เป็นการออกแบบคาน ซึ่งมี ความสูง 10 มิลลิเมตร กว้าง 10 มิลลิเมตร ความยาว 20 มิลลิเมตร โดยประมาณ

Computer Science Department Faculty of Informatics, Mahasarakham University



ภาพประกอบที่ 3.11 ชิ้นส่วนคาน 3

จากภาพประกอบที่ 3.11 เป็นการออกแบบคาน ซึ่งมี ความสูง 25 มิลลิเมตร กว้าง 10 มิลลิเมตร ความยาว 10 มิลลิเมตร โดยประมาณ

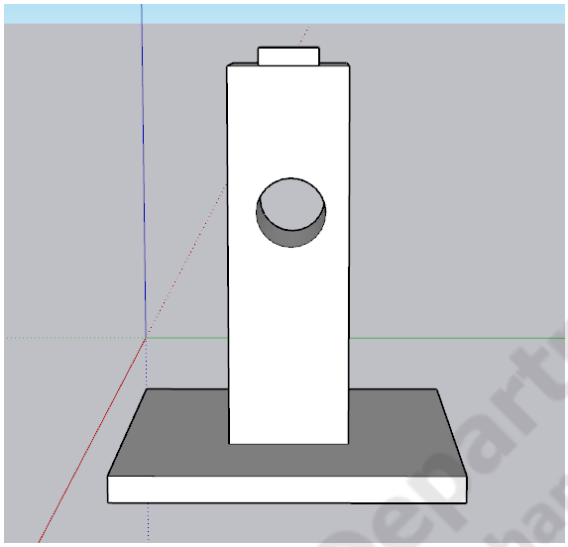


ภาพประกอบที่ 3.12 ชิ้นส่วนคาน 4

จากภาพประกอบที่ 3.12 เป็นการออกแบบคาน ซึ่งมี ความสูง 2.5 มิลลิเมตร กว้าง 10 มิลลิเมตร ความยาว 10 มิลลิเมตร โดยประมาณ

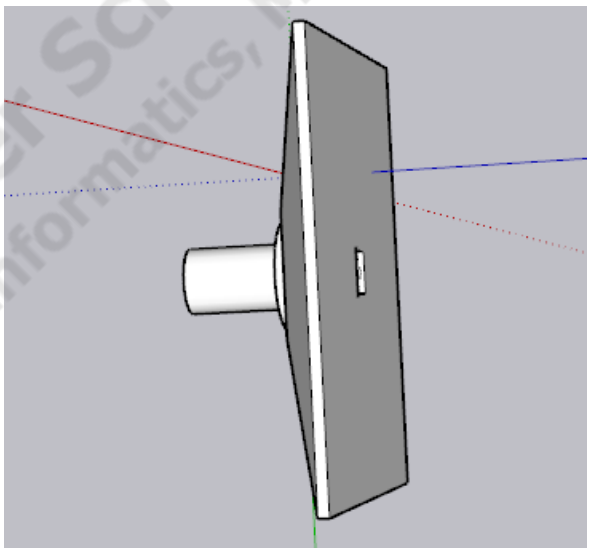
Computer Science Department
Faculty of Informatics, Mahasarakham University

3.9.3 ออกแบบที่เก็บเชือก



ภาพประกอบที่ 3.13 ชิ้นส่วนที่เก็บเชือก 1

จากภาพประกอบที่ 3.13 เป็นการออกแบบที่เก็บเชือก ซึ่งมี ความสูง 24.5 มิลลิเมตร กว้าง 19.5 มิลลิเมตร ความยาว 19.5 มิลลิเมตร โดยประมาณ

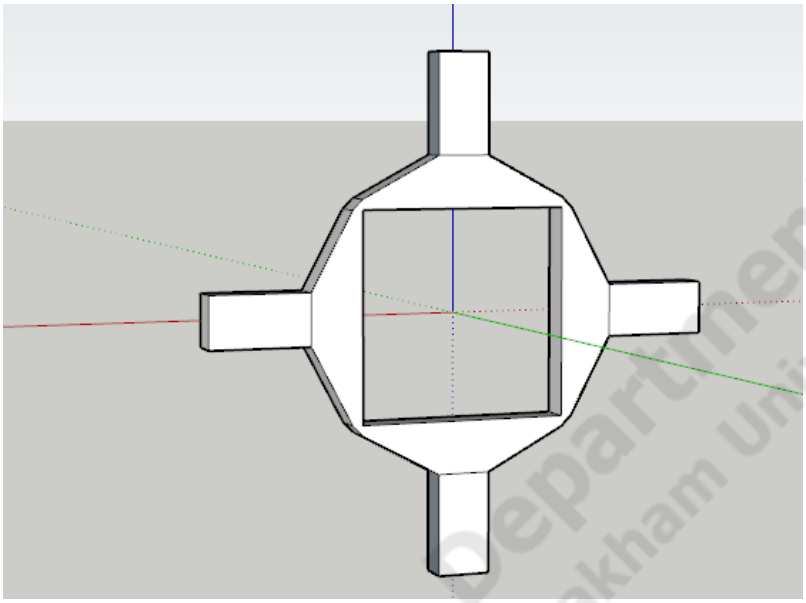


ภาพประกอบที่ 3.14 ชิ้นส่วนที่เก็บเชือก 2

จากภาพประกอบที่ 3.14 เป็นการออกแบบที่เก็บเชือก ซึ่งมี ความสูง 17 มิลลิเมตร กว้าง 41.5 มิลลิเมตร ความยาว 41.5 มิลลิเมตร ทำหน้าที่เก็บเชือก โดยมีมอเตอร์ทำหน้าที่หมุนเพื่อให้เก็บเชือก

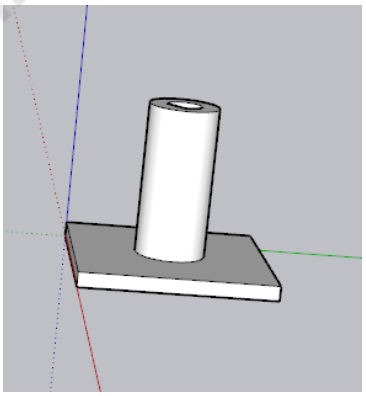
Computer Science Department
Faculty of Informatics, Mahasarakham University

3.9.4 ออกแบบการวัดรอบ



ภาพประกอบที่ 3.15 ชิ้นส่วนวัดรอบ 1

จากภาพประกอบที่ 3.15 เป็นการออกแบบที่วัดรอบ ซึ่งมี ความสูง 3 มิลลิเมตร กว้าง 49 มิลลิเมตร ความยาว 20 มิลลิเมตร โดยประมาณ ทำหน้าที่ช่วยให้เซนเซอร์นับรอบ โดย 1 รอบ เซนเซอร์นับรอบจะนักได้ 4 ครั้ง และจะคำนวณหาความยาวของเชือกตาม ครั้ง โดย 1 ครั้ง เชือกจะมีความยาวประมาณ 7 มิลลิเมตร โดยมีมอเตอร์ทำหน้าที่หมุนเพื่อให้วัดรอบ

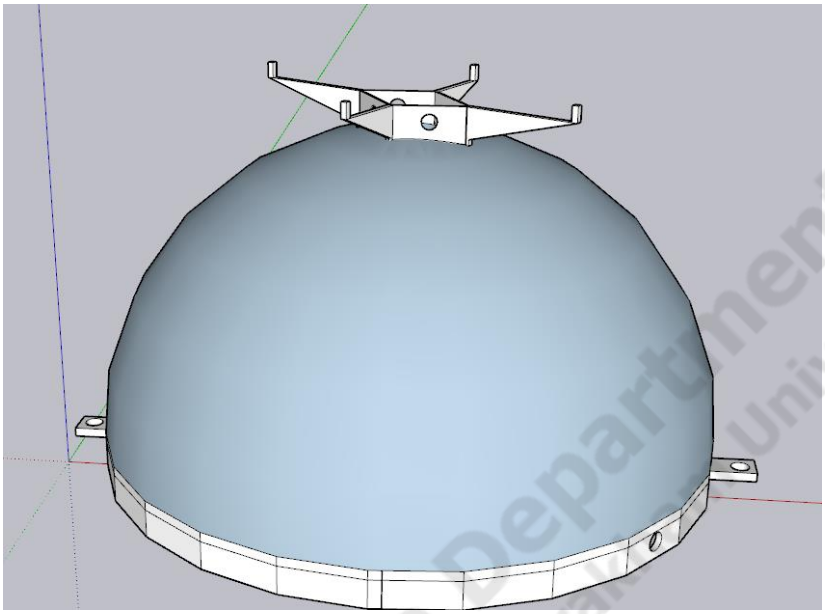


ภาพประกอบที่ 3.16 ชิ้นส่วนวัดรอบ 2

จากภาพประกอบที่ 3.16 เป็นการออกแบบที่วัดรอบ ซึ่งมี ความสูง 16.5 มิลลิเมตร กว้าง 19.5 มิลลิเมตร ความยาว 19.5 มิลลิเมตร โดยประมาณ

