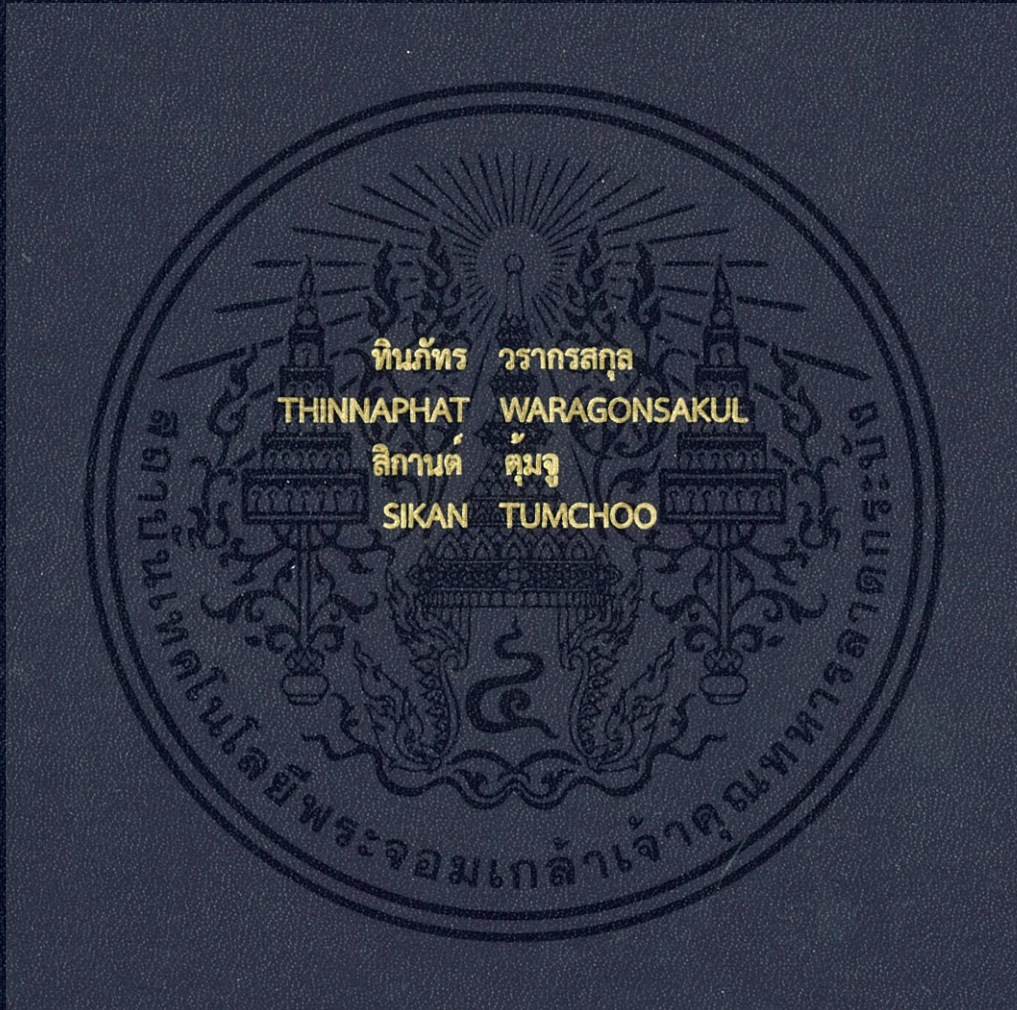


ระบบเฝ้าระวังและแจ้งเตือนผู้ป่วยลุกออกจากเตียง
Surveillance and Alarm System for a Patient Leaving off His/Her Bed



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมชีวการแพทย์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

ระบบเฝ้าระวังและแจ้งเตือนผู้ป่วยลุกออกจากเตียง
Surveillance and Alarm System for a Patient Leaving off His/Her Bed

โดย

ทินภัทร วรากรสกุล รหัส 56010484
สิกานต์ ตุ่มจุก รหัส 56011300



อาจารย์ที่ปรึกษา
ผศ.ดร. กิตติพล ชิตสกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมชีวการแพทย์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ ปีการศึกษา 2559
สาขาวิชา วิศวกรรมชีวการแพทย์
คณะ วิศวกรรมศาสตร์
เรื่อง สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ระบบเฝ้าระวังและแจ้งเตือนผู้ป่วยลุกออกจากเตียง
 Surveillance and Alarm System for a Patient Leaving off His/Her Bed
ผู้จัดทำ ทินภัทร วรากรสกุล รหัส 56010484
 สิกานต์ ตุ่มจุก รหัส 56011300

รายงานนี้ผ่านการตรวจโดยอาจารย์ที่ปรึกษาแล้ว




(ผศ.ดร. กิตติพล ชิตสกุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบเฝ้าระวังและแจ้งเตือนผู้ป่วยลุดออกจากเตียง
รายชื่อนักศึกษา	ทินภัทร วรากรสกุล รหัส 56010484
	สิกานต์ ตุ่มจู่ รหัส 56011300
ปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมชีวการแพทย์
ปีการศึกษา	2559
อาจารย์ที่ปรึกษาปริญญานิพนธ์	ผศ.ดร. กิติพล ชิตสกุล

บทคัดย่อ

ในปัจจุบันพบว่าแพทย์และพยาบาลไม่สามารถดูแลผู้ป่วยทุกคนได้อย่างทั่วถึงตลอดเวลา แม้แพทย์และพยาบาลจะมีความระมัดกุมในการดูแลแค่ไหน แต่ก็ยังมีปัญหาเกิดตามโรงพยาบาลต่างๆ แม้เพียงช่วงเวลาสั้นๆ เช่น ผู้ป่วยหนีออกจากโรงพยาบาลโดยที่ไม่มีใครรู้ตัว ทำให้เกิดปัญหาตามมา ภายหลัง ผู้ป่วยหลังผ่าตัดที่แพทย์สั่งห้ามลุดออกจากเตียงเพื่อป้องกันอาการบาดเจ็บจากบาดแผล ผ่าตัด เป็นต้น โครงการระบบเฝ้าระวังผู้ป่วยลุดหรือหนีออกจากเตียงจึงถูกพัฒนาขึ้นทั้งนี้ก็เพื่อตัวผู้ป่วยเองและเพื่อช่วยแพทย์หรือพยาบาลสามารถเฝ้าระวังผู้ป่วยได้ตลอดและได้รับการแจ้งเตือนได้ทันเวลา โครงการนี้ใช้ไมโครคอนโทรลเลอร์ประมวลผลข้อมูลจากเซ็นเซอร์วัดค่าแรงดันจากเบาะลมที่อยู่ใต้ผู้ป่วย จากนั้นข้อมูลที่ประมวลผลแล้วจะส่งผ่านตัวไมโครคอนโทรลเลอร์ไปยังเว็บเซิร์ฟเวอร์เพื่อเก็บข้อมูลไว้ในฐานข้อมูล ดังนั้นแพทย์หรือพยาบาลสามารถตรวจสอบข้อมูลการเฝ้าระวังผู้ป่วย และการแจ้งเตือนผ่านทางหน้าเว็บเพจ โดยเชื่อมต่อกับเครือข่ายอินเทอร์เน็ตสามารถดูข้อมูลเป็นกราฟเพื่อดูการขยับตัวของผู้ป่วย โครงการนี้สามารถนำไปพัฒนาต่อยอดเพื่อให้สามารถรองรับการใช้งานกับผู้ป่วยได้หลายราย แจ้งเตือนผ่านช่องทางการสื่อสารรูปแบบอื่นๆ และประยุกต์ใช้งานกับเซ็นเซอร์ที่ใช้ในทางการแพทย์อื่นๆ ได้มากขึ้น นอกจากนี้กรณีทีอินเทอร์เน็ตขาดการเชื่อมต่อก็สามารถเรียกดูข้อมูลเก่าๆ จากฐานข้อมูลได้เพื่อนำไปวิเคราะห์ผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title Surveillance and Alarm System for a Patient Leaving off His/Her Bed
Student Mr.Thinnaphat Waragonsakul Student ID 56010484
 Miss.Sikan Tumchoo Student ID 56011300
Degree Bachelor of Engineering
Program Biomedical Engineering
Year 2016
Thesis Advisor Asst. Prof. Dr. Kitiphol Chitsakul

ABSTRACT

In the present, due to number of patients in hospitals the caretakers cannot all times take care for each patient even they will be careful. In a short period of time some patients get off their beds or worse get out from the hospital. It may pose some problems to both the caretakers and the patients later. A postoperative patient by example, the doctor always proscribed getting off bed to prevent the injury from a surgery wound. A project therefor for automatic notification to the caretakers when a patient gets off the bed has been developed. Based on a pneumatic pad beneath the patient on the bed and a pressure sensor, the variation of pressure in the pad due to the patient's weight is detected and processed in the real-time by a NodeMCU™ connected to a web server for storing the data in its database via the WIFI. So doctors or nurses can verify the movement of the patients and also notification via web pages in the form of plotted graph. This project can be further developed to support the multiple patients. Notifications via other forms of communication channels and applications with many kinds of sensor, may be possible. In addition, if the Internet connection is lost, data will continue recording in the database and can be later retrieved.

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดี ทางคณะผู้จัดทำได้รับความกรุณาอย่างยิ่งจากบุคคลหลายท่านซึ่งอาจจะไม่สามารถนำมากล่าวได้ทั้งหมด ผู้มีพระคุณท่านแรกที่คุณศึกษาใคร่กราบขอบพระคุณคือ ผศ.ดร. กิติพล ชิตสกุล ซึ่งท่านเป็นอาจารย์ที่คอยให้คำปรึกษา ตรวจสอบ และแก้ไขข้อบกพร่องต่างๆ ด้วยความเอาใจใส่ในทุกขั้นตอน เพื่อให้โครงการนี้สำเร็จสมบูรณ์ที่สุด และให้ความรู้เพิ่มเติมในด้านต่างๆอีกมากมาย ขอขอบคุณผู้มีพระคุณท่านที่สอง คือ รศ.ดร. ชูชาติ ปิณฑวิรุจน์ ซึ่งอาจารย์ให้ความรู้ คำแนะนำ อีกทั้งความช่วยเหลือ ในการแก้ไขปัญหาต่างๆ ที่เกิดขึ้น แก่คณะผู้จัดทำตลอดระยะเวลาการศึกษา ขอขอบคุณผู้มีพระคุณท่านที่สาม นายภัทร นรเศรษฐ์โสภณ รุ่นที่นักศึกษาปริญญาโทที่ให้คำปรึกษาและแก้ไขข้อบกพร่องต่างๆของโครงการ ตลอดจนเทคนิคและความรู้ต่างๆเกี่ยวกับการเขียนโปรแกรม ต้องขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่เอื้อเฟื้อสถานที่ห้องแลป Physio และ HM-202 ในการทำโครงการ

ท้ายที่สุดนี้ คณะผู้จัดทำหวังเป็นอย่างยิ่งว่าโครงการเล่มนี้คงจะเป็นประโยชน์ต่อนักศึกษาหน่วยงานที่เกี่ยวข้อง ตลอดจนผู้ที่สนใจต่อไป

ทินภัทร
สิกันต์

วารากรสกุล
ตุ้มจุก

สารบัญ

	หน้า
บทคัดย่อ.....	I
Abstract.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญรูปภาพ.....	VII
สารบัญตาราง.....	X
บทที่ 1 บทนำ (Introduction)	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการศึกษา.....	1
1.3 สมมติฐานของการศึกษา	1
1.4 ขอบเขตการวิจัย	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีและความรู้พื้นฐานที่เกี่ยวข้อง (Theory and Related Basic Knowledge)	3
2.1 Internet of Things	3
2.1.1 A wireless sensor network (WSN).....	4
2.1.2 Access Technology	4
2.1.3 Gateway Sensor Nodes	5
2.1.4 แบ่งกลุ่ม Internet of Things.....	6
2.1.5 IPv6.....	7
2.1.6 การประยุกต์ใช้ Internet of Things.....	8
2.2 WIFI.....	9
2.2.1 ลำดับมาตรฐานของ Wireless หรือ Wifi.....	11
2.2.2 ประโยชน์ของ WIFI มีอะไรบ้าง.....	12
2.3 ESP8266 NodeMCU	12
2.3.1 ข้อดีของบอร์ด Arduino ESP8266	13
2.3.2 ความแตกต่างระหว่าง V0.9 vs V1.0.....	14
2.4 JavaScript	16
2.4.1 JavaScript ทำอะไรได้บ้าง	16

สารบัญ (ต่อ)

2.4.2	ข้อดีและข้อเสียของ JavaScript.....	17
2.4.3	ภาษา Script ที่ทำงานบนฝั่ง server (Server-side Script).....	17
2.5	Node.js.....	17
2.5.1	Concept การทำงาน.....	18
2.5.2	Node.js เอามาใช้ประโยชน์อย่างไร.....	20
2.5.3	Node.js ทำงานแบบ event driven.....	20
2.5.4	Socket.io.....	22
2.5.5	jQuery.....	23
2.6	HTML.....	24
2.6.1	ตัวอย่าง CODE ภาษา HTML.....	25
2.6.2	ผลลัพธ์หน้าจอสแสดงผล.....	25
2.7	CSS.....	26
2.7.1	ประโยชน์ของ CSS.....	27
2.8	Pressure Sensor.....	27
2.8.1	Integrated Silicon Pressure Sensor On-Chip Signal Conditioned.....	27
2.9	การดูแลผู้ป่วยในเตียง.....	30
2.10	อัมพฤกษ์อัมพาต.....	30
2.10.1	โรคอัมพาตโรคอัมพฤกษ์เกิดจากอะไร.....	31
2.10.2	ปัจจัยเสี่ยงต่อการเกิดอัมพาตอัมพฤกษ์มีอะไรบ้าง.....	31
2.10.3	โรคอัมพาตโรคอัมพฤกษ์มีอาการอย่างไร.....	32
2.10.4	การรักษา.....	33
2.10.5	ดูแลอย่างไรไม่เป็นอัมพฤกษ์อัมพาต.....	33
2.11	โรคกล้ามเนื้ออ่อนแรง.....	34
2.11.1	โรคกล้ามเนื้ออ่อนแรงเอ็มจี (Myasthenia gravis).....	34
2.11.2	โรคกล้ามเนื้ออ่อนแรงเอแอลเอส (Amyotrophic lateral sclerosis).....	38
บทที่ 3	วิธีดำเนินการวิจัย (RESEARCH METHODOLOGY).....	41
3.1	การทำงานของเซิร์ฟเวอร์ (Server).....	42
3.1.1	การทำ Real-Time web application.....	42

สารบัญ(ต่อ)

3.2 การรับส่งสัญญาณผ่าน NodeMCU.....	52
3.3 การทำงานส่วนของฮาร์ดแวร์ (Hardware).....	53
3.3.1 กล่องเซนเซอร์ (Sensor Box).....	53
3.3.2 หมอนลม (Pneumatic Pad).....	55
บทที่ 4 การทดลองและผลการทดลอง (EXPERIMENT AND RESULTS OF DATA).....	56
4.1 การทดลองในส่วนการรับส่งสัญญาณผ่าน NodeMCU	56
4.2 การทดลองในส่วนของเซิร์ฟเวอร์ (Server).....	58
4.3 การทดลองในส่วนของฮาร์ดแวร์ (Hardware).....	62
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ (CONCLUSION AND SUGGESTION).....	65
5.1 สรุปผลการวิจัย.....	65
5.1.1 ส่วนของฮาร์ดแวร์.....	65
5.1.2 ส่วนของตัวส่งข้อมูล.....	65
5.1.3 ส่วนของเว็บเซิร์ฟเวอร์.....	65
5.1.4 ส่วนของฮาร์ดแวร์.....	65
5.2 ข้อเสนอแนะ.....	65
5.3 แนวทางการพัฒนางานวิจัย.....	66
เอกสารอ้างอิง.....	67
ภาคผนวก.....	70
ภาคผนวก ก. คู่มือการใช้งานเบื้องต้น.....	71
ภาคผนวก ข. การติดตั้ง Eps8266 Nodemcu และ Node.js.....	74

สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 Kevin Ashton ผู้ที่ถูกยกย่องให้เป็นบิดาของ Internet of Things.....	3
รูปที่ 2.2 อธิบาย Wireless Sensor Network โดย purelink.ca.....	4
รูปที่ 2.3 แสดง WSN Nodes	5
รูปที่ 2.4 Diagram อธิบายการเชื่อมต่อ Gateway หลายๆตัวเข้ากับ local network	6
รูปที่ 2.5 ภาพอธิบายแต่ละ Network Layers ของ Internet of Things โดย IBM	7
รูปที่ 2.6 10 อันดับการประยุกต์ใช้ Internet of Things ยอดนิยม	8
รูปที่ 2.7 Apple iBook ที่มีตัวรับสัญญาณ Wireless มาให้ด้วย	10
รูปที่ 2.8 ตรารับรองว่าอุปกรณ์นั้นๆรองรับมาตรฐาน WiFi ประเภทต่างๆ.....	11
รูปที่ 2.9 มาตรฐาน WiFi ประเภทต่างๆ	12
รูปที่ 2.10 ESP8266 NodeMCU	12
รูปที่ 2.11 NodeMCU Devkit v0.9	14
รูปที่ 2.12 แสดงส่วนประกอบของ NodeMCU Devkit v0.9.....	14
รูปที่ 2.13 NodeMCU Devkit V1.0	15
รูปที่ 2.14 แสดงส่วนประกอบของ NodeMCU Devkit V1.0.....	15
รูปที่ 2.15 Event-loop วนอยู่ตลอดเวลาเพื่อดู Event ใหม่ที่เข้ามา.....	18
รูปที่ 2.16 Event-queue.....	19
รูปที่ 2.17 การใช้ Data.toString.....	19
รูปที่ 2.18 การพิมพ์ Do other things และ Hello World.....	20
รูปที่ 2.19 ทำงานแบบ event driven	21
รูปที่ 2.20 การใช้คำสั่ง <script>.....	24
รูปที่ 2.21 การเรียกใช้ jQuery จาก CDN	24
รูปที่ 2.22 ตัวอย่าง CODE ภาษา HTML.....	25
รูปที่ 2.23 ผลลัพธ์จากภาษา HTML.....	26
รูปที่ 2.24 MPXV5010 GC7U	27
รูปที่ 2.25 MPXV5050GC6U.....	28
รูปที่ 2.26 MPXV5004GP.....	29
รูปที่ 2.27 สาเหตุการเกิดโรคอัมพาตโรคอัมพฤกษ์	31
รูปที่ 2.28 ต่อมไทมัส.....	34