Computer Science Department
Faculty of Informatics, Mahasarakham University

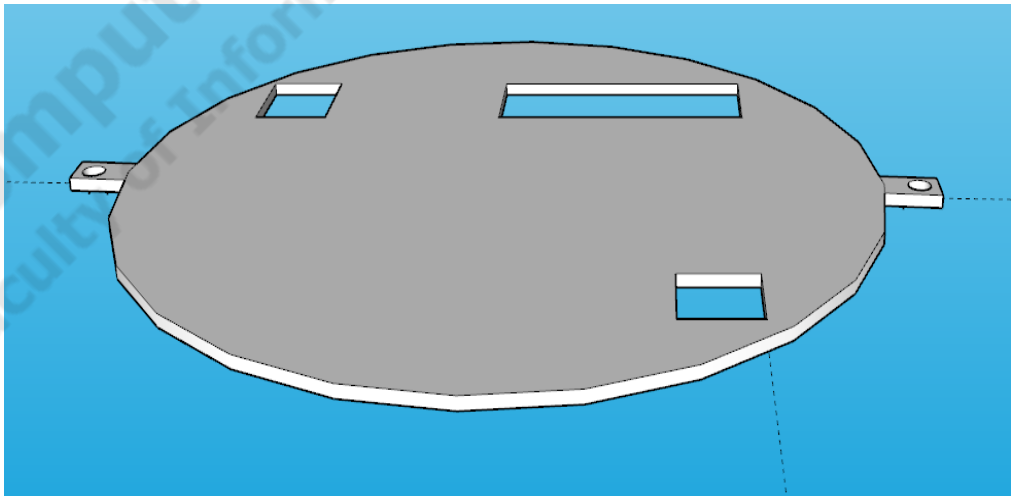
3.9.5 ออกแบบหลังคา



ภาพประกอบที่ 3.17 หลังคาของระบบตรวจวัดระดับน้ำ

จากภาพประกอบที่ 3.17 เป็นหลังคาที่ใช้ครอบอุปกรณ์บางส่วนของเรือซึ่งมี ความสูง 123 มิลลิเมตร ความกว้าง 229 มิลลิเมตร ความยาว 229 มิลลิเมตร โดยประมาณ บนสุดติดตั้งแผงโซล่าเซลล์

3.9.6 ออกแบบฐานวางอุปกรณ์ชั้นที่สอง

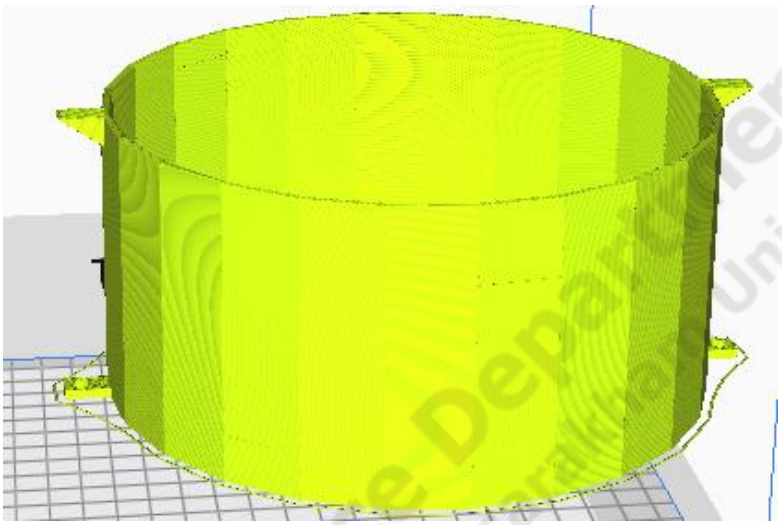


ภาพประกอบที่ 3.18 ฐานวางอุปกรณ์ชั้นที่สอง

Computer Science Department
Faculty of Informatics, Mahasarakham University

จากภาพประกอบที่ 3.18 ซึ่งมี ความสูง 3 มิลลิเมตร ความกว้าง 229 มิลลิเมตร ความยาว 229 มิลลิเมตร โดยประมาณ ใช้วางอุปกรณ์

3.9.7 ออกแบบส่วนครอบฐาน



ภาพประกอบที่ 3.19 ส่วนครอบฐาน

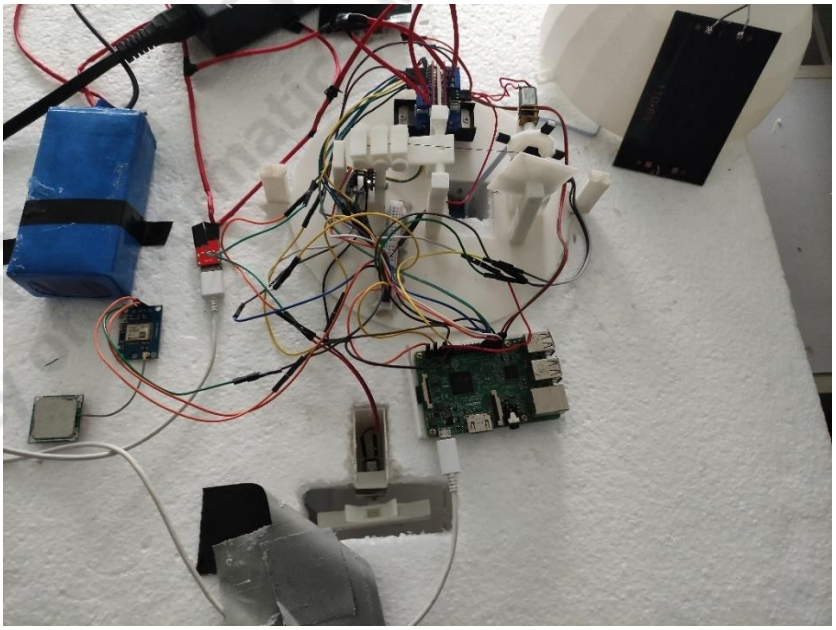
จากภาพประกอบที่ 3.19 ซึ่งมี ความสูง 98 มิลลิเมตร ความกว้าง 229 มิลลิเมตร ความยาว 229 มิลลิเมตร ใช้ล้อมอุปกรณ์

Computer Science Department
Faculty of Informatics, Mahasarakham University

3.10 ผลการออกแบบและพัฒนาระบบ

3.10.1 โมเดลระบบตรวจวัดระดับน้ำ

ภาพประกอบที่ 3.20 โมเดลระบบตรวจวัดระดับน้ำภายนอก



ภาพประกอบที่ 3.21 โมเดลระบบตรวจวัดระดับน้ำภายใน

ภาพประกอบที่ 3.22 ทุ่นลอยน้ำและตะกั่ว

3.10.2 การพัฒนาระบบในส่วนระบบตรวจวัดระดับน้ำ

```
1 import RPi.GPIO as GPIO
2 import time
3 time.sleep(60)
4 import serial
5 import string
6 import pynmea2
7 import requests
```

ภาพประกอบที่ 3.23 import library

ที่มา: [8] Ken RobotSiam, “การใช้งาน GPIO ใน Raspberry Pi ด้วย Python, ” . [Online]. Available: https://bit.ly/2TZYCmB. [Accessed 3 August 2020].