สารบัญรูปภาพ(ต่อ)

รูปที่ 2.29	โรคเอดส์	38
รูปที่ 3.1	บล็อกไดอะแกรมระบบการทำงาน	41
รูปที่ 3.2	การ check version	43
รูปที่ 3.3	ทดลอง run ไฟล์งาน	43
รูปที่ 3.4	แสดงผู้ใช้งานล็อกอินเข้าสู่ระบบ	47
รูปที่ 3.5	หน้าเพจล็อกอิน	48
รูปที่ 3.6	หน้าเพจ Register	48
รูปที่ 3.7	ส่วนการเก็บข้อมูลหรือ Database	49
รูปที่ 3.8	หน้าเพจหลักหรือหน้าเลือกห้องผู้ป่วย	49
รูปที่ 3.9	หน้า Update ข้อมูลของผู้ป่วย	50
รูปที่ 3.10	หน้าเพจแสดงผลกราฟ	50
รูปที่ 3.11	หน้าเพจเกี่ยวกับผู้จัดทำ	51
รูปที่ 3.12	หน้าเพจสำหรับติดต่อ	52
รูปที่ 3.13	NodeMCU	52
รูปที่ 3.14	การทำงานของหน้าเพจเลือกห้องผู้ป่วย	53
รูปที่ 3.15	ส่วนประกอบภายในกล่องเซนเซอร์	54
รูปที่ 3.16	ส่วนประกอบภายนอกกล่องเซนเซอร์	54
รูปที่ 3.17	หมอนลม	55
รูปที่ 3.18	การเชื่อมต่อระหว่างหมอนลมและกล่องเซนเซอร์	55
รูปที่ 4.1	แสดงการเชื่อมต่อ NodeMCU	56
รูปที่ 4.2	การส่งข้อมูลของ NodeMCU	57
รูปที่ 4.3	ข้อมูลของ NodeMCU	57
รูปที่ 4.4	หน้าเลือกห้องข้อมูลของผู้ป่วย	58
รูปที่ 4.5	ตัวอย่างหน้าเว็บเมื่อผู้ป่วยไม่อยู่บนเตียงจะมีสัญญาณเตือนและเสียงดังออกมา	59
รูปที่ 4.6	ตัวอย่างหน้าเลือกห้องผู้ป่วย	59
รูปที่ 4.7	หน้ากราฟตอนยังไม่ได้รับข้อมูลข้อมูลแต่จะแสดงข้อมูลย้อนหลัง	60
รูปที่ 4.8	Server รับข้อมูล	61
รูปที่ 4.9	กราฟแสดงผลเมื่อได้รับข้อมูลแบบเรียลไทม์	61

สารบัญรูปรภาพ(ต่อ)

รูปที่ 4.10 การวางอุปกรณ์.....	62
รูปที่ 4.11 การทดลองวัดค่าจากเซนเซอร์.....	62



สารบัญตาราง

หน้า

ตารางที่ 4.1 การทดลองเทียบค่าน้ำหนัก.....	63
ตารางที่ 4.2 สรุปค่าน้ำหนัก.....	64



บทที่ 1

บทนำ

(Introduction)

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบันนี้เทคโนโลยีมีความก้าวหน้าเป็นอย่างมากและมีการใช้งานอินเทอร์เน็ตกันอย่างแพร่หลาย ชีวิตประจำวันของเราล้วนผูกติดอยู่กับสิ่งเหล่านี้ ความรวดเร็วในการสื่อสารข้อมูลต่างๆ ทำให้อินเทอร์เน็ตถือเป็นปัจจัยสำคัญในปัจจุบันเลยก็ว่าได้ เพราะเช่นนี้จึงได้มีแนวคิดที่จะนำอินเทอร์เน็ตมาประยุกต์ใช้กับการทำโครงการ สิ่งที่กำลังเป็นที่สนใจและกำลังได้รับความนิยมมากในขณะนี้คือ เรื่อง IoT (The internet of thing) นั่นคือเทคโนโลยีอินเทอร์เน็ตที่เชื่อมต่อกับอุปกรณ์และเครื่องมือต่างๆ เข้าไว้ด้วยกัน โดยเครื่องมือต่างๆ จะสามารถเชื่อมโยงและสื่อสารกันได้โดยผ่านระบบอินเทอร์เน็ต

คณะผู้จัดทำจึงได้นำเทคโนโลยี IoT มาประยุกต์ใช้กับการส่งสัญญาณแจ้งเตือนสถานะผู้ป่วยว่าในขณะที่นั้น ผู้ป่วยยังอยู่บนเตียงหรือไม่ มีการเคลื่อนไหวหรือมีอาการผิดปกติหรือไม่ เพื่อสะดวกในการดูแลของพยาบาล ญาติและเพื่อเพิ่มความปลอดภัยในแก่ตัวผู้ป่วยด้วย

โดยหวังเป็นอย่างยิ่งว่าการแจ้งเตือนสถานะผู้ป่วย จะสามารถเพิ่มความปลอดภัยให้กับผู้ป่วยและเพื่อเพิ่มประสิทธิภาพ ความรวดเร็วในการเข้าถึงตัวผู้ป่วยเมื่อมีอันตรายได้ไม่มากก็น้อยและทำให้นักพัฒนาไทยเห็นความสำคัญในการพัฒนาอุปกรณ์การแพทย์ในประเทศไทยมากขึ้น

1.2 วัตถุประสงค์ของการศึกษา

1. เพื่อเฝ้าระวังผู้ป่วยลุกออกจากเตียงหรือหล่นจากเตียง
2. เพื่อเพิ่มประสิทธิภาพและความรวดเร็วในการเข้าถึงตัวผู้ป่วยเมื่อเกิดอันตราย
3. เพื่อนำความรู้ในหลายๆด้านมาประยุกต์รวมกัน
4. เพื่อประยุกต์ใช้งาน IoT ในทางการแพทย์
5. เพื่อเรียนรู้กระบวนการทำงานเป็นขั้นตอน การทำงานเป็นทีมและความสามารถในการแก้ปัญหาข้อขัดแย้งและอุปสรรคในการทำงานที่อาจเกิดขึ้น

1.3 สมมติฐานของการศึกษา

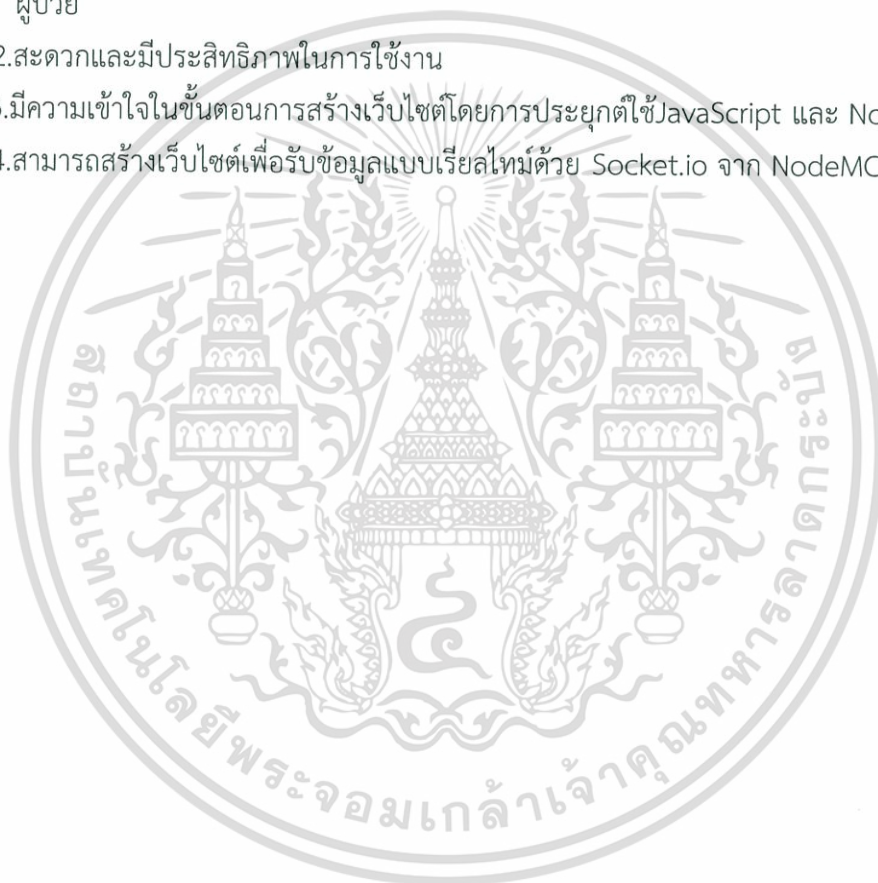
งานวิจัยนี้มุ่งเน้นการส่งสัญญาณแจ้งเตือนสถานะผู้ป่วย โดยให้ได้ผลลัพธ์ที่สามารถนำไปใช้ได้จริง สามารถช่วยเหลือผู้ป่วยได้ทันเวลาเมื่อเกิดปัญหาหรือได้รับอันตราย นอกจากนี้ยังสามารถบอกสถานะขณะนอนหลับหรือมีการเคลื่อนไหวได้ โดยดูผ่านทางกราฟบนหน้าเว็บเพจที่ออกแบบมาเพื่อเฝ้าระวังผู้ป่วยโดยเฉพาะและส่งข้อมูลผ่านระบบอินเทอร์เน็ต

1.4 ขอบเขตการวิจัย

1. ออกแบบหน้าเว็บเพจ
2. ส่วนแสดงผลในรูปแบบกราฟแบบเรียลไทม์
3. รับส่งสัญญาณผ่าน NodeMCU
4. เชื่อมต่อข้อมูลระหว่าง pressure sensor, NodeMCU และกราฟบนเว็บเพจ
5. ออกแบบแผ่นที่นอนลมและตัวเก็บ sensor

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เป็นประโยชน์ต่อการใช้งานในโรงพยาบาลเพราะสามารถแจ้งเตือนได้ในทันทีเมื่อเกิดปัญหากับผู้ป่วย
2. สะดวกและมีประสิทธิภาพในการใช้งาน
3. มีความเข้าใจในขั้นตอนการสร้างเว็บไซต์โดยการประยุกต์ใช้ JavaScript และ Node.js
4. สามารถสร้างเว็บไซต์เพื่อรับข้อมูลแบบเรียลไทม์ด้วย Socket.io จาก NodeMCU



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

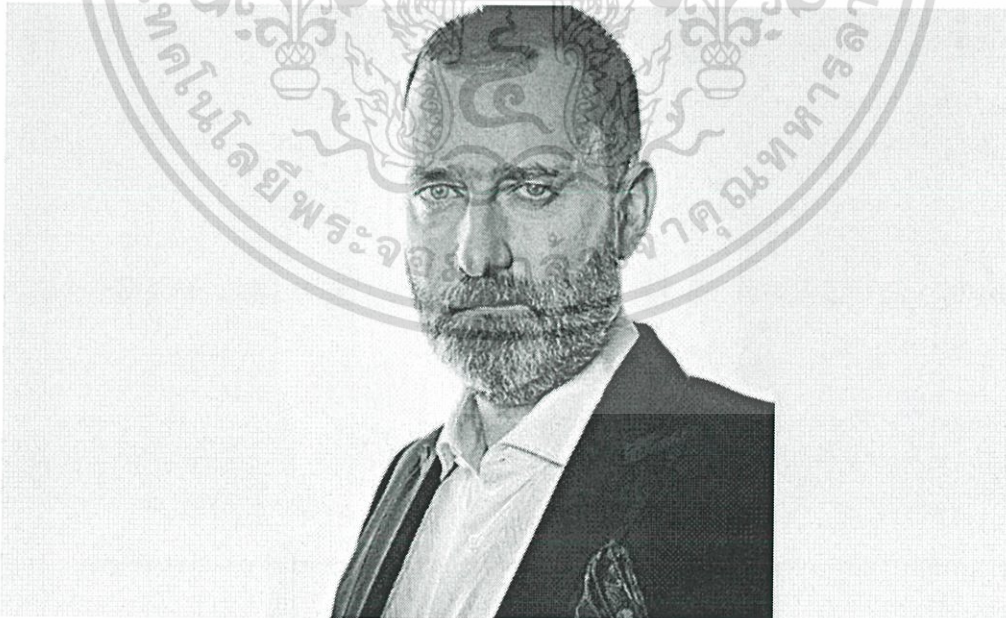
บทที่ 2

ทฤษฎีและความรู้พื้นฐานที่เกี่ยวข้อง

(Theory and Related Basic Knowledge)

2.1 Internet of Things

แนวคิด Internet of Things นั้นถูกคิดขึ้นโดย Kevin Ashton ในปี 1999 ซึ่งเขาเริ่มต้นโครงการ Auto-ID Center ที่มหาวิทยาลัย Massachusetts Institute of Technology หรือ MIT จากเทคโนโลยี RFID ที่จะทำให้เป็นมาตรฐานระดับโลกสำหรับ RFID Sensors ต่างๆที่จะเชื่อมต่อกันได้ ต่อมาในยุคหลังปี 2000 โลกมีอุปกรณ์อิเล็กทรอนิกส์ออกมาเป็นจำนวนมากและมีการใช้คำว่า Smart ซึ่งในที่นี้คือ smart device, smart grid, smart home, smart network, smart intelligent transportation ต่างๆเหล่านี้ล้วนมีโครงสร้างพื้นฐานที่สามารถเชื่อมต่อกับโลกอินเทอร์เน็ตได้ ซึ่งการเชื่อมต่อเหล่านี้เองก็เลยมาเป็นแนวคิดที่ว่าอุปกรณ์เหล่านี้ก็ย่อมสามารถสื่อสารกันได้ด้วยเช่นกัน โดยอาศัยตัว Sensor ในการสื่อสารถึงกัน นั้นแปลว่านอกจาก Smart devices ต่างๆจะเชื่อมต่ออินเทอร์เน็ตได้แล้วมันยังสามารถเชื่อมต่อไปยังอุปกรณ์ตัวอื่นได้ด้วยโดย Kevin นิยามมันไว้ตอนนั้นว่าเป็น “internet-like” หรือพูดง่ายก็คืออุปกรณ์อิเล็กทรอนิกส์สามารถสื่อสารพูดคุยกันเองได้ ซึ่งศัพท์คำว่า “Things” ก็แทนอุปกรณ์อิเล็กทรอนิกส์ที่กล่าวมาก่อนหน้านี้นั่นเอง

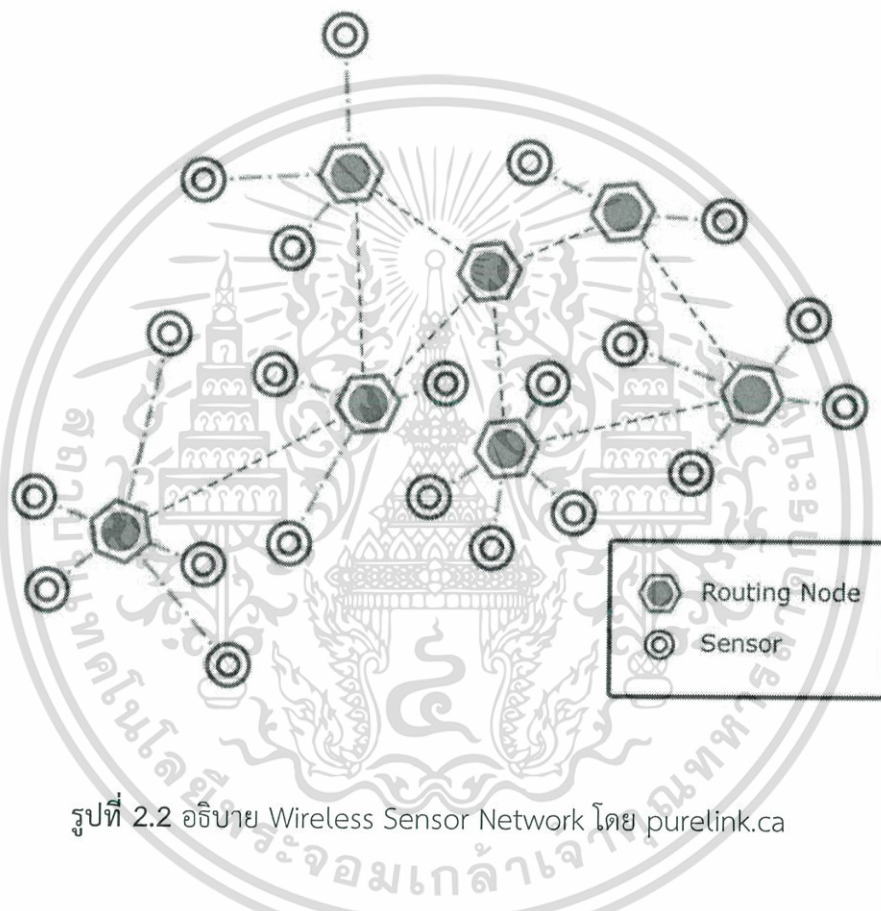


รูปที่ 2.1 Kevin Ashton ผู้ที่ถูกยกย่องให้เป็นบิดาของ Internet of Things

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1 A wireless sensor network (WSN)

ตัวแปรสำคัญสำหรับ Internet of Things ที่ใช้ในการสื่อสารนั้นไม่เพียงแต่ Internet network เพียงเท่านั้นแต่ยังมีตัวแปรอื่นเข้ามาเกี่ยวข้องอีกนั่นคือ Sensor node ต่างๆจำนวนมากที่ทำให้เกิด wireless sensor network (WSN) ให้กับอุปกรณ์ต่างๆสามารถเชื่อมต่อเข้ามาได้ ซึ่งเจ้า WSNs นี้เองสามารถตรวจจับปรากฏการณ์ต่างๆ (physical phenomena) ในเครือข่ายได้ด้วย ยกตัวอย่างเช่น แสง อุณหภูมิ ความดัน เป็นต้น เพื่อส่งค่าไปยังอุปกรณ์ในระบบให้ทำงานหรือสั่งงานอื่นๆต่อไป