บรรทัดที่ 1 ถึง 7 เป็นการ import library ที่ใช้งาน

Computer Science Department Faculty of Informatics, Mahasarakham University

```
10 switch1 = 17
11 switch2 = 4
12 GPIO.setmode(GPIO.BCM)
13 GPIO.setup(switch1,GPIO.IN,pull_up_down=GPIO.PUD_UP)
14 GPIO.setup(switch2,GPIO.IN,pull_up_down=GPIO.PUD_UP)
15
16 state_switch1 = 0
17 state_switch2 = 0
18
19 GPIO.setup(27,GPIO.IN)
20 GPIO.setwarnings(False)
21
22 GPIO.setup(16,GPIO.OUT)
23 GPIO.setup(12,GPIO.OUT)
24 GPIO.setup(13,GPIO.OUT)
25
26 GPIO.setup(20,GPIO.OUT)
27 GPIO.setup(21,GPIO.OUT)
28
29 GPIO.setup(5,GPIO.OUT)
30 GPIO.setup(6,GPIO.OUT)
31
32 GPIO.setup(19,GPIO.OUT)
33 GPIO.setup(26,GPIO.OUT)
34
35 GPIO.output(20,GPIO.LOW)
36 GPIO.output(21,GPIO.LOW)
37
38 GPIO.output(5,GPIO.LOW)
39 GPIO.output(6,GPIO.LOW)
40
41 GPIO.output(19,GPIO.LOW)
42 GPIO.output(26,GPIO.LOW)
43
44 p=GPIO.PWM(16,1000)
45 p2=GPIO.PWM(12,1000)
46 p3=GPIO.PWM(13,1000)
47 p.start(10)
48 p2.start(10)
49 p3.start(10)
```

ภาพประกอบที่ 3.24 โค้ดการเซตช่องทางการรับ-ส่งข้อมูล

บรรทัดที่ 10 ถึง 49 เป็นโค้ดที่ใช้เซตช่องทางการรับ-ส่งข้อมูลระหว่างตัวเซนเซอร์และ ราสเบอร์รี่ พาย

Computer Science Department
Faculty of Informatics/Mahasarakham University

```
50  def set_command(command,delay):
51      print("Command  = ",command,"delay = ",delay)
52      delay = int(delay)
53      p2.ChangeDutyCycle(11)
54      p3.ChangeDutyCycle(10)
55
56
57      if(command=="go"):
58          print("go",delay)
59          GPIO.output(5,GPIO.HIGH)
60          GPIO.output(6,GPIO.LOW)
61          GPIO.output(19,GPIO.HIGH)
62          GPIO.output(26,GPIO.LOW)
63          time.sleep(int(delay))
64          GPIO.output(5,GPIO.LOW)
65          GPIO.output(6,GPIO.LOW)
66          GPIO.output(19,GPIO.LOW)
67          GPIO.output(26,GPIO.LOW)
68          return
69      elif(command=="back"):
70          GPIO.output(5,GPIO.LOW)
71          GPIO.output(6,GPIO.HIGH)
72          GPIO.output(19,GPIO.LOW)
73          GPIO.output(26,GPIO.HIGH)
74          time.sleep(int(delay))
75          GPIO.output(5,GPIO.LOW)
76          GPIO.output(6,GPIO.LOW)
77          GPIO.output(19,GPIO.LOW)
78          GPIO.output(26,GPIO.LOW)
79          return
80
81      elif (command=="left"):
82          p2.ChangeDutyCycle(5)
83          p3.ChangeDutyCycle(5)
84          GPIO.output(5,GPIO.HIGH)
85          GPIO.output(6,GPIO.LOW)
86          GPIO.output(19,GPIO.LOW)
87          GPIO.output(26,GPIO.LOW)
88          time.sleep(4)
89          GPIO.output(5,GPIO.LOW)
90          GPIO.output(6,GPIO.LOW)
91          return
92      elif (command=="right"):
93          p2.ChangeDutyCycle(5)
94          p3.ChangeDutyCycle(5)
95          GPIO.output(5,GPIO.LOW)
96          GPIO.output(6,GPIO.LOW)
97          GPIO.output(19,GPIO.HIGH)
98          GPIO.output(26,GPIO.LOW)
99          time.sleep(4)
100         GPIO.output(19,GPIO.LOW)
101         GPIO.output(26,GPIO.LOW)
102         return
103     return
```

ภาพประกอบที่ 3.25 โค้ดฟังก์ชันการควบคุมเซ็นเซอร์จากการรับคำสั่ง

ฟังก์ชันการควบคุมเซ็นเซอร์จากการรับคำสั่ง

บรรทัดที่ 50 set_command(command,delay) รอรับค่าคำสั่ง go ,back, left, right และ ค่าระยะเวลา

บรรทัดที่ 51 ให้แสดงค่า command และ ค่า day

บรรทัดที่ 52 แปลงค่า delay ให้เป็น int

บรรทัดที่ 53 กำหนดรอบของมอเตอร์ด้านขวา 11%

บรรทัดที่ 54 กำหนดรอบของมอเตอร์ด้านซ้าย 10%

Computer Science Department
Faculty of Informatics Mahasarakham University

บรรทัดที่ 57 fi(command == “go”) ถ้าคำสั่งที่รับเป็นสตริงเข้ามามีค่าเท่ากับ “go” จะสั่งมอเตอร์หมุนเพื่อให้เรือเคลื่อนที่ไปข้างหน้า
บรรทัดที่ 59 กำหนดขั้วบวกของมอเตอร์ตัวด้านขวาเป็น HIGH
บรรทัดที่ 60 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านขวาเป็น LOW
บรรทัดที่ 61 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น HIGH
บรรทัดที่ 62 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW
บรรทัดที่ 63 สั่งหน่วงเวลาการหมุนของมอเตอร์ด้วยคำสั่ง time.sleep(int(delay)) จาก delay ที่รับเข้ามา
บรรทัดที่ 54 กำหนดขั้วบวกของมอเตอร์ตัวด้านขวาเป็น LOW
บรรทัดที่ 65 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านขวาเป็น LOW
บรรทัดที่ 66 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW
บรรทัดที่ 67 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW
บรรทัดที่ 69 elif(command == “back”) ถ้าคำสั่งที่รับเป็นสตริงเข้ามามีค่าเท่ากับ “back” จะสั่งมอเตอร์หมุนเพื่อให้เรือเคลื่อนที่ถอยหลัง
บรรทัดที่ 70 กำหนดขั้วบวกของมอเตอร์ตัวด้านขวาเป็น LOW
บรรทัดที่ 71 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านขวาเป็น HIGH
บรรทัดที่ 72 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW
บรรทัดที่ 73 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น HIGH
บรรทัดที่ 74 สั่งหน่วงเวลาการหมุนของมอเตอร์ด้วยคำสั่ง time.sleep(int(delay)) จาก delay ที่รับเข้ามา
บรรทัดที่ 75 กำหนดขั้วบวกของมอเตอร์ตัวด้านขวาเป็น LOW
บรรทัดที่ 76 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านขวาเป็น LOW
บรรทัดที่ 77 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW
บรรทัดที่ 78 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW
บรรทัดที่ 81 elif(command == “left”) ถ้าคำสั่งที่รับเป็นสตริงเข้ามามีค่าเท่ากับ “left” จะสั่งมอเตอร์หมุนเพื่อให้เรือเลี้ยวซ้าย
บรรทัดที่ 82 กำหนดรอบของมอเตอร์ด้านขวา 5%
บรรทัดที่ 83 กำหนดรอบของมอเตอร์ด้านซ้าย 5%
บรรทัดที่ 84 กำหนดขั้วบวกของมอเตอร์ตัวด้านขวาเป็น HIGH
บรรทัดที่ 85 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านขวาเป็น LOW
บรรทัดที่ 86 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW

Computer Science Department
Faculty of Informatics Mahasarakham University

บรรทัดที่ 87 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW

บรรทัดที่ 88 สั่งหน่วงเวลาการหมุนของมอเตอร์ 4 วินาที

บรรทัดที่ 89 กำหนดขั้วบวกของมอเตอร์ตัวด้านขวาเป็น LOW

บรรทัดที่ 90 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านขวาเป็น LOW

บรรทัดที่ 92 elif(command == “right”) ถ้าคำสั่งที่รับเป็นสตริงเข้ามามีค่าเท่ากับ “right” จะสั่งมอเตอร์หมุนเพื่อให้เรือเลี้ยวขวา

บรรทัดที่ 93 กำหนดรอบของมอเตอร์ด้านขวา 5%

บรรทัดที่ 94 กำหนดรอบของมอเตอร์ด้านซ้าย 5%

บรรทัดที่ 95 กำหนดขั้วบวกของมอเตอร์ตัวด้านขวาเป็น LOW

บรรทัดที่ 96 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านขวาเป็น LOW

บรรทัดที่ 97 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น HIGH

บรรทัดที่ 98 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW

บรรทัดที่ 99 สั่งหน่วงเวลาการหมุนของมอเตอร์ 4 วินาที

บรรทัดที่ 100 กำหนดขั้วบวกของมอเตอร์ตัวด้านซ้ายเป็น LOW

บรรทัดที่ 101 กำหนดให้ขั้วลบของมอเตอร์ตัวด้านซ้ายเป็น LOW

ฟังกชันส่งค่าขึ้นเซิฟเวอร์

```
105 def save_data(lat,longt,p,tur,deep,loca):
106
107     latitude = lat
108     longtitude = longt
109     ph = p
110     turbidity = tur
111     deep = deep
112     location = loca
113     url = "http://tradeoptionth.com/api/insert_data.php"
114     payload = 'insert_data=1&latitude_boat='+ latitude +'&longtitude_boat='+longtitude+'&ph_boat='+ph+'&turbidity_boat='+turbidity+'&deep_boat='+deep+'&location='+location
115     headers = {'Content-Type': 'application/x-www-form-urlencoded'}
116     response = requests.request("POST", url, headers=headers, data=payload)
117
118     print(response.text)
119     return
```

ภาพประกอบที่ 3.26 ฟังกชันส่งค่าขึ้นเซิร์ฟเวอร์

บรรทัดที่ 113 กำหนด url ของเซิฟเวอร์

บรรทัดที่ 114 สร้างตัวแปรมาเก็บค่าข้อมูล latitude, longtitude, ph, turbidity , deep, location

บรรทัดที่ 115 สร้างตัวแปร headers

บรรทัดที่ 116 ส่งข้อมูลแบบ POST ไปยังเซิร์ฟเวอร์

Computer Science Department
Faculty of Informatics, Mahasarakham University

ฟังก์ชันดึงตะกั่วขึ้น

```
123  def pull_up(location,deep):
124      k = 0
125      i = 0.0
126      n = 1
127      dep = deep
128
129
130      input_state = GPIO.input(switch1)
131      p.ChangeDutyCycle(16)
132      GPIO.output(20,GPIO.LOW)
133      GPIO.output(21,GPIO.HIGH)
134
135      lat = 0.0
136      lng = 0.0
137      while True:
138
139
140          port="/dev/ttyAMA0"
141          ser=serial.Serial(port, baudrate=9600, timeout=0.1)
142          dataout = pynmea2.NMEAStreamReader()
143          newdata=ser.readline()
144
145          if newdata[0:6] == "$GPGLL":
146              newmsg=pynmea2.parse(newdata)
147              lat=newmsg.latitude
148              lng=newmsg.longitude
149
150
151          input_state2 = GPIO.input(switch2)
152          if input_state2 == False :
153
154
155              save_data(str(lat),str(lng),"7.5","20",str(dep),location)
156              GPIO.output(20,GPIO.LOW)
157              GPIO.output(21,GPIO.LOW)
158              return
159
```

ภาพประกอบที่ 3.27 ฟังก์ชันดึงตะกั่วขึ้น

บรรทัดที่ 130 รับข้อมูลสถานะของเซนเซอร์กันชน1

บรรทัดที่ 131 กำหนดรอบของมอเตอร์ 16%

บรรทัดที่ 140 ถึง 148 เป็นฟังก์ชันการเรือกใช้ จีพีเอส โมดูล

บรรทัดที่ 151 รับข้อมูลสถานะของเซนเซอร์กันชน2

บรรทัดที่ 152 ถึง 158 ทำการตรวจสอบตัวกันชน2 ถ้าเข้าเงื่อนไขจะทำการรีเทิร์นข้อมูล

ฟังก์ชันปล่อยตะกั่วลง

Computer Science Department
Faculty of Informatics Mahasarakham University

```
def set_cmd(lo):
    l = lo
    i = 0.0
    n = 1
    x = 0
    input_count=True
    while True:
        input_state = GPIO.input(switch1)
        if input_state == False  :
            p.ChangeDutyCycle(0.9)
            GPIO.output(20,GPIO.HIGH)
            GPIO.output(21,GPIO.LOW)
            input_count = GPIO.input(27)
            if input_count == True:
                #print('true')
                if n == 1 :
                    i = i+1
                    #print('Pass ' ,i)
                    n=0
                    input_state = False
                    time.sleep(0.1)
            elif input_state == False:
                n=1

        if input_state == True  :
            deep = i*7/500
            GPIO.output(20,GPIO.LOW)
            GPIO.output(21,GPIO.LOW)
            time.sleep(2)
            pull_up(1,deep)
            break

    return
```

ภาพประกอบที่ 3.28 ฟังก์ชันปล่อยตะกั่วลง

บรรทัดที่ 171 ตรวจสอบสถานะตัวกันชน 1

บรรทัดที่ 172 ถึง 175 สั่งให้มอเตอร์หมุนทำการปล่อยตะกั่วลง เมื่อ ตัวกันชน 1 มีสถานะเป็น Fales

บรรทัดที่ 176 ถึง 184 ทำการนับจำนวนรอบ

บรรทัดที่ 187 ถึง 193 สั่งให้มอเตอร์หยุดหมุนทำการคำนวณค่าความลึก deep = i*7/500 (6)

ฟังก์ชันดึงคำสั่งจากเซิฟเวอร์

```
def call_command():
    response = requests.get('http://tradeoptionth.com/api/get_command.php')
    data = response.json()
    datap = data["Command"]
    datac = datap[0]
    return {'id': datac.get("id"),'command':datac.get("command"),'delay':datac.get("delay"),'location':datac.get("location")}

list_cmd0 = call_command()
ck = int(list_cmd0["id"])
```

ภาพประกอบที่ 3.29 โค้ดฟังก์ชันรอดึงข้อมูลคำสั่งจากเซิฟเวอร์

บรรดทัดที่ 198 กำหนด url ที่ต้องการดึงข้อมูลคำสั่ง

Computer Science Department
Faculty of Informatics Mahasarakham University

บรรทัดที่ 199 สร้างตัวแปรมารับชุดคำสั่งเป็น dict

บรรทัดที่ 202 ส่งกลับข้อมูลคำสั่งประกอบด้วย ‘id’,’command’,’delay’

ฟังก์ชันเรียกการทำงานของโปรแกรม โดยจะทำงานทุก ๆ 1 วินาที

```
207 while True:
208     list_cmd = call_command()
209     id_command = list_cmd["id"]
210     command = list_cmd["command"]
211     delay = list_cmd["delay"]
212     location = list_cmd["location"]
213     x = int(id_command)
214 
215     if (x > ck):
216         print(id_command,command,delay,location)
217         if command == "1":
218             set_cmd(location)
219         else:
220             set_command(command, delay)
221 
222     else:
223         print("WAIT...")
224     ck = x
225     time.sleep(1)
```

ภาพประกอบที่ 3.30 โค้ดฟังก์ชันเรียกการทำงานของโปรแกรม

บรรทัดที่ 64 กำหนด id คำสั่งเริ่มต้นเก็บไว้ที่ตัวแปรชื่อ ck

บรรทัดที่ 65 สร้างลูปด้วย while True: และบรรทัดที่ 77 กำหนดการหน่วงเวลาของลูปด้วยคำสั่ง time.sleep(1) เพื่อให้หน่วงเวลาของลูป

บรรทัดที่ 70 สร้างตัวแปรเพื่อเก็บ id คำสั่งที่รับเข้ามาใหม่ไว้ที่ตัวแปรชื่อ x

บรรทัดที่ 72 if(x > ck) ถ้า x เป็นคำสั่งที่เรียกมาใหม่มีค่ามากกว่า ck ที่เป็น id คำสั่งเริ่มต้น จะส่งคำสั่งไปยังฟังก์ชันควบคุมเซ็นเซอร์โดยจะส่ง command เป็นคำสั่งที่รับเข้ามาใหม่ และ delay ที่รับเข้ามาใหม่บรรทัดที่ 73

บรรทัดที่ 74 ถ้าตัวแปร x เป็นคำสั่งที่รับเข้ามาใหม่มีค่าน้อยกว่าหรือเท่ากับตัวแปร ck ที่เป็นตัวแปรเริ่มต้น โปรแกรมจะแสดงข้อความ “WAIT...” และเริ่มทำงานใหม่ไปเรื่อย ๆ จนกว่าจะได้รับคำสั่งเข้ามาใหม่

Computer Science Department Faculty of Informatics, Mahasarakham University

3.10.3 พัฒนาในส่วนแอปพลิเคชัน

ตัวอย่าง หน้าหลัก UI Interface ของแอปพลิเคชัน

ภาพประกอบที่ 3.31 หน้าหลัก UI Interface ของแอปพลิเคชัน

ที่มา : [5] Teerasej Jiraphatchandej, “android studio วิธีใช้,”19 April 2017. [Online]. Available: https://bit.ly/3womCOK. [Accessed 25 August 2020].

เส้นด้านล่าง Location คือ ที่ให้กรอกชื่อของสถานที่ Delay คือระยะเวลาในการเคลื่อนที่ กรอบสีแดงทีมีคำว่า PROCESS คือ ปุ่มการวัดค่า กรอบสีเทาที่มีคำว่า SHOW คือปุ่มแสดงข้อมูล ลูกศร บน ล่าง ซ้าย ขวา คือ ปุ่มควบคุมการเคลื่อนที่ วงกลมสีแดงมีสี่เหลี่ยมตรงกลาง คือปุ่มหยุดการเคลื่อนที่ import library ที่ใช้

Computer Science Department
Faculty of Informatics, Mahasarakham University

```
import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.media.Image;
import android.os.AsyncTask;
import android.os.Build;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.SeekBar;
import android.widget.TextView;
import android.widget.Toast;
import com.example.wathermeansurement.helper.WebServicesHelper;
import com.example.wathermeansurement.utility.TagUtils;
import org.json.JSONException;
import org.json.JSONObject;
import java.util.Random;
```

ภาพประกอบที่ 3.32 โค้ดการ import library

ประกาศตัวแปร ประเภท Component สำหรับแสดงผล และการทำงานในส่วนต่าง ๆ ภายในแอปพลิเคชันมือถือ

```
public class MainActivity extends AppCompatActivity {
    String id_buoy =null,location = null;
    int prodressValue = 0;
    String buoy_data ="0";
    private TextView name_buoy_id,result1,txTime,txDelay;
    String var_command = "";
    String var_process = "";
    String delay = "5";
    private Button button3,button_sw;
    private View image_down,image_up,image_left,image_right,image_stop;
    SeekBar seekBar1 ;
    private EditText editText;

    private MainActivity ma;
    private Handler Unirest;
```

ภาพประกอบที่ 3.33 ประกาศตัวแปร ประเภท Component

Computer Science Department Faculty of Informatics Mahasarakham University

ประกาศตัวแปรเพื่อรับข้อมูลจาก Component ผ่าน ID

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    name_buoy_id = findViewById(R.id.name_buoy_id);
    ma = this;
    button3 = findViewById(R.id.button3);
    image_down = findViewById(R.id.image_down);
    image_left = findViewById(R.id.image_left);
    image_right = findViewById(R.id.image_right);
    image_up = findViewById(R.id.image_up);
    image_stop = findViewById(R.id.image_stop);
    seekBar1 = findViewById(R.id.seekBar);
    txTime = findViewById(R.id.tx_time);
    txDelay = findViewById(R.id.txDelay);
    button_sw = findViewById(R.id.button_show);
    editText = findViewById(R.id.editTextTex1);
```

ภาพประกอบที่ 3.34 การประกาศตัวแปรเพื่อรับข้อมูลจาก Component ผ่าน ID

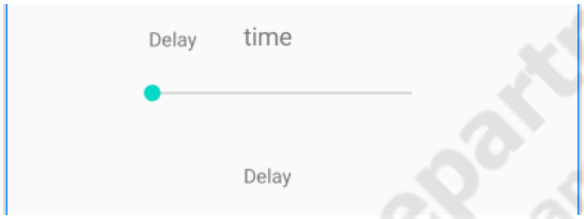
กำหนดคำสั่งของปุ่มต่าง ๆ ให้ทำงานเมื่อมีการกด

```
image_down.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        var_command = "back";
        init_save_datatoDB_speed();
    }
});
image_up.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        var_command = "go";
        init_save_datatoDB_speed();
    }
});
image_left.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        var_command = "left";
        init_save_datatoDB_speed();
    }
});
image_right.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        var_command = "right";
        init_save_datatoDB_speed();
    }
});
```

ภาพประกอบที่ 3.35 โค้ดกำหนดคำสั่งของปุ่มต่าง ๆ

Computer Science Department Faculty of Information Technology Mahasarakham University

บรรทัดที่ 50 เมื่อมีการกดปุ่ม down จะกำหนดคำสั่งให้เป็น “back” ที่บรรทัดที่ 53
บรรทัดที่ 54 เรียกใช้ฟังก์ชัน init_save_dataDB_speed() เพื่อบันทึกคำสั่งไปยังฐานข้อมูล
บรรทัดที่ 57 เมื่อมีการกดปุ่ม up จะกำหนดคำสั่งให้เป็น “go” ที่บรรทัดที่ 60
บรรทัดที่ 61 เรียกใช้ฟังก์ชัน init_save_dataDB_speed() เพื่อบันทึกคำสั่งไปยังฐานข้อมูล
บรรทัดที่ 64 เมื่อมีการกดปุ่ม left จะกำหนดคำสั่งให้เป็น “left” ที่บรรทัดที่ 67
บรรทัดที่ 68 เรียกใช้ฟังก์ชัน init_save_dataDB_speed() เพื่อบันทึกคำสั่งไปยังฐานข้อมูล
การสร้างและกำหนดค่าให้กับคอมโพเนนท์ Seekbar และ textView เพื่อกำหนด



ภาพประกอบที่ 3.36 GUI การกำหนดเวลาการเคลื่อนที่

```
105  seekBar1.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
106      int pros;
107      @Override
108      public void onProgressChanged(SeekBar seekBar, int i, boolean b) {
109          pros = i;
110          delay = pros+"";
111          txTime.setText(delay);
112      }
113      @Override
114      public void onStartTrackingTouch(SeekBar seekBar) {
115      }
116      @Override
117      public void onStopTrackingTouch(SeekBar seekBar) {
118          txDelay.setText(delay);
119      }
120  });
```

ภาพประกอบที่ 3.37 โค้ดสร้างและกำหนดค่าให้กับคอมโพเนนท์ Seekbar และ textView
บรรทัดที่ 105 ประกาศการทำงานของ seekbar เพื่อรับค่าที่มีการทำงาน
บรรทัดที่ 111 เปลี่ยนค่าที่แสดงบน textView time ให้เป็นค่าการทำงานของ seekbar
บรรทัดที่ 118 เปลี่ยนค่าที่แสดงบน textView Delay เป็นค่าที่จะบันทึกบนฐานข้อมูล
ฟังก์ชันเรียกใช้ webservice

```
130  public void init_save_datatoDB_speed(){
131      String url = TagUtils.WEBSERVICEURI + "read_data_commands.php"; //edit
132      new FeedTask_set_speed().execute(url);
133  }
```

ภาพประกอบที่ 3.38 โค้ดฟังก์ชันเรียกใช้ webservice

Computer Science Department
Faculty of Informatics, Mahasarakham University

บรรทัดที่ 131 ประกาศ url เพื่อเก็บที่อยู่ webservice

ฟังก์ชันรับค่า json และเตรียมส่งข้อมูล