รูปที่ 2.2 อธิบาย Wireless Sensor Network โดย purelink.ca

2.1.2 Access Technology

การพัฒนา Internet of Things นั้นนอกจากจะพัฒนาเทคโนโลยีในฝั่ง Hardware ได้แก่ processors, radios และ sensors ซึ่งจะถูกรวมเข้าด้วยกันเรียกว่า a single chip หรือ system on a chip (SoC) แล้วก็ยังพัฒนา WSN ไปพร้อมๆกันด้วย และเมื่อพูดถึงการเชื่อมต่อ ปัจจุบันได้มีการพัฒนาเทคโนโลยีสำหรับการเชื่อมต่อสำหรับ Internet of Things หรือ Access technology มีอยู่ 3 ตัวได้แก่

1. Bluetooth 4.0
2. IEEE 802.15.4e
3. WLAN IEEE 802.11™ (Wi-Fi)

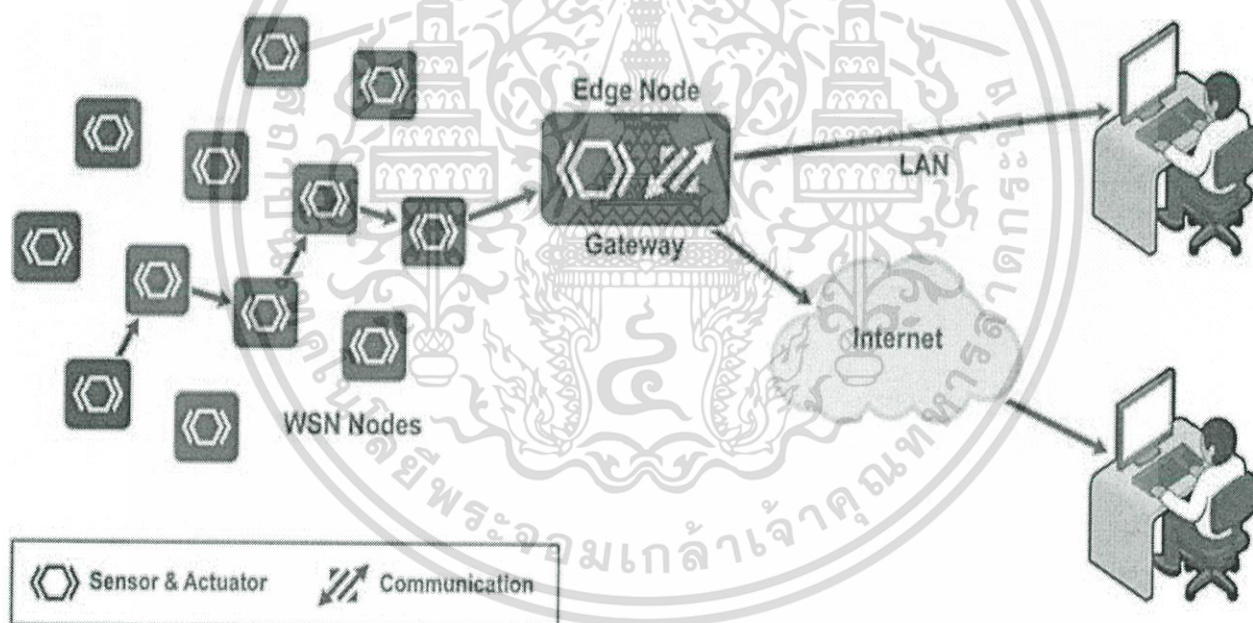
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในแต่ละ Access technologies นั้นมีการส่งข้อมูลที่แตกต่างกันดังนี้

	IEEE 802.15.4e	Bluetooth	WLAN IEEE 802.11
Frequency	868/915 MHz 2.4 GHz	2.4 GHz	2.4, 5.8 Ghz
Data rate	250 Kbps	723 Kbps	11 – 105 Mbps
Power	Very low	Low	High

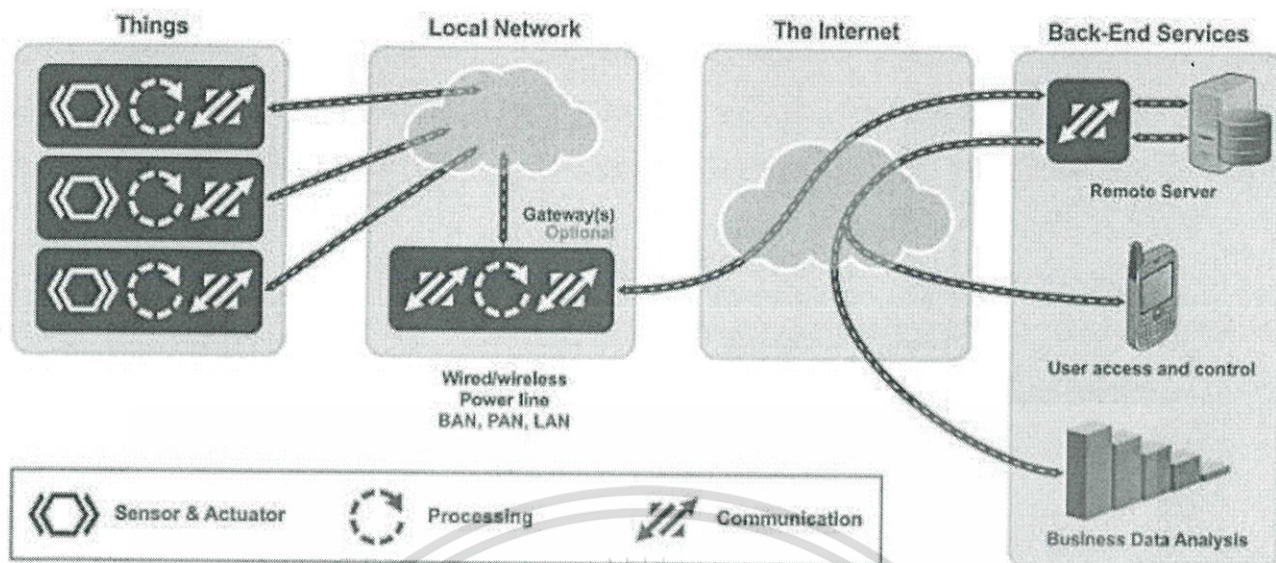
2.1.3 Gateway Sensor Nodes

เมื่อมีโครงข่าย Sensor nodes แล้วก็จำเป็นจะต้องมี Gateway Sensor Nodes เพื่อจะเชื่อมต่อไปยังโลกอินเทอร์เน็ตด้วย โดยตัว Gateway นี้จะทำหน้าที่เชื่อมต่อไปยังเครือข่าย Internet ให้อุปกรณ์ทั้งหมดในโครงข่าย Sensor nodes ทั้งหมดส่งข้อมูลเข้าสู่อินเทอร์เน็ตได้นั่นเอง และเจ้า Gateway ที่ว่านี้ก็อยู่ภายใต้ Local network ซึ่งจะมีการกำหนดกันต่อไปว่า Gateway ภายใต Local network ที่ว่านั้นจะให้เชื่อมต่อไปยัง Internet ได้ด้วยหรือไม่ถ้าไม่ได้อุปกรณ์ที่เชื่อมเข้ามาใน Gateway ก็อาจจะสื่อสารกันได้เฉพาะภายใน Local network เองได้เท่านั้น



รูปที่ 2.3 แสดง WSN Nodes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 Diagram อธิบายการเชื่อมต่อ Gateway หลายๆตัวเข้ากับ local network

2.1.4 แบ่งกลุ่ม Internet of Things

ปัจจุบันมีการแบ่งกลุ่ม Internet of Things ออกตามตลาดการใช้งานเป็น 2 กลุ่ม ได้แก่

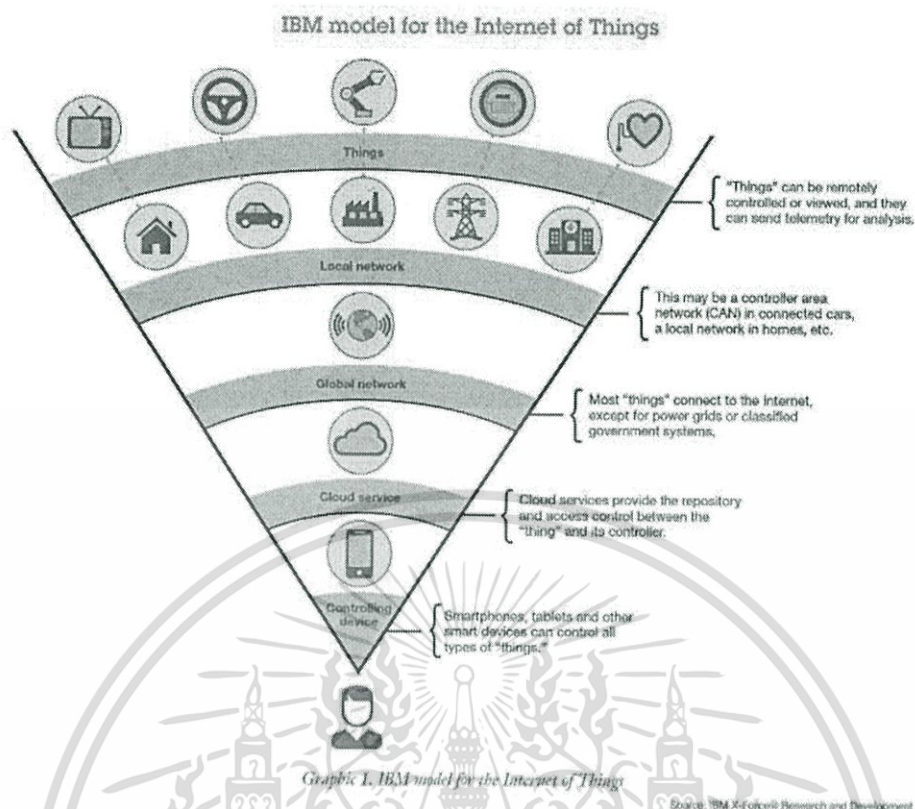
2.1.4.1 Industrial IoT

คือแบ่งจาก local network ที่มีหลายเทคโนโลยีที่แตกต่างกันในโครงข่าย Sensor nodes โดยตัวอุปกรณ์ IoT Device ในกลุ่มนี้จะเชื่อมต่อแบบ IP network เพื่อเข้าสู่อินเทอร์เน็ต

2.1.4.2 Commercial IoT

คือแบ่งจาก local communication ที่เป็น Bluetooth หรือ Ethernet (wired or wireless) โดยตัวอุปกรณ์ IoT Device ในกลุ่มนี้จะสื่อสารภายในกลุ่ม Sensor nodes เดียวกันเท่านั้นหรือเป็นแบบ local devices เพียงอย่างเดียวอาจไม่ได้เชื่อมสู่อินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ภาพอธิบายแต่ละ Network Layers ของ Internet of Things โดย IBM

2.1.5 IPv6

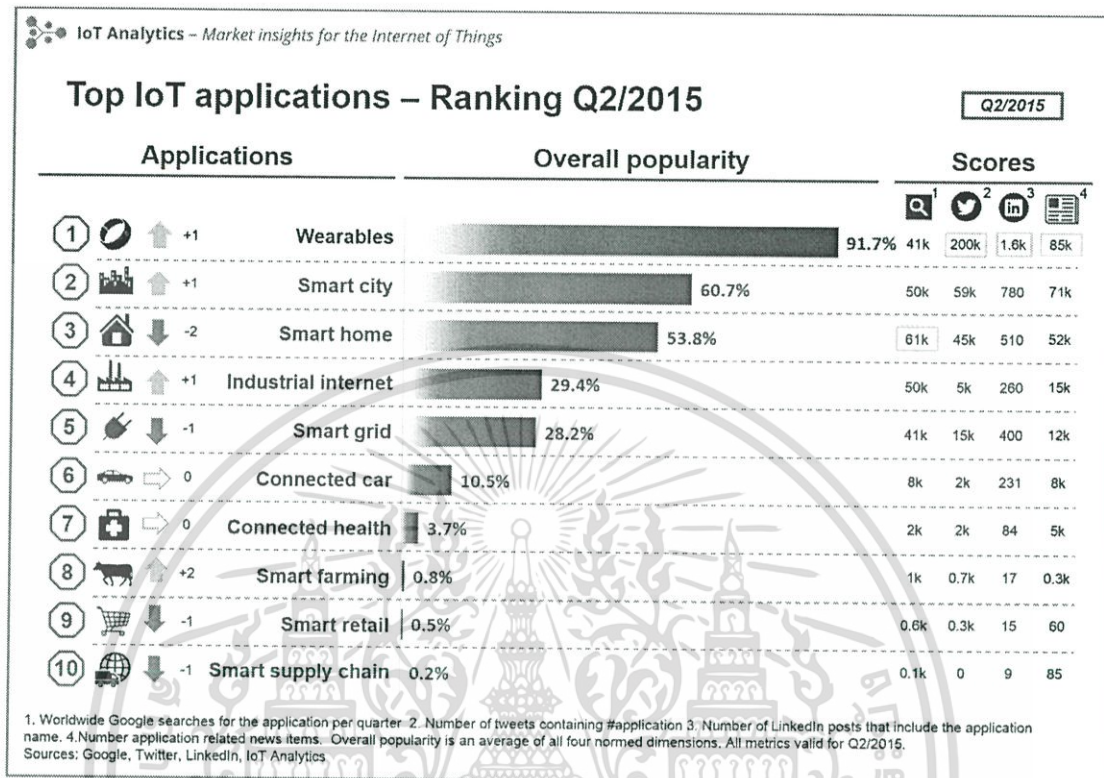
คือส่วนสำคัญของ Internet of Things ตัวอุปกรณ์ IoT devices ต่างๆนั้นจำเป็นจะต้องมีหมายเลขระบุเพื่อใช้ในการสื่อสารเปลี่ยนเสมือนที่อยู่บ้านของเรานั่นเอง และการที่จะทำให้อุปกรณ์เหล่านั้นที่มีอยู่เป็นจำนวนมาก(รวมถึงอนาคตที่จะผลิตกันออกมา) จำเป็นจะต้องใช้ IP Address version 6 หรือ IPv6 มากำกับเพื่อให้ได้หมายเลขที่ไม่ซ้ำกันและต้องใช้ได้ทั้ง

IoT network ที่เป็น LAN, PAN, และ BAN: Body Area Network หรือการสื่อสารของตัว Sensor กับร่างกายมนุษย์

Internet network (protocols) ที่เป็น IP, UDP, TCP, SSL, HTTP, HTTPS, และอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6 การประยุกต์ใช้ Internet of Things



รูปที่ 2.6 10 อันดับการประยุกต์ใช้ Internet of Things ยอดนิยม

อันดับที่ 1 Wearables คือ อุปกรณ์คอมพิวเตอร์ขนาดเล็กที่สามารถติดตั้งและใช้งานบนส่วนต่างๆ ของร่างกาย เพื่อความสะดวกในการใช้งานเพราะสามารถนำติดตัวไปได้ทุกที่ ปัจจุบันมีการพัฒนาออกมาเป็นรูปแบบต่างๆ เช่น นาฬิกา สายรัดข้อมือ และแว่นตา

อันดับที่ 2 Smart City หรือเมืองอัจฉริยะ หมายถึง เมืองที่มีการนำเทคโนโลยีมาปรับใช้เพื่อทำให้คุณภาพ ของประชากรดีขึ้น เช่น การจัดการพลังงานไฟฟ้า ระบบจัดการน้ำจัดการขยะ เป็นต้น

อันดับที่ 3 Smart Home หรือบ้านอัจฉริยะ หมายถึง การนำเทคโนโลยีมาควบคุม อุปกรณ์ต่างๆ ภายในหรือภายนอกบ้านได้ เพื่อให้เกิดความสะดวกสบาย และความปลอดภัย เช่น ประตูอัตโนมัติ เซนเซอร์ตรวจจับการเคลื่อนไหว การเปิดปิดไฟอัตโนมัติ เป็นต้น

อันดับที่ 4 Industrial internet เป็นการนำ IoT สำหรับอุตสาหกรรมและโรงงานการผลิต

อันดับที่ 5 Smart grid หรือโครงข่ายไฟฟ้าอัจฉริยะ เป็นการนำเทคโนโลยีสารสนเทศและการสื่อสารมาบริหารจัดการควบคุมการผลิต ส่ง และจ่ายพลังงานไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อันดับที่ 6 Connected car เป็นรถยนต์อัจฉริยะที่มีการติดตั้งระบบเครือข่ายอินเทอร์เน็ตแบบไร้สาย

อันดับที่ 7 Connected health เป็นแนวคิดการสร้างเครือข่ายเชื่อมโยงชุมชนเข้ากับระบบสุขภาพแบบครบวงจร

อันดับที่ 8 Smart farming หรือฟาร์มอัจฉริยะ คือ การนำเทคโนโลยีสมัยใหม่เข้ามาผสมผสานเข้ากับงานด้านเกษตร

อันดับที่ 9 Smart retail เป็นการนำเทคโนโลยีมาช่วยเพิ่มโอกาสในการดำเนินธุรกิจห้างร้าน

อันดับที่ 10 Smart Supply Chain คือ การจัดการในส่วนของกระบวนการที่เกิดขึ้นระหว่างผู้ผลิตกับผู้ขาย

2.2 WI-FI

การเชื่อมต่ออินเทอร์เน็ตในยุคปัจจุบันนี้ไม่จำเป็นต้องมีการเชื่อมต่อกับสายนำสัญญาณ UTP อีกต่อไปแล้วหลังมีการคิดค้นเทคโนโลยี Wireless Lan ขึ้นมาซึ่งการเริ่มต้นการทดสอบเริ่มต้นจากกลุ่มนักศึกษาจากมหาวิทยาลัยฮาวาย ซึ่งการทดสอบในครั้งนั้นเกิดขึ้นในปี 1970 ด้วยการทดสอบการต่อระบบเครือข่ายแบบสตาร์โดยใช้คอมพิวเตอร์ทั้งหมด 7 เครื่องด้วยกัน โดยแต่ละเครื่องจะอยู่บนเกาะฮาวายทั้งหมด 4 เกาะ เครื่องคอมพิวเตอร์ทั้งหมดนั้นจะเชื่อมต่อเป็นเครือข่ายไร้สายที่เรียกว่า ALOHAnet และส่งแพ็กเกจข้อมูลระหว่างคอมพิวเตอร์ในเครื่องแม่ข่ายและลูกข่าย โดยใช้เครื่องส่งและรับสัญญาณวิทยุอาร์เมเจอร์ (Amateur Radio) เป็นเสมือนฮับ (hub) คอยทำหน้าที่ในการส่งแพ็กเกจข้อมูลให้กับเครื่องลูกข่ายทุกเครื่องและรับข้อมูลจากเครื่องลูกข่ายแต่ละเครื่องด้วยการใช้ความถี่ที่ส่งและรับคนละความถี่กัน

ซึ่งการทดสอบครั้งนั้นถือเป็นครั้งแรกในการเปิดตัวเครือข่ายไร้สาย ต่อมาในปี 1991 สถาบัน IEEE (Institute of Electrical and Electronics Engineers) หรือเราจะเรียกว่าสถาบันวิศวกรรมทางด้านไฟฟ้าและอิเล็กทรอนิกส์ ซึ่งได้ออกมาจัดตั้งมาตรฐานของ Wireless LAN เป็นครั้งแรกซึ่งได้กำหนดมาตรฐาน IEEE 802.11 ขึ้นและในปีเดียวกันนั้นได้รองรับการพัฒนาเชิงพาณิชย์

ในยุคเริ่มต้นนั้นอุปกรณ์ที่ใช้งานในระบบ Wireless นั้นยังมีราคาแพงอยู่สถานที่ที่ใช้จะเป็นองค์กร โรงพยาบาลและมหาวิทยาลัยเท่านั้นจนกระทั่งในปี 1999 บริษัทแอปเปิ้ลได้ผลิต iBook ซึ่งเป็นเน็ตบุ๊กที่มีการติดตั้งตัวรับสัญญาณ Wireless มาด้วย ซึ่งเป็นครั้งแรกที่มีการผลิตมาใช้กับลูกค้าในระดับผู้ใช้งานทั่วไป อุปกรณ์ชิ้นนี้ถือว่าเป็นอุปกรณ์ชิ้นแรกที่ปฏิวัติวงการ Wireless เพราะหลังจากนั้นก็ยังมีผู้ผลิตมากมายที่ทำการพัฒนาอุปกรณ์รับสัญญาณ Wireless ทำให้ราคาของอุปกรณ์ถูกลง และมีผู้ใช้งานมากขึ้นตามไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 Apple iBook เครื่องคอมพิวเตอร์โน้ตบุ๊กในยุคแรกๆ
ที่มีตัวรับสัญญาณ Wireless มาให้ด้วย

แต่เมื่อมีผู้ใช้งานมากขึ้น ผู้ผลิตแต่ละค่ายก็พยายามพัฒนาอุปกรณ์ของตนเองทำให้อุปกรณ์ Wireless ไม่มีทิศทางที่แน่นอนเลยทำให้เกิดองค์กรที่จะมากำหนดมาตรฐานของอุปกรณ์ Wireless ขึ้นมาโดยให้ชื่อองค์กรนี้ว่า Wi-Fi organization เป็นสถาบันที่ตั้งขึ้นมาจากความร่วมมือระหว่างบริษัทเอกชนที่ทำธุรกิจด้านการสื่อสารเดิมชื่อว่า WECA (Wireless Ethernet Compatibility Alliance) ซึ่งต่อมาได้เปลี่ยนชื่อเป็น Wi-Fi organization ในปี 2000 ซึ่งบริษัทที่ทำการก่อตั้งองค์กรนี้ครั้งแรกก็คือ บริษัท 3com, Aironet ซึ่งในปัจจุบันก็คือ Cisco, Harris Semiconductor ปัจจุบันเปลี่ยนชื่อเป็น Intersil, Lucent ปัจจุบันเปลี่ยนชื่อเป็น Aere, Nokia และ Symbol Technology ซึ่งในปัจจุบันนี้สมาชิกอยู่กว่า 320 บริษัทแล้ว จะเรียกได้ว่าองค์กรนี้เป็นหน่วยงานที่ควบคุมการใช้เครื่องหมายการค้า Wi-Fi Certificated ออกมาใช้กับสมาชิกเพื่อแสดงให้ลูกค้าเห็นว่า เป็นอุปกรณ์ที่สามารถทำงานร่วมกันได้อย่างไม่มีปัญหาผู้ผลิตแต่ละรายที่เป็นสมาชิกจะได้โลโก้กันไปใส่ไว้ที่ผลิตภัณฑ์ของตัวเอง เนื่องจากการตั้งค่าต่างๆของอุปกรณ์ Wireless มีสเปคที่ต้องการ การเชื่อมต่อกันระหว่างอุปกรณ์จำเป็นต้องมีรหัสความปลอดภัย จึงมีความจำเป็นที่ต้องใช้คุณสมบัติตรงส่วนนี้ร่วมกันระหว่างอุปกรณ์กระจายสัญญาณและอุปกรณ์รับสัญญาณ ซึ่งอุปกรณ์ที่มีสัญลักษณ์ Wi-Fi Certificated จะสามารถเชื่อมต่อกันได้แบบไม่มีปัญหาซึ่งหลายคนสงสัยว่า Wireless และ Wi-Fi แตกต่างกันอย่างไรร จากข้อมูลจะเห็นได้ว่า ทั้ง Wireless และ Wi-Fi นั้นมีความหมายเหมือนกันเพียงแต่ว่าผู้ใช้งานจะเรียก Wi-Fi กันมากกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ตรารับรองว่าอุปกรณ์นั้นๆรองรับมาตรฐาน Wi-Fi ประเภทต่างๆ

การสื่อสารไร้สายทุกรูปแบบมีพื้นฐานการเชื่อมต่อจากคลื่นสนามแม่เหล็กไฟฟ้าทั้งสิ้น โดยคลื่นเหล่านี้จะมีตัวแปรที่ผูกพันกันอยู่คือความถี่กับความยาวคลื่น อาทิเช่น คลื่นวิทยุที่มีความถี่ต่ำจะมีความยาวคลื่นมาก โดยคลื่นความถี่ที่ใช้อยู่ในระบบ Wireless ที่มีอยู่ในตลาดคือ 2.4 GHz กับ 5 GHz ขึ้นอยู่กับมาตรฐาน IEEE ที่ใช้ ซึ่ง IEEE 802.b/g จะใช้ความถี่ที่ 2.4 GHz และมาตรฐาน IEEE 802.11a จะใช้ความถี่ที่ 5 GHz

2.2.1 ลำดับมาตรฐานของ Wireless หรือ Wi-Fi

- ปี 1997 มีการประกาศใช้มาตรฐานตัวแรกคือ IEEE 802.11
- ปี 1999 มาตรฐาน IEEE 802.11a และ IEEE 802.11b ประกาศใช้งานอย่างเป็นทางการ
- ปี 2000 WECA ใช้ชื่อ Wi-Fi แทนคำว่า Wireless Fidelity สำหรับเรียกอุปกรณ์ต่างๆที่รองรับมาตรฐาน IEEE 802.11b และ IEEE 802.11g
- ปี 2002 IEEE กำหนดขอบเขตมาตรฐาน IEEE 802.11 ให้มีตั้งแต่ 802.11a ถึง 802.11i
- ปี 2003 มาตรฐาน IEEE 802.11g ประกาศใช้งานอย่างเป็นทางการ
- ปี 2006 มีการเปิดตัวอุปกรณ์ ที่มีมาตรฐาน 802.11n (Pre-N) แต่ยังไม่ได้รับการรับรองจาก IEEE
- ปี 2007 มีการเปิดตัวอุปกรณ์ Wireless ที่ใช้มาตรฐาน 802.11n
- ปี 2009 มีการประกาศมาตรฐาน IEEE 802.11n อย่างเป็นทางการ

802.11-Standard	Standard seit	Frequenzb. (GHz)	Bandbreite (MHz)	Modulation	Datenrate (Mbit/s)
802.11	1997	2,4 GHz	20 MHz	DSSS/FHSS	2 Mbit/s
802.11a	1999	5 GHz	20 MHz	OFDM	54 Mbit/s
802.11ac	2013	5 GHz	40/80/160	OFDM	6,93 Gbit/s
802.11ad	2012	60 GHz	2160	SC-OFDM	6,76 Gbit/s
802.11b	1999	2,4 GHz	20	DSSS	11 Mbit/s
802.11g	2003	2,4 GHz	20	DSSS/OFDM	54 Mbit/s
802.11n	2009	2,4/5 GHz	20/40	OFDM	600 Mbit/s

DSSS, direct sequence spread spectrum
 FHSS, frequency hopping spread spectrum
 OFDM, orthogonal frequency division multiplex
 SC-OFDM, single carrier orthogonal frequency division multiplex

รูปที่ 2.9 มาตรฐาน Wi-Fi ประเภทต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ประโยชน์ของ WI-FI มีอะไรบ้าง

การเชื่อมต่อเครือข่ายไร้สายหรือ Wireless LAN นั้นเป็นการเชื่อมต่ออุปกรณ์ต่างๆให้สามารถใช้งานอินเทอร์เน็ตและแลกเปลี่ยนข้อมูลซึ่งกันและกันมาก โดยประโยชน์ของ W-I-FI นั้นมีอยู่มากมายอาทิเช่น

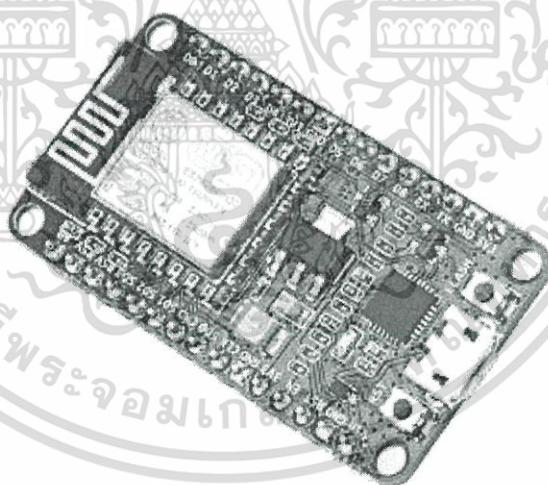
2.2.2.1 ช่วยลดค่าใช้จ่ายในการเดินระบบเครือข่ายซึ่งปกติแล้วการเชื่อมโยงเครือข่ายนั้นจำเป็นต้องใช้สายนำสัญญาณในการเชื่อมโยงเครือข่าย และต้องเสียค่าใช้จ่ายในการเดินสายสัญญาณ แต่สำหรับระบบ WI-FI ไม่จำเป็นเพราะระบบ WI-FI จะส่งคลื่นวิทยุผ่านอากาศไปยังเครื่องรับ

2.2.2.2 มีความยืดหยุ่นในการใช้งานเพราะการใช้งาน WI-FI นั้นไม่จำเป็นต้องอยู่กับที่ สามารถเคลื่อนย้ายไปไหนก็ได้ภายในรัศมีของการกระจายสัญญาณ

2.2.2.3 ใช้มาตรฐาน IEEE 802 ซึ่งเป็นมาตรฐานที่ยอมรับกันทั่วไปพร้อมกันนั้นอุปกรณ์ต่างๆ ที่เกี่ยวกับ WI-FI ก็มีราคาถูกลงและมีให้เลือกซื้อหาหลายยี่ห้อ

2.2.2.4 ช่วยส่งเสริมธุรกิจและธุรกรรมทางการเงิน อาทิเช่น การซื้อขายผ่านอินเทอร์เน็ต การทำธุรกรรมผ่านอินเทอร์เน็ต หรือแม้กระทั่งเป็นจุดเด่นของการดำเนินธุรกิจด้านบริการได้อีกด้วย

2.3 ESP8266 NodeMCU



รูปที่ 2.10 ESP8266 NodeMCU

NodeMCU คือ แพลตฟอร์มหนึ่งที่ใช้ช่วยในการสร้างโปรเจค Internet of Things (IoT) ที่ประกอบไปด้วย Development Kit (ตัวบอร์ด) และ Firmware (Software บนบอร์ด) ที่เป็น open source สามารถเขียนโปรแกรมด้วยภาษา Lau ได้ ทำให้ใช้งานได้ง่ายขึ้น มาพร้อมกับโมดูล Wi-Fi (ESP8266) ซึ่งเป็นหัวใจสำคัญในการใช้เชื่อมต่อกับอินเทอร์เน็ตนั่นเอง ตัวโมดูลESP8266นั้นมีอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยกันหลายรุ่นมาก ตั้งแต่เวอร์ชันแรกที่เป็น ESP-01 ไล่ไปเรื่อยๆจนปัจจุบันมีถึง ESP-12 แล้ว และที่ฝังอยู่ใน NodeMCU version แรกนั้นก็จะเป็น ESP-12 แต่ใน version2 นั้นจะใช้เป็น ESP-12E แทน ซึ่งการใช้งานโดยรวมก็ไม่แตกต่างกันมากนัก NodeMCU นั้นมีลักษณะคล้ายกับ Arduino ตรงที่มีพอร์ต Input Output ติดตั้งมาในตัว สามารถเขียนโปรแกรมคอนโทรลอุปกรณ์ I/O ได้โดยไม่ต้องผ่านอุปกรณ์อื่นๆ และเมื่อไม่นานมานี้ก็มีนักพัฒนาที่สามารถทำให้ Arduino IDE ใช้งานร่วมกับ Node MCU ได้ จึงทำให้ใช้ภาษา C/C++ ในการเขียนโปรแกรมได้ ทำให้เราสามารถใช้งานมันได้หลากหลายมากยิ่งขึ้น NodeMCU ตัวนี้สามารถทำอะไรได้หลายอย่างมากโดยเฉพาะเรื่องที่เกี่ยวข้องกับ IoT ไม่ว่าจะเป็นการทำ Web Server ขนาดเล็ก การควบคุมการเปิดปิดไฟผ่าน Wi-Fi และอื่นๆ อีกมากมาย

2.3.1 ข้อดีของบอร์ด Arduino ESP8266

2.3.1.1 เป็นแบบ Open Source Project มี Source code ให้ได้เรียนรู้อยู่บน GitHub ตามลิงค์ <https://github.com/esp8266/Arduino>

2.3.1.2 สามารถกด upload sketch ได้ เชื่อมต่อบอร์ด USB กับคอมพิวเตอร์ใช้งานง่าย ขนาดของบอร์ดต่อลง Protoboard ได้

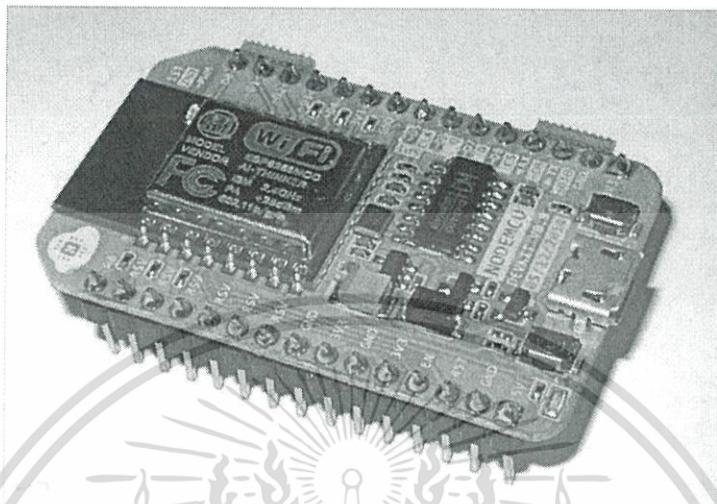
2.3.1.3 ซิปภายใน ESP8266 มี CPU ขนาด 32 bit แตกต่างจาก Arduino ที่เป็น CPU 8 bit

2.3.1.4 ถึงแม้ว่า I/O จะไม่มากเท่าของ Arduino แต่เราสามารถเขียนโปรแกรมลงบนขา GPIO ได้ทุกขาพอกัน เป็นข้อดีที่เพิ่มมาจากความต้องการใช้ WI-FI เชื่อมต่อเมื่อต้องการเล่น Arduino ทำให้ต้องซื้อ Module Wi-Fi เพิ่ม นั่นคือ NodeMCU (ESP8266) มีต้นทุนต่ำกว่ามาก

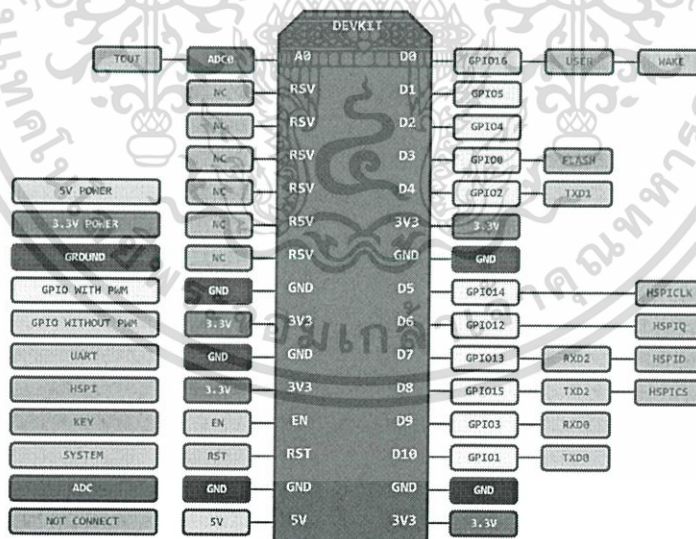
2.3.1.5 มีอุปกรณ์หลายอย่างที่ใช้งานที่แรงดัน +3.3 V เป็นส่วนใหญ่ ดังนั้นเราสามารถนำ NodeMCU (ESP8266) มาใช้เชื่อมต่อได้โดยตรง

2.3.2 ความแตกต่างระหว่าง V0.9 vs V1.0

2.3.2.1 NodeMCU Devkit V0.9



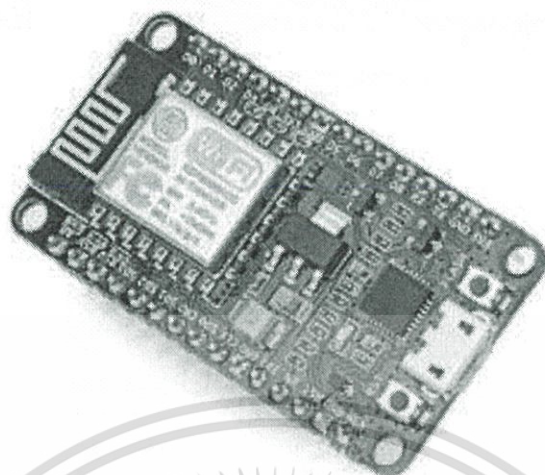
รูปที่ 2.11 NodeMCU Devkit V0.9



รูปที่ 2.12 แสดงส่วนประกอบของ NodeMCU Devkit V0.9

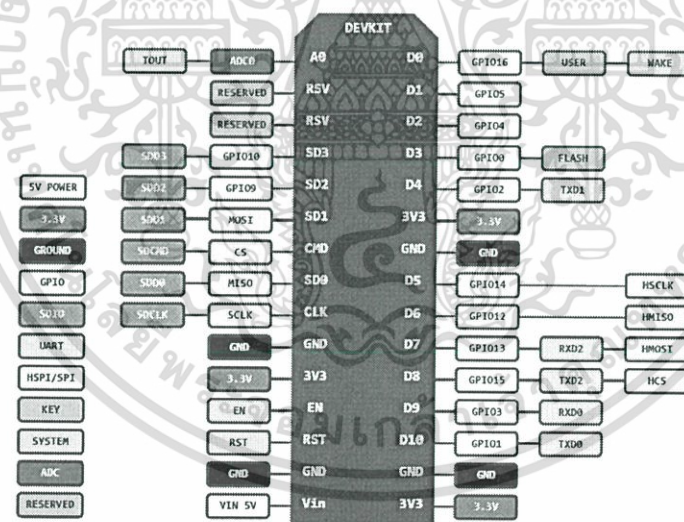
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.2 NodeMCU Devkit V1.0



รูปที่ 2.13 NodeMCU Devkit V1.0

PIN DEFINITION



D8(GPIO16) can only be used as gpio read/write, no interrupt supported, no pwm/i2c/iw supported.