```
135    public class FeedTask_set_speed extends AsyncTask<String, Void, JSONObject> {
136        protected void onPreExecute() { super.onPreExecute(); }
139        protected JSONObject doInBackground(String... params) {
140            JSONObject result = null;
141            try {
142                String postJson = genPostJson_set_speed();
143                result = new WebServicesHelper( requestXml: null).postServiceJSON(params[0], postJson);
144                return result;
145            } catch (Exception e) {
146                return null;
147            }
148        }
```

ภาพประกอบที่ 3.39 โค้ดฟังก์ชันรับค่า json และเตรียมส่งข้อมูล

บรรทัดที่ 142 ประกาศสริงเพื่อเก็บค่าที่ต้องการส่ง

บรรทัดที่ 143 นำข้อมูลที่เตรียมเสร็จแล้วเก็บไว้ในตัวแปร result จากนั้นส่งค่ากลับที่บรรทัด 144

ฟังก์ชันสร้างข้อมูล json ที่ต้องการส่ง

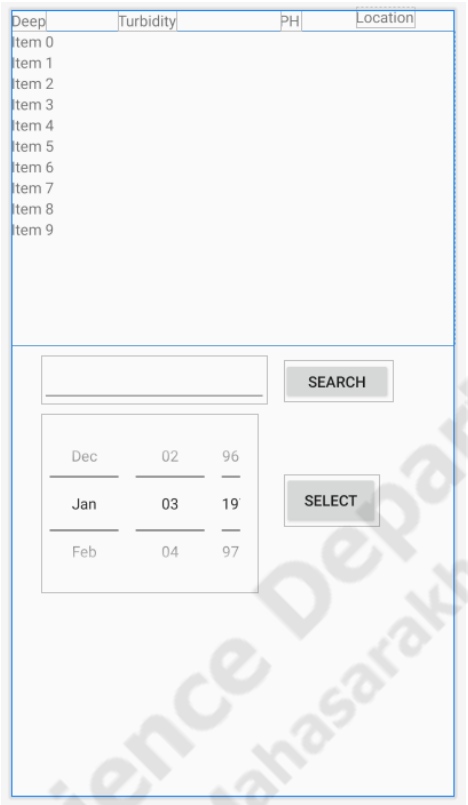
```
166    public String genPostJson_set_speed() {
167        String result = null;
168
169        try {
170            JSONObject jsonData = new JSONObject();
171            jsonData.put( name: "command",var_command);
172            jsonData.put( name: "process",var_process);
173            jsonData.put( name: "speed",delay);
174
175            result = jsonData.toString();
176        } catch (Exception e) {
177            result = null;
178        }
179        return result;
180    }
```

ภาพประกอบที่ 3.40 โค้ดฟังก์ชันสร้างข้อมูล json ที่ต้องการส่ง

บรรทัดที่ 170 ประกาศ json และสร้าง json ด้วย “command”, ”process”, ”speed” นำข้อมูล json มาเก็บไว้ที่ตัวแปร result และส่งกลับใน บรรทัดที่ 179

Computer Science Department
Faculty of Informatics, Mahasarakham University

หน้าต่างแสดงผล



ภาพประกอบที่ 3.41 หน้าต่างแสดงผล

หน้าจอแสดงข้อมูลจากฐานข้อมูล ประกอบด้วยความลึก ความขุ่น ค่า PH และ Location สามารถค้นหาสถานที่ที่ต้องการดูข้อมูล (ปุ่ม SEARCH) และสามารถเลือกดูตามวันที่ได้ (ปุ่ม SELECT)

```
57      etext = findViewById(R.id.editTextSearch);
58      button_searc = findViewById(R.id.button_search);
59      button_searc.setOnClickListener(new View.OnClickListener() {
60          @Override
61          public void onClick(View view) {
62              data_select="";
63              selec = etext.getText().toString();
64              init();
65          }
66      });
```

ภาพประกอบที่ 3.42 ฟังก์ชันแสดงสถานที่ด้วยตำแหน่งที่ค้นหา

ฟังก์ชันแสดงสถานที่ด้วยตำแหน่งที่ค้นหา

บรรทัดที่ 57 ประกาศ edit text สำหรับกรอกตำแหน่งที่ต้องการค้นหาข้อมูล

Computer Science Department
Faculty of Informatics, Mahasarakham University

บรรทัดที่ 58 ประกาศปุ่มค้นหา

บรรทัดที่ 59 ถึง 66 ฟังก์ชันการกำหนดค่าให้กับตัวแปร select จาก edit text และเรียกใช้ฟังก์ชัน init() เพื่อส่งค่าที่ต้องการค้นหา

```
67      datePicker = findViewById(R.id.datePicker1);
68      button_sl = findViewById(R.id.button_select);
69      button_sl.setOnClickListener(new View.OnClickListener() {
70          @Override
71          public void onClick(View view) {
72              data_select = datePicker.getYear()+"-"+ (datePicker.getMonth() + 1)+"-"+datePicker.getDayOfMonth();
73              selec = "";
74              Log.i( tag: "date",  msg: "" + data_select);
75              Log.i( tag: "select",  msg: "" + selec);
76              init();
77          }
78      });
```

ภาพประกอบที่ 3.43 ฟังก์ชันแสดงข้อมูลจากการเลือกวันที่

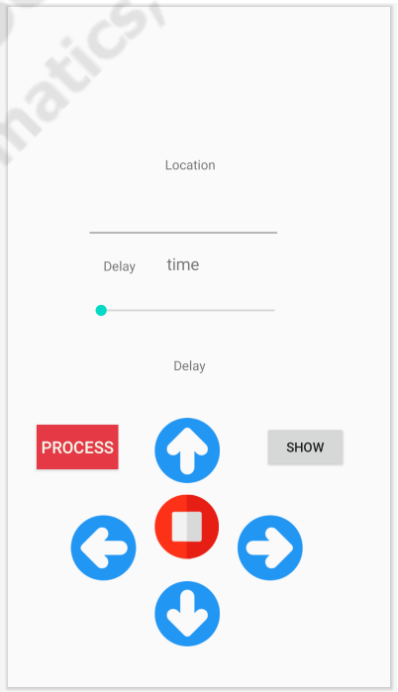
ฟังก์ชันแสดงข้อมูลจากการเลือกวันที่

บรรทัดที่ 67 ประกาศตัวแปรประเภท datepicker สำหรับเลือกวันที่

บรรทัดที่ 68 ประกาศตัวแปรประเภท button สำหรับส่งข้อมูลวันที่ที่ต้องการนำมาแสดง

บรรทัดที่ 69 ถึง 78 ฟังก์ชันการเลือกแสดงข้อมูลตามวันที่ที่เลือกปัจจุบัน และเรียกใช้ฟังก์ชัน init()

หน้าต่างควบคุม



ภาพประกอบที่ 3.44 หน้าต่างควบคุม

Computer Science Department
Faculty of Informatics, Mahasarakham University

จากภาพประกอบที่ 3.42 จะใช้ปุ่มลูกศรที่ชี้ไปในทิศทางต่างๆในการเคลื่อนที่ ปุ่มวงกลมที่มีสี่เหลี่ยมสีขาว คือ ปุ่มสั่งหยุดการเคลื่อนที่ ปุ่มที่เขียนว่า” PROCESS” คือปุ่มสั่งวัดค่าความลึก ปุ่มที่เขียนว่า “SHOW” คือปุ่มแสดงข้อมูล