รูปที่ 2.14 แสดงส่วนประกอบของ NodeMCU Devkit V1.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. NodeMCU V1.0 มีลักษณะยาวขึ้นแต่แคบลง สามารถเสียบลงบน protoboard ได้
- 2.ชิพแปลง USB2Serial ของ V0.9 เป็น CH340 แต่ของV1.0 เป็น CP2102 ในเวอร์ชันแรก ไดรเวอร์จะมีปัญหา OS ค่อนข้างมาก แต่ใน CP2102 จะใช้บน MacOS และมีปัญหาน้อยกว่า
- 3.ชิพ WI-FI ของ V0.9 เป็น ESP-12 แต่ใน V1.0 เป็น ESP-12E ทำให้มีจำนวนพินที่มากขึ้น และ พื้นที่ Flash มากขึ้น
 - ใช้ชิพ Flash ความจุ 32Mbits (4MBytes)
- 4.มีปุ่ม RST (รีเซ็ตการทำงาน) และ ปุ่ม Flash (สำหรับโปรแกรมเฟิร์มแวร์ใหม่)
- 5.มีขา AO รับอินพุตแรงดันแบบอนาล็อกสำหรับวงจร ADC (ขนาด 10 บิต) อ่านค่า 0V ถึง 1V

2.4 JavaScript

JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต ที่กำลังได้รับความนิยมอย่างสูง JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปที่ละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจกต์โอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server)

2.4.1 JavaScript ทำอะไรได้บ้าง

- 2.4.1.1 JavaScript ทำให้สามารถใช้เขียนโปรแกรมแบบง่ายๆได้ โดยไม่ต้องพึ่งภาษาอื่น
- 2.4.1.2 JavaScript มีคำสั่งที่ตอบสนองกับผู้ใช้ เช่นเมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็สามารส่งให้เปิดหน้าต่างใหม่ได้ ทำให้เว็บไซต์ของเรามีปฏิสัมพันธ์กับผู้ใช้มากขึ้น นี่คือข้อดีของ JavaScript เลยก็ว่าได้ที่ทำให้เว็บไซต์ต่างๆทั้งหลายเช่น Google Map ต่างหันมาใช้
- 2.4.1.3 JavaScript สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ นั่นคือสามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้ หรือหน้าแสดงเนื้อหาสามารถซ่อนหรือแสดงเนื้อหาได้แบบง่ายๆนั่นเอง
- 2.4.1.4 JavaScript สามารถใช้ตรวจสอบข้อมูลได้ สังเกตว่าเมื่อเรากรอกข้อมูลบางเว็บไซต์ เช่น Email เมื่อเรากรอกข้อมูลผิดจะมีหน้าต่างฟ้องขึ้นมาว่าเรากรอกผิด หรือลืมกรอกอะไรบางอย่าง เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.5 JavaScript สามารถใช้ในการตรวจสอบผู้ใช้ได้เช่น ตรวจสอบว่าผู้ใช้ใช้ web browser อะไร

2.4.1.6 JavaScript สร้าง Cookies (เก็บข้อมูลของผู้ใช้ในคอมพิวเตอร์ของผู้ใช้เอง) ได้

2.4.2 ข้อดีและข้อเสียของ JavaScript

การทำงานของ JavaScript เกิดขึ้นบนเบราว์เซอร์ (เรียกว่าเป็น client-side script) ดังนั้นไม่ว่าจะใช้เบราว์เซอร์อะไร หรือที่ไหน ก็ยังสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า server-side script) ดังนั้นจึงต้องใช้บนเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านี้เท่านั้น อย่างไรก็ตาม จากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษา server-side script อยู่ (ความจริง JavaScript ที่ทำงานบนเซิร์ฟเวอร์ก็มี ซึ่งต้องอาศัยเซิร์ฟเวอร์ที่สนับสนุนโดยเฉพาะเช่นกัน แต่ไม่เป็นที่นิยมนัก)

2.4.3 ภาษา Script ที่ทำงานบนฝั่ง server (Server-side Script)

ในปัจจุบันมีผู้พัฒนาภาษา JavaScript ทางฝั่ง Server-side โดยมีที่มาจาก Google Chrome ซึ่งเมื่อก่อน Browser จะประมวลผล JavaScript ซ้ำมากแต่เมื่อ Google ใช้งาน Google V8 JavaScript Engine ทำให้การประมวลผลมีความไวสูงมากแถมเป็น open source อีกด้วยจนความสามารถนี้ไปเข้าตา Ryan Dahl ทำให้เขาตั้ง V8 ออกมาใช้ทางฝั่ง Server บ้าง โดยเติม function บางอย่างให้ Server ต้องการใช้ทำให้เราสามารถเขียน JavaScript ทางฝั่ง Server ได้จึงเป็นต้นกำเนิดของ “Node.js”

2.5 Node.js

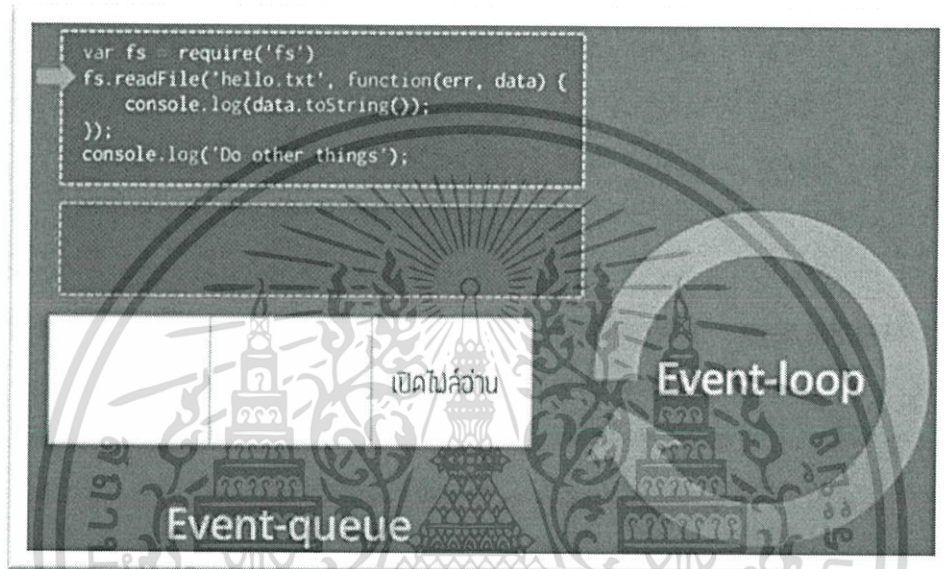
Node.js คือ environment เพื่อให้เราเขียน software มารันตามที่เราต้องการ ภายใต้ environment ของ node นี้ โดยภาษาที่เราจะใช้เขียน คือภาษา JavaScript เหมือนที่เราใช้เขียนบนหน้าเว็บเลย โดยจะมีตัว compiler คือ Google JavaScript engine V8 ก็คือตัวประมวลผลภาษา JavaScript ที่ทาง Google พัฒนาขึ้นมาตัวเอง (เป็น open source) สรุปคือ เราเขียนโปรแกรมด้วยภาษา JavaScript โดยความสามารถมันทำได้หลายอย่างมาก ไม่ว่าจะเป็นประมวลผลงานต่างๆ หรือติดต่อผ่าน socket หรือไปจนถึงสั่ง command line บนเครื่องเลย ทำได้หมด node.js นำ V8 มาทำเป็นตัว compiler เพราะว่ามันประมวลผลได้รวดเร็วมาก ผลที่ได้คือมันทำตัวคล้ายๆกับจะเป็น parallel execution ด้วย คือ ทำงานหลายอย่างพร้อมกันในเวลาเดียวกัน แต่ความจริงคือไม่ใช่ เพราะว่ามันเป็น single thread แต่มันปรับกระบวนการโดยเอาแต่ละงานไปเข้าคิวเวลาเอาไว้ก่อน (ยังไม่ประมวลผล) แล้วค่อยประมวลผลงานตามคิวเวลา ไม่ได้เรียงตามขั้นตอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่เสร็จหรือไม่เสร็จ ซึ่งกระบวนการเข้าคิวเวลามันใช้เวลาน้อยมากๆ เราเลยมองเห็นเป็นว่ามันเริ่มต้นทำงานได้พร้อมกัน

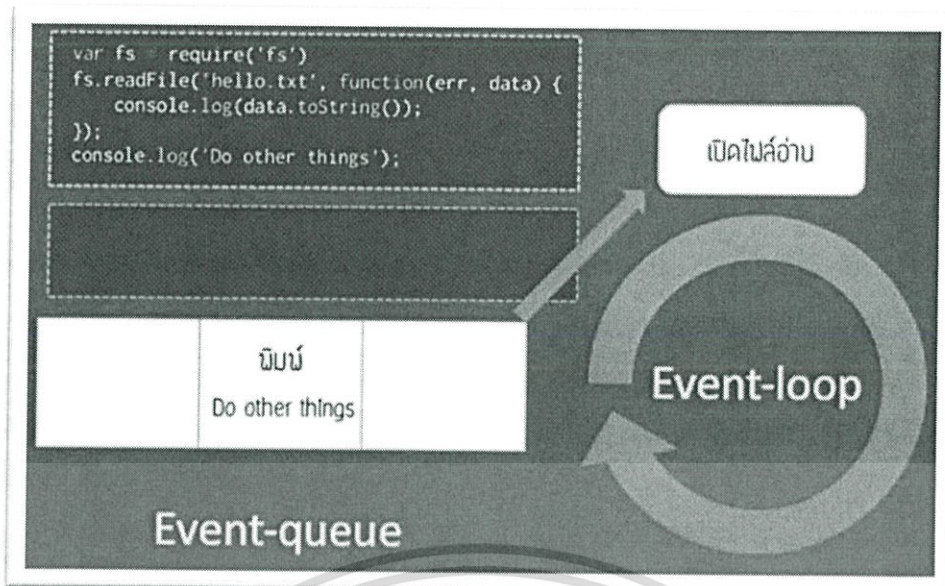
2.5.1 Concept การทำงาน

จะมี Event-loop วนอยู่ตลอดเวลาเพื่อดูว่ามี Event อะไรใหม่ๆเข้ามาบ้างโดย Event ที่ยกตัวอย่างเมื่อเราสั่ง read file Event ก็จะทำคำสั่ง เปิดอ่านไฟล์ หลังจากนั้นก็จะมาทำบรรทัด console.log สั่งให้พิมพ์ Do other things



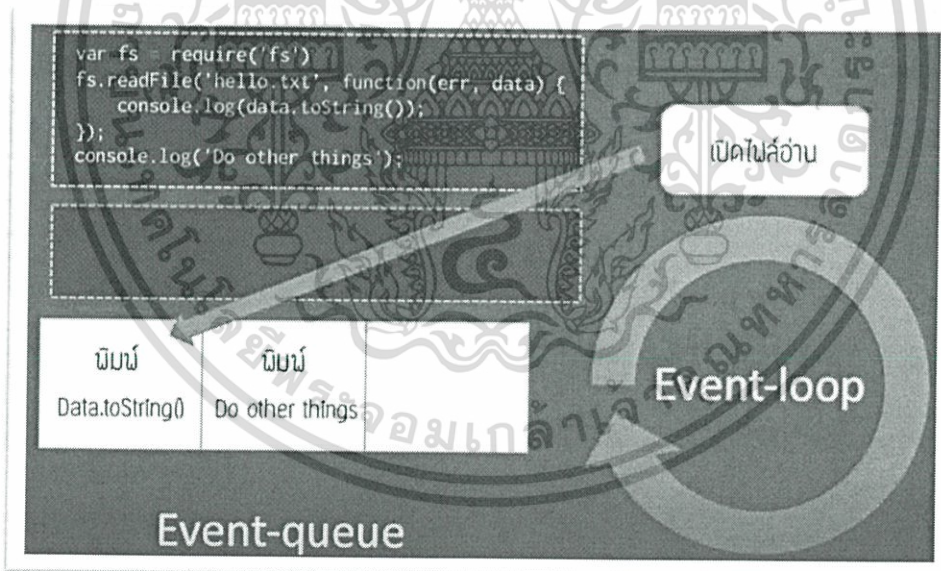
รูปที่ 2.15 Event-loop วนอยู่ตลอดเวลาเพื่อดู Event ใหม่ที่เข้ามา

โดยจะนำทุก Event มาต่อกันที่ Event-queue Event-loop จะวนรอบแล้วดูว่า Event ไหนอยู่ที่หัวคิวก็จะเริ่มทำก่อน



รูปที่ 2.16 Event-queue

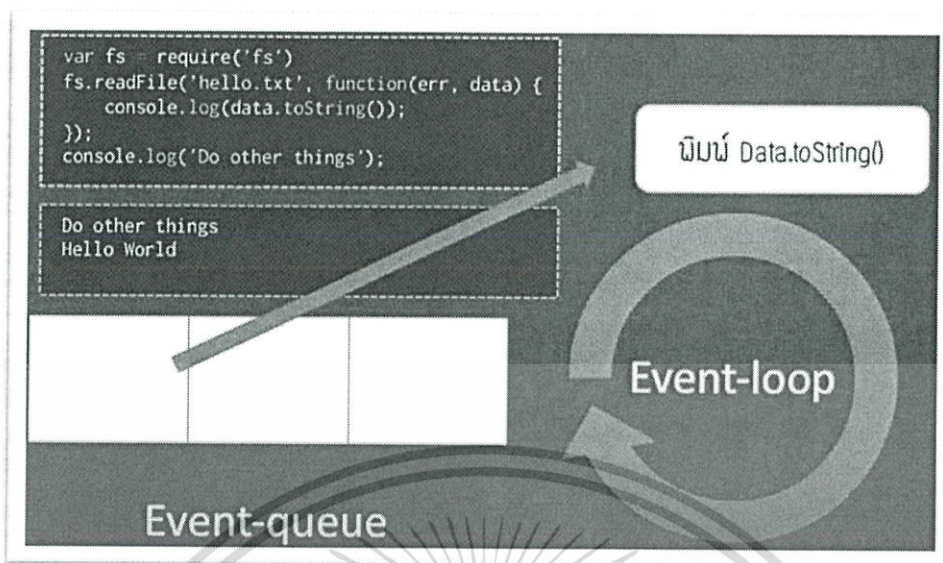
พอพบว่าเปิดอ่านไฟล์อยู่ก่อนมันก็จะมี Event กลับมาว่าเปิดอ่านไฟล์แล้วให้พิมพ์ว่า Data.toString() ก็จะเอา Event ใหม่ไปต่อคิวหลังจากคำสั่ง พิมพ์ Do other things



รูปที่ 2.17 การใช้ Data.toString()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะพิมพ์ Do other things และ Hello World ตามลำดับ