```
121         editText = findViewById(R.id.editTextTex1);
122         button3.setOnClickListener(new View.OnClickListener() {
123             @Override
124             public void onClick(View view) {
125                 var_command = "1";
126                 var_process = "1";
127                 location = editText.getText().toString();
128                 init_save_datatoDB_speed();
129             }
130         });
```

ภาพประกอบที่ 3.45 ฟังก์ชันกำหนดสถานที่ที่อุปกรณ์ทำงาน

ฟังก์ชันกำหนดสถานที่ที่อุปกรณ์ทำงาน

บรรทัดที่ 121 ประกาศตัวแปรประเภท edit text เพื่อระบุตำแหน่งที่อุปกรณ์ทำงาน

บรรทัดที่ 122 ฟังก์ชันส่งคำสั่งให้อุปกรณ์ทำงาน ส่งตำแหน่งที่อุปกรณ์ทำงาน และเรียกใช้ฟังก์ชัน init_save_datatoDB_speed()

Computer Science Department
Faculty of Informatics, Mahasarakham University

3.10.4 พัฒนาในส่วนเว็บไซต์

รับข้อมูล json และบันทึกค่าลงฐานข้อมูล

```php
$data = json_decode(file_get_contents('php://input'), true);
$command = $data["command"];
$delay = $data["speed"];
$process = $data["process"];

$sql_data ="INSERT INTO `tb_command`(`id_command`, `cmd_command`, `process_command`, `delay_command`)
                    VALUES (NULL,'".$command."','".$process."','".$delay."')";

$mysqli  =  getmysqli();
if($result = $mysqli -> query($sql_data)) {
    $jsonre = $jsonre.'{
        "status"         : "1",
        "message"        : "บันทึกข้อมูลสำเร็จ"
    }';
}else{

    $jsonre = $jsonre.'{
        "status"         : "0",
        "message"        : "บันทึกข้อมูลไม่สำเร็จ"
    }';
}
print_r($jsonre);
echo $command,"THIS VALUE";
```

ภาพประกอบที่ 3.46 โค้ดรับข้อมูล json และบันทึกค่าลงฐานข้อมูล

ที่มา : [9] Ninenik Narkdee, “php javascript api mysql การสร้าง, ” 2 May 2019. [Online]. Available: https://bit.ly/3cBjkzR. [Accessed 5 August 2020].

บรรทัดที่ 7 สร้าง json และรับค่าข้อมูล ด้วยตัวแปร $command เพื่อรับข้อมูลคำสั่ง $delay เพื่อรับเวลาที่ใช้ในการหน่วง $process เพื่อรับสถานการณ์ทำงานของคำสั่ง

บรรทัดที่ 16 ตรวจสอบการบันทึกข้อมูล

บรรทัดที่ 19 ส่งข้อมูลตอบกลับถ้าการบันทึกข้อมูลสำเร็จจะแสดงข้อความ “บันทึกข้อมูลสำเร็จ”

บรรทัดที่ 25 ส่งข้อมูลตอบกลับถ้าการบันทึกข้อมูลไม่สำเร็จจะแสดงข้อความ “บันทึกข้อมูลไม่สำเร็จ”

การส่งคำสั่งไปยัง Raspbery pi3 ด้วย json

```php
$jsonre='{ "Command" : [';
$sql_data ="SELECT * FROM `tb_command` ORDER BY id_command DESC LIMIT 1 ";
$mysqli  =  getmysqli();
if($result = $mysqli -> query($sql_data)) {
        if( $result->num_rows > 0 ) {
                while($row = $result->fetch_assoc()) {
                    $jsonre = $jsonre.'{
                                        "id"     :  "'.$row["id_command"].'",
                                        "command"        : "'.$row["cmd_command"].'",
                                        "process"       : "'.$row["process_command"].'",
                                        "delay"      : "'.$row["delay_command"].'"
                                        },';
                }
        }else{
                $jsonre = $jsonre.',';
        }
}else{
    $jsonre=login_False();
}
$jsonre = substr($jsonre, 0, -1);
$jsonre =$jsonre."]}";
```

ภาพประกอบที่ 3.47 โค้ดการส่งคำสั่งไปยัง Raspberry pi 3 ด้วย json

บรรทัดที่ 6 สร้าง json ชื่อ “command” เพื่อเก็บข้อมูลคำสั่ง

บรรทัดที่ 7 select ขุดข้อมูลคำสั่งจากฐานข้อมูลมา 1 ข้อมูลล่าสุด

บรรทัดที่ 13 ถึง 15 การสร้างข้อมูล json และกำหนดค่าที่ต้องการส่ง

ฟังก์ชันการแสดงผลด้วยกราฟ

```html
<html>
  <head>
    <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <script type="text/javascript">
      google.charts.load('current', {'packages':['corechart']});
      google.charts.setOnLoadCallback(drawChart);

      function drawChart(rc_data) {
        var data = google.visualization.arrayToDataTable(rc_data);
        var options = {
          title: 'DATA FROM BOAT',
          curveType: 'function',
          legend: { position: 'bottom' }
        };

        var chart = new google.visualization.LineChart(document.getElementById('curve_chart'));

        chart.draw(data, options);
      }
    </script>
  </head>
  <body>
    <center>
      <input style="height: 20px; font-size:18px;" type="text" name = "name_search" id = "name_search" >
      <div id="curve_chart" style="width: 2000px; height: 800px;"></div>

    </center>
  </body>
</html>
```

ภาพประกอบที่ 3.48 ฟังก์ชันการแสดงผลด้วยกราฟ

บรรทัดที่ 3 include javascript สำหรับสร้างกราฟ

บรรทัดที่ 4 include jquery สำหรับร้องขอข้อมูลที่จะนำมาสร้างกราฟ

บรรทัดที่ 9 ประกาศฟังก์ชันการสร้างกราฟรับข้อมูลที่จะแสดงผ่านทาง พารามิเตอร์

บรรทัดที่ 10 สร้างตัวแปรเพื่อเก็บข้อรับข้อมูลที่จะทำไปแสดง

บรรทัดที่ 11 ถึง 15 ประกาศตัวแปร option สำหรับกำหนดหัวข้อของกราฟ

บรรทัดที่ 17 ประกาศตัวแปรเพื่อกำหนดรูปแบบของกราฟที่จะแสดง

บรรทัดที่ 19 คำสั่งแสดงผลกราฟด้วยข้อมูล data และ คุณสมบัติของกราฟ option ผ่านพารามิเตอร์

บรรทัดที่ 25 สร้าง field สำหรับค้นหาข้อมูลด้วยการกรอกชื่อสถานที่หรือ ตัวย่อของสถานที่ที่ต้องการแสดงผล

บรรทัดที่ 26 แสดงผลกราฟด้วยการเรียก id และกำหนดขนาดและความสูงของการแสดงผลของกราฟ

```php
<?php
    if(isset($_POST["get_data_graph"])){
        $get_name = $_POST["name_location"];
        echo select_data($get_name);

```

ภาพประกอบที่ 3.49 ฟังก์ชันการส่งข้อมูลเพื่อแสดงผลในรูปแบบกราฟ

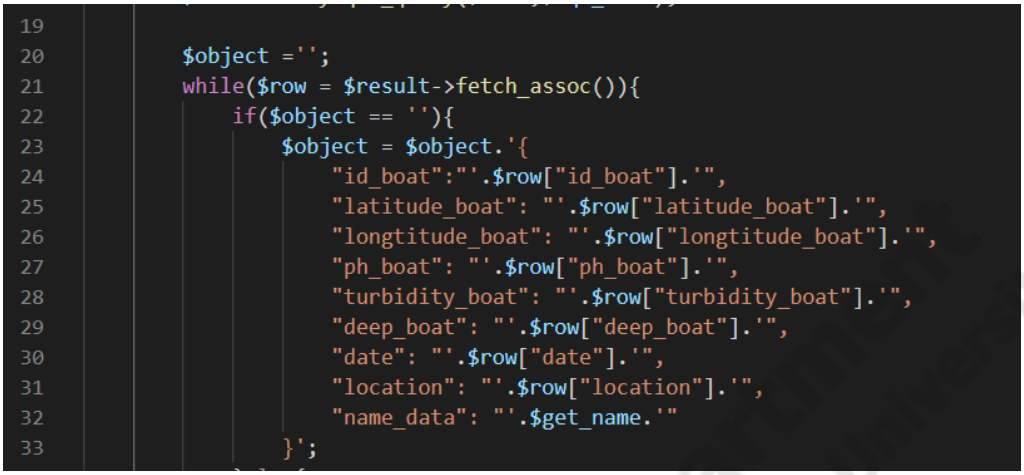
บรรทัดที่ 2 ถึง 4 เมื่อมีการส่งข้อมูลเข้ามาในรูปแบบ POST จะเรียกใช้ฟังก์ชัน select_data และส่ง get_name ผ่านพารามิเตอร์

```php
function select_data($get_name){
    require_once('../config/config.php');
    header('Content-Type: application/json');

    if(empty($get_name)){
        $sql_data ="SELECT * FROM `tb_boat` ";
    }
    else{
        $sql_data ="SELECT * FROM `tb_boat` WHERE location LIKE '%".$get_name."%'";
    }

    $result = mysqli_query($conn,$sql_data);

    $object ='';
```

ภาพประกอบที่ 3.50 select ข้อมูลทั้งหมดออกมาแสดง

บรรทัดที่ 11 ถึง 13 เลือกข้อมูลด้วยตัวแปร get_name ถ้า ตัวแปร get_name มีค่าเท่ากับว่าง จะ select ข้อมูลทั้งหมดออกมาแสดง

บรรทัดที่ 14 ถ้าตัวแปร get_name ไม่เป็นค่าว่างจะ select ข้อมูลสถานที่ที่ตรงกับตัวแปร หรือมีตัวอักษรที่อยู่ในตัวแปร get_name

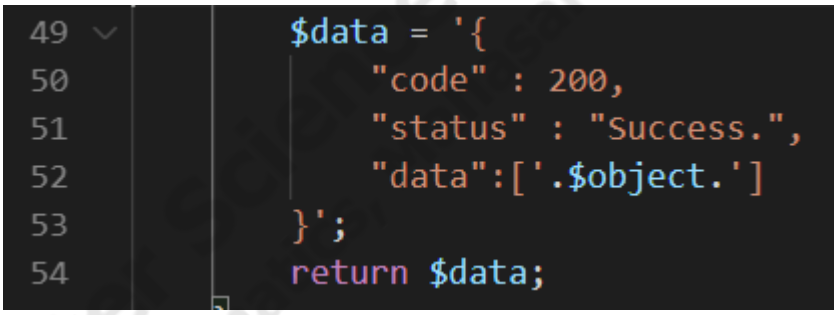


```
19
20          $object ='';
21          while($row = $result->fetch_assoc()){
22              if($object == ''){
23                  $object = $object.'{
24                      "id_boat":"'.$row["id_boat"].'",
25                      "latitude_boat": "'.$row["latitude_boat"].'",
26                      "longtitude_boat": "'.$row["longtitude_boat"].'",
27                      "ph_boat": "'.$row["ph_boat"].'",
28                      "turbidity_boat": "'.$row["turbidity_boat"].'",
29                      "deep_boat": "'.$row["deep_boat"].'",
30                      "date": "'.$row["date"].'",
31                      "location": "'.$row["location"].'",
32                      "name_data": "'.$get_name.'"
33                  }';
```

ภาพประกอบที่ 3.51 ประกาศตัวแปร json สำหรับเก็บข้อมูลที่จะส่งกลับไปแสดงผล

บรรทัดที่ 20 ประกาศตัวแปร json สำหรับเก็บข้อมูลที่จะส่งกลับไปแสดงผล

บรรทัดที่ 21 ถึง บรรทัดที่ 33 กำหนดค่าที่จะส่งในรูปแบบ json



```
49          $data = '{
50              "code" : 200,
51              "status" : "Success.",
52              "data":['.$object.']
53          }';
54          return $data;
```

ภาพประกอบที่ 3.52 return ตัวแปร data เพื่อกลับไปแสดงผล

ฟังก์ชันร้องขอข้อมูลด้วย Ajax

```
<script>
  get_data_from_db()
  setInterval(get_data_from_db,1000)
  function get_data_from_db(){
    var array_data = [['Date', 'Deep']];
    var eNameData = $("#name_search").val();
     $.ajax({
        type : 'POST',
        url : 'http://tradeoptionth.com/api/graph.php',
        data : {
            get_data_graph : 1,
            name_location : eNameData
        },
        success:function (response) {
            console.log('Success :'+ response)
            length_data = response.data.length
            for(var i in response.data){
                array_data.push([response.data[i].latitude_boat+","+response.data[i].longtitude_boat+"\\"+response.data[i].location,Number(response.data[i].deep_boat)
                console.log(response.data[i])
            }
            drawChart(array_data)
        },
        error:function (data,error) {
            console.log('Error : '+error)
        }
    });
  }
</script>
```

ภาพประกอบที่ 3.53 ฟังก์ชันร้องขอข้อมูลด้วย Ajax

บรรทัดที่ 49 ถึง 54 return ตัวแปร data เพื่อกลับไปแสดงผล ถ้าสถานะสำเร็จจะ return “Success”
บรรทัดที่ 34 ตั้งค่าให้มีการเรียกฟังก์ชันทุก ๆ 1 วินาที
บรรทัดที่ 36 ประกาศตัวแปรอาเรย์เพื่อเก็บข้อมูลที่จะไปแสดงผล
บรรทัดที่ 37 ประกาศตัวแปรรับข้อมูลค้นหา
บรรทัดที่ 38 ถึง 57 ประกาศตัวประ Ajax
บรรทัดที่ 40 ประกาศ url ที่จะร้องขอข้อมูล
บรรทัดที่ 41 ถึง 44 ประกาศตัวแปร data ที่จะส่งข้อมูลร้องขอข้อมูล
บรรทัดที่ 45 ถึง 51 จัดข้อมูลให้อยู่ในรูปแบบอาเรย์เพื่อนำไปแสดงผล
บรรทัดที่ 52 เรียกใช้ฟังก์ชัน drawChart และส่งข้อมูลที่จะแสดงผลผ่านพารามิเตอร์
บรรทัดที่ 54 ถึง 56 เมื่อเกิดข้อผิดพลาดจะแสดงผล “Error”

```php
<?php
require_once('../config/config.php');
header('Content-Type: application/json');

$data = json_decode(file_get_contents('php://input'), true);
$command = $data["command"];
$delay = $data["speed"];
$process = $data["process"];
$location = $data["location"];
```

ภาพประกอบที่ 3.54 ฟังก์ชันการส่งคำสั่งไปยังฐานข้อมูล

ฟังก์ชันการส่งคำสั่งไปยังฐานข้อมูล
บรรทัดที่ 5 ประกาศตัวแประรับข้อมูล json จากแอปพลิเคชัน
บรรทัดที่ 6 ประกาศตัวแปรเพื่อรับค่า “command”
บรรทัดที่ 7 ประกาศตัวแปรเพื่อรับค่า “speed”
บรรทัดที่ 8 ประกาศตัวแปรเพื่อรับค่า “process”

บรรทัดที่ 9 ประกาศตัวแปรเพื่อรับค่า “location”

```
10
11  $sql_data ="INSERT INTO `tb_commands`( `command`, `process`, `delay`,`location`)
12                                      VALUES ('".$command."','".$process."','".$delay."','".$location."')";
13
14  if($result = $conn -> query($sql_data)) {
15      $jsonre = $jsonre.'{
16          "status"          : "1",
17          "message"         : "บันทึกคำสั่ง"
18      }';
19  }else{
20
21      $jsonre = $jsonre.'{
22          "status"          : "0",
23          "message"         : "บันทึกข้อมูลไม่สำเร็จ"
24      }';
25  }
26  print_r($jsonre);
```

ภาพประกอบที่ 3.55 บันทึกข้อมูล

บรรทัดที่ 11 ส่งคำสั่งไปยังฐานข้อมูลด้วยคำสั่ง sql

บรรทัดที่ 14 ถึง 18 ถ้า บันทึกข้อมูลสำเร็จ return “บันทึกคำสั่ง” ไปยังแอปพลิเคชัน

บรรทัดที่ 19 ถึง 24 ถ้า บันทึกข้อมูลไม่สำเร็จ return “บันทึกข้อมูลไม่สำเร็จ” ไปยังแอปพลิเคชัน

Computer Science Department
Faculty of Informatics, Mahasarakham University