รูปที่ 2.18 การพิมพ์ Do other things และ Hello World

2.5.2 Node.js เอามาใช้ประโยชน์อย่างไร

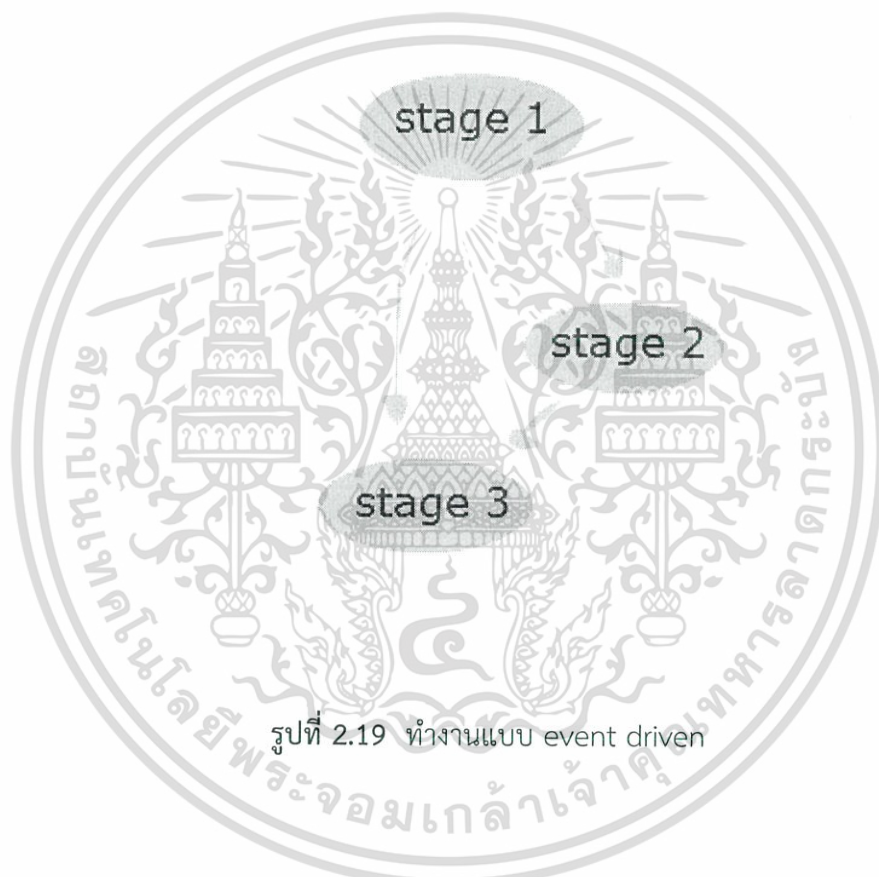
ส่วนใหญ่จะนิยมใช้ node.js ในงานที่ทำเป็นเบื้องหลังก็คือ งานที่ประมวลผลฝั่ง server ซึ่งเป็นงานที่อาจจะต้อง interface กับผู้ใช้ หรือไม่ต้อง interface ก็ได้ ตัวอย่างงานที่ต้อง interface กับคนใช้ก็คือ การทำตัวเองเป็น http server ในการดึงหน้าเว็บมาแสดงผลให้กับ user หรือว่า การเปิด socket เพื่อรับส่งข้อมูลกันระหว่าง server กับคนใช้ ที่อาจจะเอาไปทำเป็นห้อง chat , ทำเกม, ทำระบบที่ป้อนข้อมูลเพื่อคำนวณเอาผลลัพธ์ เป็นต้น มันเอาไว้ประมวลผลล้วนๆเลย หน้าตาก็บ่อยให้เป็นหน้าที่ของ html + css + javascript ที่เขียนหน้าเว็บตามปกติ หรือ ตัวอย่างงานที่ไม่ต้อง interface กับคนใช้ก็เช่นเอาไปทำ spider crawler เว็บ คือให้เปิดเว็บอ่านแล้วเก็บข้อมูลไปเรื่อยๆ หรือว่าโปรแกรมที่รอรับค่าจาก streaming ที่ต่างๆ เพื่อเอามามันทักเอาไว้ เหล่านี้ไม่จำเป็นต้อง interface กับคนใช้ต่างก็ใช้ node.js ทำงานด้วยกันทั้งนั้น

2.5.3 Node.js ทำงานแบบ event driven

การทำงานของ node เค้าจะเรียกมันว่า การขับเคลื่อนด้วย event ต่างๆที่เกิดขึ้น ทำให้เรากระโดดจาก event หนึ่งที่เสร็จแล้วไปอีก event หนึ่งได้ด้วยการสั่งงานมันต่อเนื่องกันไปเรื่อยๆ หรือว่าการสั่งให้ หลากๆ event เริ่มทำงานในเวลาใกล้ๆเคียงกันเลยก็ได้เช่นกัน ประโยชน์อีกอย่างที่ได้จาก event driven ก็คือ การสั่งให้มันรอรับ event นั้นไปตลอดกาลโดยไม่เปลือง resource เช่น การเชื่อมต่อไปยัง streaming channel สักที่หนึ่ง อาจจะเป็น text หรือข้อมูลบางอย่างเช่นปริมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

น้ำฝนทิ้งเอาไว้ หากต้นทางของ streaming ยังไม่มีข้อมูลส่งมา มันก็จะไม่เกิด event ใดๆ node.js ก็ จะอยู่นิ่งๆ แต่หากต้นทาง streaming มาแล้ว node.js ก็ จะทำงานเพื่อตอบสนองต่อ event ที่เกิดขึ้น นั้นทันที (ตรงนี้ขึ้นอยู่กับว่าเราจะเอาข้อมูลไปทำอะไรต่อ) ด้วยประโยชน์ของ event driven นี้ทำให้ เราเอามาต่อยอดได้อีกหลายอย่างเช่น การ subscribe pubsub (เช่น pubsub ใน redis) หากเกิด event publish เมื่อไร node.js ที่ subscribe รอเอาไว้ช้านานแล้ว ก็จะถูกกระตุ้นแล้วทำงานตามที่ เราเขียนเอาไว้โดยทันที ซึ่งแบบนี้จะทำให้ลดการสูญเสีย header ในการเริ่มต้นประมวลตั้งแต่จุดแรก ไปได้มาก รวมทั้งไม่ต้องไปหน่วงเวลาเพื่อคอยเช็ค event เหมือนเวลาเราเขียน ajax เลยแม้แต่น้อย เรียกได้ว่าอยู่เฉยๆเถอะ ถ้ามีข้อมูลมาเดี่ยวไปกระตุ้นให้ทำงานเอง (โดยต้องรันงานนั้นรอเอาไว้ก่อน แล้ว แต่อย่างที่บอก เปลืองทรัพยากรน้อยมาก)



รูปที่ 2.19 ทำงานแบบ event driven

ปัจจุบัน NodeJS ถูกนำมาทำเป็น Web Server , Mobile Hybrid , IOT , Webkit , TVOS ,OS อื่นๆอีกมาก เรียกได้ว่าเข้าถึงได้หลากหลายเทคโนโลยี

ข้อดี

1. ประมวลผล JavaScript ได้อย่างรวดเร็ว
2. ใช้ JavaScript เขียนทางฝั่ง Font-end และทางฝั่ง Back-end
3. Non-blocking I/O คือจะสามารถทำงานได้อย่างต่อเนื่องโดยไม่ต้องรอให้ function ทั้งหมดทำงานเสร็จต่างคนต่างทำงานได้ในหลายๆ function ส่วน blocking จะต้องรออ่านไฟล์ ทั้งหมดให้เสร็จก่อนถึงจะเปิดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Real-time application

5. มี Module หรือ package ให้เลือกใช้เยอะมาก

ข้อเสีย

1. Single-thread ใช้ CPU เดียวเวลาใช้งานจะใช้ได้แค่ core เดียว

2. ไม่เหมาะกับ CPU-intensive task พวก function นานๆจะทำงานช้าควรจะถูกแบ่งเป็น task การทำงานย่อยๆจะได้ทำงานไม่นานมาก

3. Callback Hell แก้ไขได้ด้วย promise

2.5.4 Socket.io

คือ Library ที่เขียนขึ้นโดย ภาษา node.js (javascript) ทำงานแบบ real-time และรองรับการทำงานหลายๆ อย่างพร้อมกันในเวลาเดียว (Asynchronous) การทำงานของ socket.io คือทำการวนลูปตลอดเวลา เพื่อคอยรับ event เข้ามาแล้วนำไปทำงานต่อ

Socket.io สร้างมาเพื่อให้การรับ/ส่งข้อมูลระหว่าง server และ client เป็นเรื่องง่าย method ที่เรานิยมใช้งานกันจะมี 2 ตัวคือ

socket.on (message, callback) ใช้สำหรับรับข้อความจาก server (receive message from server.)

socket.emit (message, args) ใช้สำหรับส่งข้อความไปที่ server (send message to server.)

2.5.4.1 web socket เอาไว้ใช้เพื่ออะไร

เปรียบเสมือนช่องทางที่เอาไว้ใช้ติดต่อสื่อสาร กล่าวคือ การเปิด socket คู่กัน เหมือนการยกหูโทรหา แล้วค้างสายเอาไว้ ไม่วางนั่นเอง วางก็ต่อเมื่อปิดหน้าเว็บไป ทีนี้หลังจากที่เราเปิด socket เราก็ได้ประโยชน์เพิ่มมา เพราะว่าเหมือนคนคุยโทรศัพท์กัน ทั้งสองฝ่ายสามารถเป็นฝ่ายเริ่มต้นพูดก่อนได้ ไม่จำเป็นว่าฝ่าย client จะต้องเป็นคนเริ่มต้นพูดก่อน หลังจากที่เราโทรหากันแล้ว ฝ่าย server อาจจะพูดไปที่ client อย่างเดียวเลยก็ได้ แบบนี้เค้าจะเรียกว่าการ "push" ซึ่งเมื่อรวมๆกันแล้ว นี่ก็คือประโยชน์ของ web socket นั่นเอง คือการยกหูโทรหากันและหลังจากที่เชื่อมต่อกันแล้ว ต่างฝ่ายต่างเริ่มพูดก่อนได้เลย "อย่าลืม event driven" การที่มีฝ่ายหนึ่งฝ่ายใด เริ่มพูดก่อน นั้นทำให้เกิด event ขึ้นแล้ว แต่ว่า event ดังกล่าว มันจะต้องเกิดจากการตกลงกันเอาไว้ล่วงหน้าก่อนแล้ว ในตัวอย่างนี้ จะยกตัวอย่างว่า client (ลูกค้า) โทรไปสั่ง (web socket) pizza โดยมีพนักงาน (server) คอยบริการเรา เราโทรสั่ง pizza แล้วระบุว่า เอาหน้าฮาวายเอี้ยน โดยการ "ระบุหน้า pizza" เค้าจะเรียกว่า "event" แล้วหน้า "ฮาวายเอี้ยน" มันก็คือ "data" ที่ส่งจาก client ไปหา server นั่นเอง (แยกกันให้ตี้นะครับ) แต่กลับกัน หากพนักงาน(server) เป็นคนพูดก่อนว่า เราขอเสนอ pizza หน้าฮาวายเอี้ยน ดังนั้น "นำเสนอ" ก็คือ "event" และ ฮาวายเอี้ยน มันก็คือ "data" ซึ่งจะถูกส่งกลับไปกระตุ้นการทำงานที่ฝั่ง client (ทำให้หิว หรือ สั่งเลย หรือ ขอหน้าอื่นเพิ่ม)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.5 jQuery

jQuery คือ JavaScript Library ซึ่งถูกออกแบบมาเพื่อให้การเขียน JavaScript นั้นมีความสะดวกและง่ายขึ้น เพราะว่าการนำ JavaScript เอาไปประยุกต์กับงานจำพวกเว็บ (Client-side JavaScript) นั้นเป็นสิ่งที่ยุ่งยาก ไม่ว่าจะเป็นเรื่องความไม่เข้ากันของ Web Browser แต่ละค่าย, DOM หรือ API เป็นต้น ดังนั้น jQuery จึงรวมเอา Object และ Function ต่างๆ ที่จำเป็นมารวบรวมไว้ในรูปแบบของ Library พอเป็นเช่นนี้แล้ว ไม่ว่าจะโค้ดที่เขียนจะใช้ JavaScript หลายบรรทัดขนาดไหน ก็สามารถทำให้สั้นลงได้ อาจทำให้เหลือสั้นเพียงแค่บรรทัดเดียวเท่านั้น

2.5.5.1 ข้อดีของ JQuery

1. ความสามารถในการทำงานแบบ ajax ก็คือทำงานได้ในรูปแบบที่ไม่ต้องโหลดหน้าใหม่ให้คนใช้เห็น
2. การสร้าง animation ได้แบบง่ายๆเลย ไม่ว่าจะทำรูปให้เคลื่อนที่ หรือ DIV
3. สามารถจัดการกับ css (style sheet) ล่อง element นั้นๆได้
4. ค้นหา element ที่เราต้องการและจัดการ เพิ่มหรือลบ Attributes ที่เราต้องการ
5. ทำ Effect ต่างๆกับ Element ที่เราต้องการ เช่นการ hide DIV ที่เราต้องการ
6. การดักเหตุการณ์ต่างๆ
7. รองรับกับภาษาได้ทุกรูปแบบ (asp, php, .net , html ,....)
8. รองรับได้ทุกๆ Browser ดีกว่าเขียน Javascript แบบเดิมๆ
9. ไม่ต้องเขียนโค้ดใหม่เพราะมีให้ใช้อยู่แล้ว
10. ทำให้นักพัฒนาเขียนโค้ดน้อยลงแต่ทำงานได้มากขึ้น
11. ความเร็วในการทำงาน ขนาดที่เล็ก

2.5.5.2 การใช้ jQuery

สำหรับการใช้ jQuery สำหรับการเขียนหน้าเว็บของเรานั้นเราสามารถเรียกออกมาใช้ได้ 2 วิธี

1. การดาวน์โหลด jQuery library โดยตรงจาก jquery.com วิธีการดาวน์โหลดจะมีอยู่ 2 รูปแบบสำหรับการดาวน์โหลด

1. Production version
2. Development version

สามารถใช้ในการเขียน HTML ได้ด้วยการอ้างแท็ก `<script>` แต่การเขียน `<script>` ควรจะอยู่ภายใต้ของ `<head>` ดังโค้ดตัวอย่าง

```
<head>
<script src="jquery-3.1.1.min.js"></script>
</head>
```

รูปที่ 2.20 การใช้คำสั่ง <script>

2. การเรียกใช้ jQuery จาก CDN (Content Delivery Network) ถ้าเราไม่ต้องการที่จะดาวน์โหลดเราก็สามารถที่จะเรียกใช้ jQuery ได้จาก CDN (Content Delivery Network) ดังโค้ดตัวอย่าง

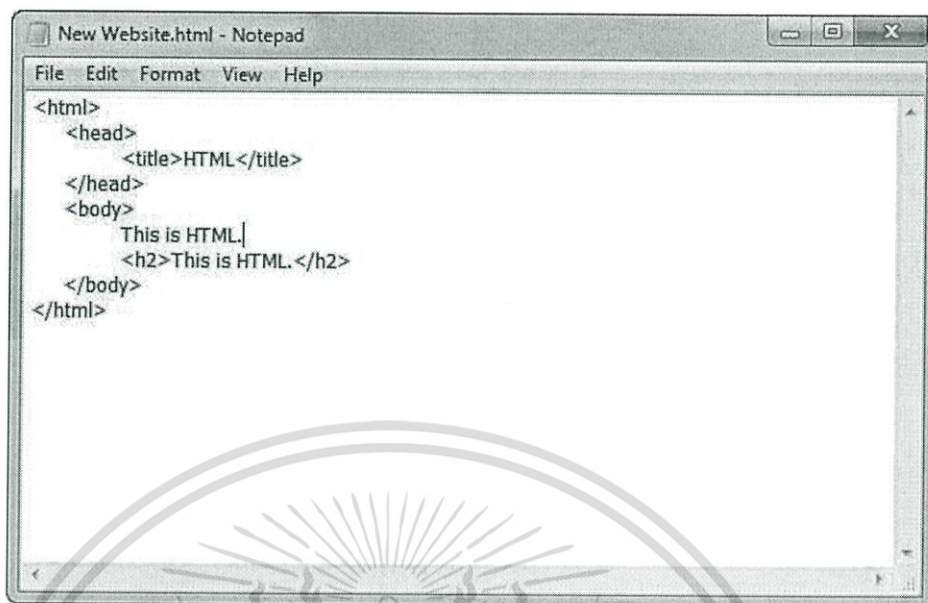
```
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
</head>
```

รูปที่ 2.21 การเรียกใช้ jQuery จาก CDN

2.6 HTML

HTML หรือชื่อเต็มๆ ก็คือ Hypertext Markup Language เป็นภาษาประเภท Markup ที่ออกแบบมาเพื่อใช้ในการสร้างเว็บเพจ มีโครงสร้างการเขียนโดยอาศัยตัวกำกับ (Tag) ควบคุมการแสดงผลข้อความ รูปภาพ เสียง อื่นๆ ที่สามารถเรียกดูผ่านทางเว็บเบราว์เซอร์ได้ แต่ละ Tag สามารถระบุหรือควบคุมการแสดงผลของเว็บให้เป็นไปตามที่ผู้ออกแบบเว็บไซต์กำหนดไว้ หรือจะให้เข้าใจง่ายๆ ก็คือ เว็บที่เราเข้าอยู่ในทุกๆ วันนี้ ก็ล้วนถูกแสดงผลด้วยโค้ด HTML ทั้งนี้ HTML พัฒนามาจากภาษา SGML และถูกพัฒนามาเรื่อยๆ ตั้งแต่เวอร์ชันแรก สู่เวอร์ชัน HTML 2.0, HTML 3.2, HTML 4.1 และตัวล่าสุดคือ HTML 5 (ซึ่งอยู่ในระหว่างการพัฒนา) และยังมีมีการพัฒนารูปแบบของ HTML แบบใหม่ที่เรียกกันว่า XHTML ซึ่งมีความสามารถและมาตรฐานที่รัดกุมกว่าอีกด้วย HTML มีรูปแบบการทำงานง่ายๆ HTML จะอ่านจากซ้ายไปขวา จากบนลงล่าง เมื่อเราพิมพ์ข้อความตัวอักษรธรรมดาลงไปแล้วทำการบันทึก พอเราเปิดไฟล์นั้นๆ มาดูผลลัพธ์ที่จะแสดงผลที่หน้าจอก็จะเป็นเหมือนตอนที่พิมพ์ข้อความตัวอักษรธรรมดาลงไป หากต้องการผลลัพธ์ที่แตกต่างต้องใช้ Tag ในการควบคุมการแสดงผล

2.6.1 ตัวอย่าง CODE ภาษา HTML



```

New Website.html - Notepad
File Edit Format View Help
<html>
  <head>
    <title>HTML</title>
  </head>
  <body>
    This is HTML.
    <h2>This is HTML.</h2>
  </body>
</html>

```

รูปที่ 2.22 ตัวอย่าง CODE ภาษา HTML

จากภาพจะเห็นว่า... HTML มีรูปแบบการเขียนที่ชัดเจน จะประกอบด้วย Tag พื้นฐาน ดังนี้

Tag <html>...</html> เป็นส่วนประกาศที่กำหนดหัวละท้ายของเอกสาร เพื่อให้เบราว์เซอร์ทราบและแสดงผลได้ถูกต้อง

Tag <head>...</head> เป็นส่วนหัวเรื่องของเอกสาร ภายในจะมี Tag <title>...</title> ใช้สำหรับการกำหนดชื่อของเอกสาร

Tag <body>...</body> เป็นส่วนที่มีรายละเอียดมากที่สุด จะบรรจุข้อมูลต่างๆ ที่ต้องการให้แสดงบนหน้าเว็บไซต์ของเรา ทั้งข้อความ รูปภาพ เป็นต้น

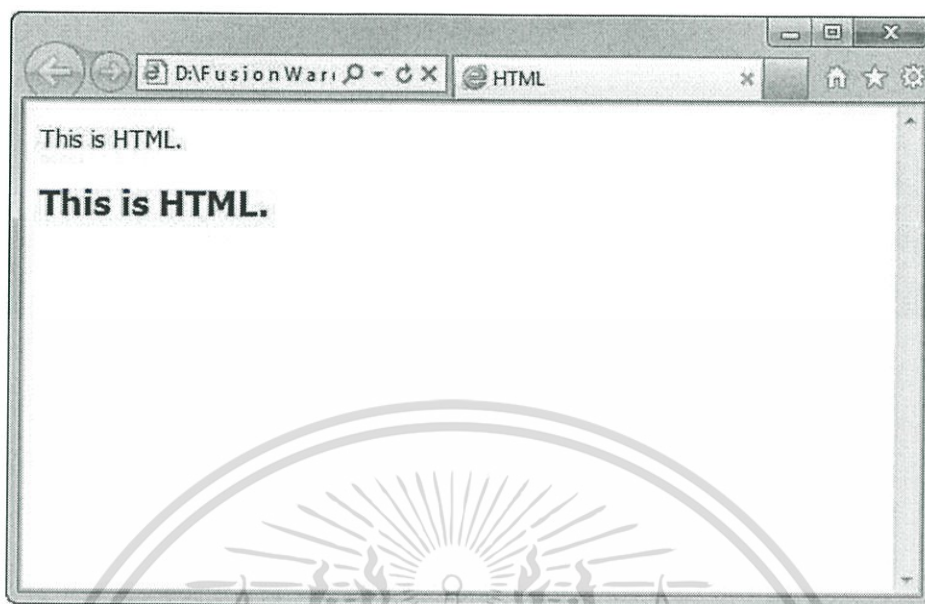
2.6.2 ผลลัพธ์หน้าจอแสดงผล

ในการสร้างเว็บเพจโดยใช้ภาษา HTML เขียนนั้น สามารถใช้โปรแกรม Text Editor ต่างๆ เขียนได้ เช่น Note Pad , Word Pad ฯลฯ หรือจะใช้โปรแกรมประเภท WYSIWYG (What You See Is What You Get) ที่เป็นเครื่องมือช่วยสร้างเว็บเพจซึ่งอำนวยความสะดวกในการสร้างหน้า HTML เช่น Dream Weaver, Microsoft FrontPage ฯลฯ

ส่วนในการเรียกใช้งานหรือทดสอบการทำงานของเอกสาร HTML ที่เราได้เขียนไว้นั้นจะใช้โปรแกรมเว็บเบราว์เซอร์ (web browser) เช่น Internet Explorer (IE), Google Chrome, Mozilla

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Firefox, Safari และอื่นๆ เป็นต้น เป็นเครื่องมือแสดงผลไฟล์ HTML ของเราออกมาเป็นตัวอักษร ภาพและเสียง



รูปที่ 2.23 ผลลัพธ์จากภาษา HTML

2.7 CSS

CSS ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตล์ชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้ง หรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ (Style) เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรกใน HTML 4.0 เมื่อปีพ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดย องค์กร World Wide Web Consortium หรือ W3C

CSS กับ HTML / XHTML นั้นทำหน้าที่คนละอย่างกัน โดย HTML / XHTML จะทำหน้าที่ในการวางโครงสร้างเอกสารอย่างเป็นทางการ ถูกต้อง เข้าใจง่าย ไม่เกี่ยวข้องกับการแสดงผล ส่วน CSS จะทำหน้าที่ในการตกแต่งเอกสารให้สวยงาม เรียกได้ว่า HTML /XHTML คือส่วน coding ส่วน CSS คือส่วน design

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1 ประโยชน์ของ CSS

2.7.1.1 CSS มีคุณสมบัติมากกว่า tag ของ html เช่น การกำหนดกรอบให้ข้อความรวมทั้งสี รูปแบบของข้อความที่กล่าวมาแล้ว

2.7.1.2 CSS นั้นกำหนดที่ต้นของไฟล์ html หรือตำแหน่งอื่น ๆ ก็ได้ และสามารถมีผล กับเอกสารทั้งหมด หมายถึงกำหนด ครั้งเดียวจุดเดียวก็มีผลกับการแสดงผลทั้งหมด ทำให้เวลาแก้ไขหรือปรับปรุงทำได้สะดวก ไม่ต้องไล่ตามแก้ tag ต่างๆ ทั่วทั้งเอกสาร

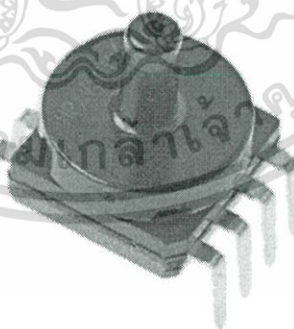
2.7.1.3 CSS สามารถกำหนดแยกไว้ต่างหากจาก ไฟล์เอกสาร html และสามารถนำมาใช้ร่วมกับเอกสารหลายไฟล์ได้ การแก้ไขก็แก้เพียง จุดเดียวก็มีผลกับเอกสารทั้งหมด

2.8 Pressure Sensor

Pressure Sensor คือ อุปกรณ์ตรวจวัดความดันของก๊าซหรือเหลว เซ็นเซอร์จะส่งสัญญาณทางไฟฟ้าที่มีความสัมพันธ์กับความดัน เซ็นเซอร์ถูกสร้างให้มีเยื่อบางที่สามารถโค้งงอตามความดัน ซึ่งระดับความโค้งงอสามารถวัดได้ด้วยการเปลี่ยนแปลงความต้านทาน หรือ การเปลี่ยนแปลงของการเก็บประจุในการพัฒนาเซ็นเซอร์แรกเริ่มทำด้วยวิธีการประดิษฐ์โครงสร้างจุลภาคบนพื้นผิว (Surface Micromachining) สำหรับใช้งานในช่วงความดันต่างๆ เพื่อประยุกต์ใช้ในอุตสาหกรรมยานยนต์และเซ็นเซอร์ตรวจวัดแรงดันในเส้นเลือด เป็นต้น

2.8.1 Integrated Silicon Pressure Sensor On-Chip Signal Conditioned, Temperature Compensated and Calibrated

2.8.1.1 MPXV5010GC7U



รูปที่ 2.24 MPXV5010 GC7U

ข้อมูลทางเทคนิค

Maximum Overload Pressure	40 kPa
Mounting Type	through hole
Package Type	Case 482C-03

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Count	8
Dimensions	10.79 x 10.79 x 13.21mm
Length	10.79 mm
Width	10.79 mm
Height	13.21 mm
Maximum Operating Pressure	10 kPa
Maximum Operating Supply Voltage	5.25 V dc
Maximum Operating Temperature	+125 °C
Maximum Output Voltage	4.7 V
Minimum Operating Pressure	0 kPa
Minimum Operating Temperature	-40 °C
Minimum Operating Supply Voltage	4.75 V dc
Minimum Output Voltage	4.475 V dc

2.8.1.2 MPXV5050GC6U



รูปที่ 2.25 MPXV5050GC6U

ข้อมูลทางเทคนิค

Maximum Overload Pressure	200 kPa
Typical Sensitivity	90 mV/kPa
Mounting Type	Surface Mount
Package Type	Case 482A-01
Pin Count	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dimensions	10.79 x 10.79 x 13.21mm
Length	10.79mm
Width	10.79mm
Height	13.21mm
Maximum Operating Pressure	50kPa
Maximum Operating Supply Voltage	5.25 V dc
Maximum Operating Temperature	+125 °C
Maximum Output Voltage	4.7 V
Minimum Operating Pressure	0kPa
Minimum Output Voltage	4.587 V dc
Minimum Operating Supply Voltage	4.75 V dc
Minimum Operating Temperature	-40 °C

2.8.1.3 MPXV5004GP



รูปที่ 2.26 MPXV5004GP

ข้อมูลทางเทคนิค

Maximum Operating Temperature	+85 °C
Minimum Operating Temperature	0 °C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 การดูแลผู้ป่วยในเตียง

“ผู้ป่วยในเตียง” หมายถึง บุคคลที่มีความเจ็บป่วยและต้องนอนอยู่ในเตียงหรือบนที่นอนตลอดเวลา อาจเป็นระยะหนึ่งหรือตลอดไป เช่น ผู้ป่วยอัมพาตที่เคลื่อนไหวไม่ได้ ผู้ป่วยชราที่มีความอ่อนเพลียมาก ผู้ป่วยเข้าเฝือกตัว เป็นต้น ซึ่งปัจจุบันมีผู้ป่วยประเภทนี้จำนวนไม่น้อยที่จำเป็นต้องได้รับการดูแลอยู่ในบ้าน เนื่องจากสถานบริการมีไม่เพียงพอนั่นเอง

การจัดห้องนอนหรือที่นอนภายในบ้านสำหรับผู้ป่วย (ทุกประเภท) ควรโปร่งมีอากาศถ่ายเทดี ไม่รกรุงรัง และถ้าเป็นไปได้ควรเป็นห้องหรือมุมใดมุมหนึ่งของบ้าน ห่างไกลจากสิ่งรบกวน เช่น แสง เสียง กลิ่น ที่สำคัญที่สุด ถึงแม้ว่าห้องหรือบริเวณจะคับแคบ ควรต้องมีความสะอาด ปราศจากสิ่งรบกวน และสัตว์ที่ก่อความรำคาญ เช่น มด แมลงสาบ แมลงวัน ฯลฯ

สำหรับที่นอนนั้นจะนอนบนเตียงหรือบนพื้น ก็แล้วแต่ความเป็นไปได้ของแต่ละครอบครัว แต่ถ้าสามารถจัดหาเตียงได้ จะมีผลดีต่อผู้ป่วย คือ ไม่ถูกรบกวนจากเสียงฝีเท้าคนเดินหรือสัตว์ และยังสามารถมองผ่านหน้าต่างออกไปภายนอก นอกจากนี้ผู้ดูแลยังได้รับความสะดวกในการพยาบาล ไม่เมื่อยล้า

ผู้ป่วยที่ต้องอยู่บนเตียงหรือนอนตลอดเวลา การดูแลที่สำคัญ ได้แก่ การอาบน้ำ การสระผม การทำความสะอาดที่นอน และการป้องกันแผลกดทับ (จะขอกล่าวเป็นตอนๆ ไป) รวมทั้งผู้ป่วยบางรายอาจต้องมีการสวนอุจจาระในบางครั้ง ซึ่งการดูแลทั้งหมดนี้ต้องกระทำบนเตียงหรือบนที่นอนทั้งนั้น

เกี่ยวกับอาหาร ผู้ป่วยที่ต้องนอนอยู่บนเตียงตลอดเวลา มักจะไม่ค่อยอยากอาหาร เนื่องจากต้องนอนอยู่เฉยๆ ทำให้ลำไส้มีการเคลื่อนไหวน้อย อาหารย่อยช้า และยังมีควมวิตกกังวลคิดมากในเรื่องความเจ็บป่วย ฯลฯ สำหรับอาหารที่จัดให้ผู้ป่วยอาจเป็น อาหารธรรมดา อาหารอ่อน อาหารเหลว และอาหารเฉพาะโรค ทั้งนี้ขึ้นอยู่กับลักษณะการเจ็บป่วยของแต่ละบุคคล

การป้อนอาหาร ผู้ดูแลควรป้อนอย่างนุ่มนวล พุดคุย บางครั้งอาจต้องปลอมโยนให้กำลังใจ และมีการจัดอาหารให้มารับประทาน ให้มีลักษณะเหมือนหรือใกล้เคียงกับภาวะธรรมดา

นอกจากความไม่สมดุลระหว่างบุคลากรทางการแพทย์กับจำนวนประชาชนแล้ว ภาวะสังคมและเศรษฐกิจในปัจจุบันยังทำให้ประชาชนจำนวนมากไม่สามารถใช้บริการของสถานพยาบาลได้ทุกครั้ง การดูแลตนเองหรือดูแลซึ่งกันและกันเมื่อมีความผิดปกติบางอย่างจึงมีความสำคัญที่จะช่วยแก้ปัญหาเฉพาะหน้า หรือบรรเทาภาวะวิกฤติก่อนที่จะไปสถานพยาบาลได้ และปัญหาบางอย่างต้องการเพียง พยาบาลในบ้าน

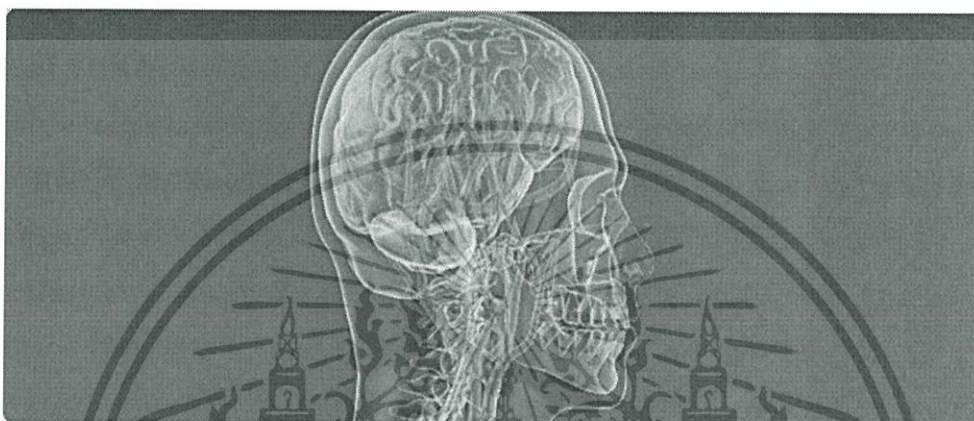
2.10 อัมพฤกษ์อัมพาต โรคเรื้อรังที่ต้องดูแล

เป็นที่ตระหนักกันว่า อัมพฤกษ์อัมพาต หรือโรคหลอดเลือดสมอง เป็นสาเหตุการตายอันดับ 3 ทั่วโลก และเป็นสาเหตุที่สำคัญของความพิการที่รุนแรง ข้อมูลทางสถิติพบว่าในแต่ละปีมี

ผู้ป่วยโรคหลอดเลือดสมองเกิดใหม่ทั่วโลกราว 10 - 15 ล้านคน ในจำนวนนี้ 5 ล้านคนเสียชีวิต และอีก 5 ล้านคนกลายเป็นคนพิการอย่างถาวร

สำหรับในประเทศไทยสถิติจากกระทรวงสาธารณสุขพบว่า โรคหลอดเลือดสมองเป็นสาเหตุการตาย หรือพิการสูงเป็นอันดับ 3 ในเพศชาย รองจากโรคเอดส์และอุบัติเหตุ และสูงเป็นอันดับ 2 ในเพศหญิงรองจากโรคเอดส์ จากข้อมูลดังกล่าวจะเห็นว่าโรคหลอดเลือดสมองเป็นโรคที่คุกคามต่อชีวิตและความเป็นอยู่ของประชากรทั่วโลก

2.10.1 โรคอัมพาตโรคอัมพฤกษ์เกิดจากอะไร?



รูปที่ 2.27 สาเหตุการเกิดโรคอัมพาตโรคอัมพฤกษ์

โรคอัมพาตโรคอัมพฤกษ์มีสาเหตุเกิดจากสมองขาดเลือดทันทีภายในระยะเวลาเป็นนาทีหรือชั่วโมง ไม่ใช่ค่อยๆเป็นค่อยๆไป ซึ่งเกิดได้จาก 2 สาเหตุสำคัญคือ

1. สมองขาดเลือด พบประมาณ 70-80% ของผู้ป่วยอัมพฤกษ์อัมพาต เกิดจากความผิดปกติของหลอดเลือดแดงที่ไปเลี้ยงสมองเกิดการตีบหรืออุดตัน ซึ่งเป็นผลจากการที่ผู้ป่วยมีปัจจัยเสี่ยงต่าง ๆ เช่น โรคเบาหวาน ความดันโลหิตสูง การบริโภคอาหารที่มีไขมันสูง การสูบบุหรี่ ภาวะหัวใจวายหรือหัวใจเต้นผิดจังหวะบางชนิด การขาดการออกกำลังกายอย่างสม่ำเสมอ โรคเลือดบางอย่าง เช่น ภาวะเลือดข้นผิดปกติ เกร็ดเลือดสูง เม็ดเลือดขาวสูงผิดปกติ ผู้ป่วยที่มีปัจจัยเสี่ยงดังกล่าวอยู่เป็นเวลานานจะเป็นผลให้ผนังหลอดเลือดหนาและแข็งตัว เกิดการตีบหรืออุดตัน ทำให้สมองขาดเลือดเกิดอัมพาตตามมาในที่สุด โดยผู้ป่วยเหล่านี้มักมีโรคหลอดเลือดหัวใจตีบร่วมด้วย

2. หลอดเลือดสมองแตก เมื่อเกิดการแตกของหลอดเลือดสมอง ก้อนเลือดจะเบียดดันเนื้อสมองส่วนที่ดีทำให้เสียหายที่เซลล์สมองทำงานผิดปกติ เกิดอัมพฤกษ์อัมพาตตามมา ภาวะนี้มักสัมพันธ์กับโรคความดันโลหิตสูงที่ไม่ได้รับการรักษาอยู่เป็นเวลานาน นอกจากนี้ยังเกิดจากความเครียด การดื่มแอลกอฮอล์ รวมทั้งยาบางชนิด

2.10.2 ปัจจัยเสี่ยงต่อการเกิดอัมพาตอัมพฤกษ์มีอะไรบ้าง?

จากการศึกษาของโครงการศึกษาโรคหลอดเลือดสมองในประเทศไทยพบว่า ปัจจัยเสี่ยงต่อโรคหลอดเลือดสมองหรืออัมพาตอัมพฤกษ์ของคนไทยกลุ่มศึกษาได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อายุ ตั้งแต่ 65 ปีขึ้นไป
- เพศ ผู้ชายมีโอกาสเกิดโรคสูงกว่าผู้หญิง
- อาชีพ คนว่างงานหรือทำงานบ้านโอกาสเกิดสูงกว่า
- ถิ่นที่พักอาศัย คนกรุงเทพฯเกิดโรคสูงกว่าคนภาคอื่นๆ
- โรคความดันโลหิตสูง
- โรคเบาหวาน
- โรคไขมันในเลือดสูง

อนึ่ง ในประเทศที่พัฒนาแล้วปัจจัยเสี่ยงนอกจากโรคความดันโลหิตสูง เบาหวาน และไขมันในเลือดสูงเช่นเดียวกับในคนไทยแล้ว ยังได้แก่ การสูบบุหรี่ โรคหลอดเลือดหัวใจ โรค หัวใจเต้นเร็ว เออเอฟ โรคหลอดเลือดแดงแข็ง โรคหลอดเลือดสมองโป่งพอง เคยมีอาการอัม พาด อัมพฤกษ์ ชั่วคราว (สมองขาดเลือดชั่วคราว) มาแล้ว และอาจจากพันธุกรรม เพราะพบโรคได้สูงขึ้นในคนมีประวัติครอบครัวเป็นโรคนี้

2.10.3 โรคอัมพาตโรคอัมพฤกษ์มีอาการอย่างไร?

อาการจากโรคอัมพาตโรคอัมพฤกษ์เป็นอาการที่เกิดขึ้นที่ เป็นนาทีหรือชั่วโมง โดยอาการขึ้นกับตำแหน่งสมองส่วนที่ขาดเลือด ซึ่งที่พบบ่อยได้แก่

- แขน ขา ด้านเดียวกันอ่อนแรงทันที
- แขน ขาด้านเดียวกัน ใบหน้า ขา
- ลิ้นสั่น พูดไม่ชัด หรือพูดไม่ได้
- อาจมีตาพร่า มองเห็นภาพไม่ชัด หรือเห็นภาพเพียงบางส่วน หรือเห็นภาพได้แคบลง
- อาจหายใจเหนื่อยหอบ ตัดขาด/หายใจลำบาก
- มึนงง วิงเวียน ทรงตัวไม่ได้
- ปวดศีรษะรุนแรงทันที
- ถ้าเกิดจากมีเลือดออกในสมอง ความดันในสมองมักขึ้นสูงส่งผลให้เกิดคลื่นไส้อาเจียน
- เมื่อมีอาการมาก อาจหมดสติ โคม่า และเสียชีวิต (ตาย) ในที่สุด

อาการสำคัญที่สุดที่ช่วยการวินิจฉัยโรคอัมพาตโรคอัมพฤกษ์ประกอบด้วย 4 อาการหลักที่เรียกว่า ฟาต (FAST) คือ

1. Facial weakness (ใบหน้าเบี้ยว ปากเบี้ยว)
2. Arm weakness (แขนอ่อนแรง ไม่มีแรง)
3. Speech difficult (พูดไม่ชัด พูดไม่ได้) และ
4. Time to act (ทุกอาการเกิดพร้อมกันทันที)

ซึ่งเมื่อมีอาการดังกล่าวให้รีบไปโรงพยาบาลทันที โทรศัพท์เรียกรถพยาบาลฉุกเฉินเบอร์เดียวทั่วประเทศไทยคือ “โทรฯ 1669” สถาบันการแพทย์ฉุกเฉินแห่งชาติ กระทรวงสาธารณสุข (สพฉ.) ฟรีตลอด 24 ชั่วโมง

2.10.4 การรักษา

การรักษาให้ได้ผลดีขึ้นอยู่กับ

1. เวลา ยิ่งได้รับการรักษาเร็วเท่าไร จะยังมีโอกาสหายเป็นปกติได้มาก
2. ความรุนแรงของโรคที่เป็น ผู้ป่วยที่อาการรุนแรงน้อยจะมีโอกาสหายได้สูงกว่า
3. ความพร้อมของเทคโนโลยีในการรักษา โดยใช้อุปกรณ์หรือเทคนิคที่เหมาะสม และยาที่มีประสิทธิภาพ ตลอดจนแพทย์ผู้เชี่ยวชาญ เป็นปัจจัยที่สำคัญของผลการรักษา

สำหรับการรักษาโรคหลอดเลือดสมอง ที่เกิดจากการตีบหรืออุดตันของหลอดเลือดในระยะเฉียบพลันที่มีการศึกษายืนยันแล้วว่า ได้ผลดีชัดเจน ได้แก่

1. การให้ยาสลายลิ่มเลือด (Tissue plasminogen activator, tPA) ทางหลอดเลือดดำแก่ผู้ป่วยโรคหลอดเลือดสมองตีบหรืออุดตันภายในเวลา 3 ชั่วโมงหลังเกิดอาการ จะเพิ่มโอกาสของการฟื้นตัวจากภาวะพิการให้อาการกลับมาใกล้เคียงปกติได้ถึง 1.5 – 3 เท่า เมื่อเทียบกับผู้ป่วยกลุ่มที่ไม่ได้รับยา อย่างไรก็ตามการใช้นี้มีความเสี่ยงของเลือดออกในสมองได้ประมาณ 6%

2. การให้รับประทานยาแอสไพรินอย่างน้อย 160 mg ต่อวันภายใน 48 ชั่วโมงหลังเกิดอาการ จะช่วยลดโอกาสเกิดโรคหลอดเลือดสมองตีบตันซ้ำและเสียชีวิตลง

3. การรับตัวผู้ป่วยไว้ในหอผู้ป่วยโรคหลอดเลือดสมองเฉียบพลัน (acute stroke unit) นับเป็นการรักษาที่ช่วยลดอัตราการตายหรือพิการอีกวิธีหนึ่ง

4. การผ่าตัดเปิดกะโหลก (Hemicraniectomy) จะพิจารณาทำเฉพาะกรณีที่มีอาการรุนแรงและมีการตีบหรืออุดตันของหลอดเลือดแดงใหญ่ Middle cerebral artery ในสมองเท่านั้น โดยมีหลักฐานการศึกษาว่าการผ่าตัดดังกล่าวสามารถลดอัตราการตายของผู้ป่วยได้

นอกจากนี้การควบคุมความดัน การป้องกันภาวะแทรกซ้อน การทำกายภาพฟื้นฟู การที่คนในครอบครัวร่วมมือกันในการดูแลยามที่ผู้ป่วยห้อย การให้กำลังใจผู้ป่วย ล้วนเป็นปัจจัยสำคัญที่มีผลต่อการฟื้นตัวของผู้ป่วยทั้งสิ้น

2.10.5 ดูแลอย่างไรไม่เป็นอัมพฤกษ์อัมพาต

เรื่องการดูแล มีความสำคัญมากในการป้องกันไม่ให้เกิดอัมพฤกษ์อัมพาต โดยมีข้อปฏิบัติดังนี้

1. การรักษาและควบคุมปัจจัยเสี่ยง ได้แก่การรักษาเบาหวาน ความดันโลหิตสูง การงดสูบบุหรี่
2. การป้องกันการกลับเป็นซ้ำด้วยการใช้ยาต้านเกล็ดเลือดหรือยาป้องกันการแข็งตัวของเลือด
3. การลดอาหารไขมัน โดยเฉพาะไขมันอิ่มตัว อาหารเค็ม กินผักและผลไม้ให้มาก
4. จำกัดการดื่มสุรา เบียร์
5. รักษาน้ำหนักให้อยู่ในเกณฑ์ปกติ
6. ออกกำลังกายให้สม่ำเสมอ

ได้มีการคาดการณ์ว่า มีผู้ป่วยใหม่ที่เป็นโรคอัมพาตเกิดขึ้นในประเทศไทยปีละ 150,000 ราย ถ้าการรณรงค์ป้องกันได้ผลจะทำให้ผู้ป่วยไม่ต่ำกว่า 50,000 ราย ปลอดภัยจากโรค ซึ่งค่าใช้จ่ายที่ใช้รักษาสำหรับผู้ป่วยโรคนี้ต่อคน คิดเป็นเงินประมาณ 100,000 บาท ต่อปี ดังนั้นการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ป้องกันอัมพาตอย่างต่อเนื่อง และจริงจิงจะสามารถประหยัดเงินประเทศชาติได้ถึงปีละห้าพันล้านบาท

2.11 โรคกล้ามเนื้ออ่อนแรง

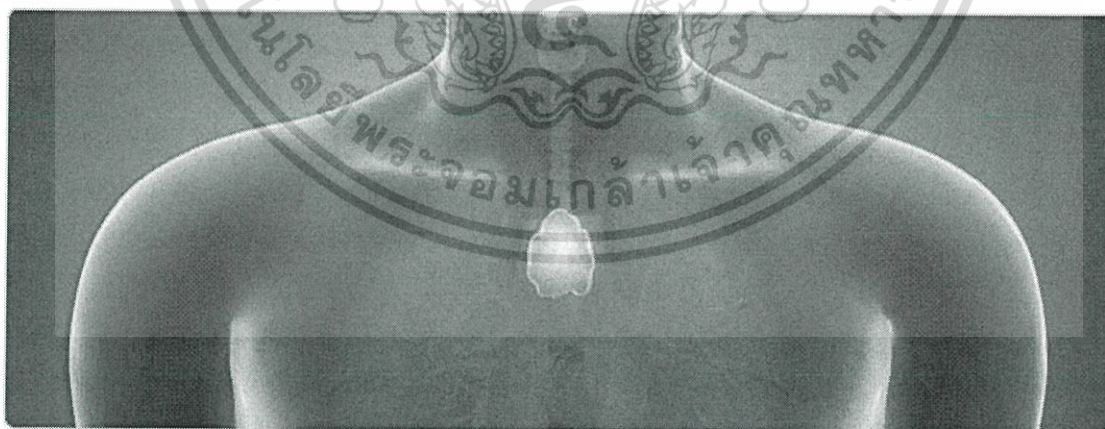
2.11.1. โรคกล้ามเนื้ออ่อนแรงเอ็มจี (Myasthenia gravis หรือเรียกย่อว่า MG)

โรคกล้ามเนื้ออ่อนแรงเอ็มจี (Myasthenia gravis หรือเรียกย่อว่า MG) โดยคำว่า Myasthenia มาจากภาษากรีก แปลว่า กล้ามเนื้อ Gravis มาจากภาษาลาติน แปลว่า เลวร้าย หรือรุนแรง ซึ่งต่อไปขอเรียกโรคนี้ว่า “โรคเอ็มจี”

โรคเอ็มจี เป็นโรค ออโตอิมมูน/โรคภูมิต้านตนเอง (Autoimmune) ชนิดเรื้อรังชนิดหนึ่ง ที่ส่งผลให้กล้ามเนื้อลาย (กล้ามเนื้อที่อยู่ในการควบคุมของสมอง/Voluntary muscle หรือ Striated muscle) ซึ่งคือ กล้ามเนื้อภายนอกร่างกาย ที่ร่างกายใช้ในการเคลื่อนไหวต่างๆ เช่น กล้ามเนื้อ แขน ขา ดวงตา ใบหน้า ช่องปาก กล่องเสียง และกล้ามเนื้อซี่โครงที่ใช้ในการหายใจ เกิดการอ่อนแรงจนไม่สามารถทำงานได้ตามปกติ ทั้งนี้อาการมักเกิดหลังจากที่กล้ามเนื้อมัดนั้นๆ ทำงานซ้ำๆ แต่หากเมื่อได้รับการพักการใช้งานของกล้ามเนื้อมัดนั้นๆ อาการกล้ามเนื้ออ่อนแรงจะดีขึ้น และกล้ามเนื้อจะฟื้นตัวกลับมาทำงานได้เป็นปกติ ทั้งนี้ การอ่อนแรงของกล้ามเนื้อนั้นๆ ไม่ได้เกิดจากโรคของสมอง หรือของไขสันหลังอักเสบ บาดเจ็บ หรือจากโรคหลอดเลือดสมอง ซึ่งเรียกว่าโรคอัมพฤกษ์ หรือ โรคอัมพาต หรือจากโรคของเส้นประสาท ดังนั้น โรคเอ็มจี จึงไม่ใช่โรคอัมพฤกษ์/อัมพาต

โรคเอ็มจี เป็นโรคพบได้ไม่บ่อยนัก ประมาณ 10 ราย ต่อประชากร 100,000 คน พบได้ในทุกอายุ ตั้งแต่เด็กแรกเกิดจนถึงผู้สูงอายุ โดยพบในผู้หญิงมากกว่าในผู้ชายประมาณ 3:2 เท่า ทั้งนี้พบโรคนี้ในเด็กได้ประมาณ 10% ของผู้ป่วยด้วยโรคนี้ทั้งหมด ในผู้ใหญ่ผู้หญิง มักพบโรคได้สูงในช่วงอายุ 20-40 ปี แต่ในผู้ใหญ่ผู้ชาย มักพบโรคได้สูงในช่วงอายุตั้งแต่ 50 ปีขึ้นไป

2.11.1.1 โรคเอ็มจีมีกลไกการเกิดอย่างไร?



รูปที่ 2.28 ต่อมไทมัส

ในการทำงานปกติของกล้ามเนื้อ จะเกิดจากเส้นประสาท (ส่งงานผ่านมาจากสมอง) ส่งการมายังกล้ามเนื้อโดยการส่งการจะถูกส่งมายังตัวรับคำสั่ง (Receptor) ซึ่งเชื่อมต่อระหว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เส้นประสาทกับกล้ามเนื้อ (Neuromuscular junction) ซึ่งตัวรับคำสั่งนี้เรียกว่า Acetylcholine receptor (AChR/เอซีเอชอาร์) ซึ่งเมื่อได้รับคำสั่งแล้ว กล้ามเนื้อก็จะหดตัวทำงานตามปกติ

ในโรคเอ็มจี ซึ่งเป็นโรคออโตอิมมูน/ภูมิต้านตนเอง (Autoimmune disease) ชนิดเรื้อรังที่ร่างกายสร้างสารภูมิต้านทาน (Antibody) ชนิด ไอจีจี (IgG, Immunoglobulin G) ซึ่งสารนี้จะไปสกัดกั้น หรือทำลายตัวรับคำสั่ง เอซีเอชอาร์ จึงส่งผลให้คำสั่งในการทำงานไปไม่ถึงกล้ามเนื้อหรือลดน้อยกว่าปกติ จึงส่งผลให้เกิดการอ่อนแรงของกล้ามเนื้อ/กล้ามเนื้ออ่อนแรงในระดับต่างๆตามความรุนแรงของโรค ตั้งแต่ น้อยๆที่อาการอ่อนแรงไม่ชัดเจน ไปจนถึงกล้ามเนื้อหมดประสิทธิภาพในการหดตัวโดยสิ้นเชิง

ทั้งนี้ อวัยวะที่แพทย์เชื่อว่าเป็นตัวก่อให้เกิดการสร้างสารภูมิต้านทานผิดปกติตัวนี้คือ ต่อมไทมัส (Thymus gland)

ต่อมไทมัส เป็นต่อมที่มีหน้าที่เกี่ยวข้องกับการสร้างภูมิคุ้มกันต้านทานโรคของร่างกาย (Immune system) เป็นต่อมที่อยู่ในช่องอกตอนบน ต่อมอยู่ใต้กระดูกอก (Sternum) โดยวางอยู่บนด้านหน้าของหัวใจ ต่อมไทมัสจะมีการเจริญเติบโตและทำงานเต็มที่ตั้งแต่แรกเกิดไปจนถึงช่วงวัยรุ่น หลังจากนั้นจะค่อยๆฝ่อตัวลงไปเรื่อยๆ โดยมีเนื้อเยื่อพังผืดและเนื้อเยื่อไขมัน (เนื้อเยื่อในกลุ่ม เนื้อเยื่อเกี่ยวพัน) เข้ามาแทนที่ตามอายุที่มากขึ้น จากแรกเกิดที่ต่อมมีน้ำหนักประมาณ 15 กรัม และในวัยรุ่นหนักประมาณ 35 กรัม เป็นหนักประมาณ 25 กรัมเมื่ออายุประมาณ 25 ปี หนักน้อยกว่า 15 กรัมเมื่ออายุประมาณ 60 ปี และลดลงเหลือเพียงประมาณ 5 กรัมเมื่ออายุประมาณ 70 ปี ทั้งนี้เมื่อแรกเกิด ต่อมไทมัสจะมีขนาดประมาณ กว้าง 4 เซนติเมตร ยาว 5 เซนติเมตร และหนา 6 มิลลิเมตร

2.11.1.2 โรคเอ็มจี มีสาเหตุหรือปัจจัยเสี่ยงจากอะไร?

ในโรคเอ็มจี ถึงแม้จะรู้ว่า โรคมีกลไกเกิดได้อย่างไรตั้งแต่กล่าวแล้วในหัวข้อกลไกการเกิดโรค แต่ปัจจุบันก็ยังไม่ทราบสาเหตุว่า อะไรเป็นสาเหตุให้ร่างกาย หรือต่อมไทมัสสร้างสารภูมิต้านทานชนิดก่อโรคเอ็มจี อย่างไรก็ตาม พบว่าโรคเอ็มจี มีความสัมพันธ์กับโรคของต่อมไทมัส โดยประมาณ 85 % พบเกิดร่วมกับมีโรคเซลล์ต่อมไทมัสเจริญเกินปกติ (Thymus hyperplasia) และประมาณ 10-15% เกิดร่วมกับโรคเนื้องอกต่อมไทมัส (Thymoma) นอกจากนี้ มีรายงานพบว่า พบโรคเอ็มจีเกิดร่วมกับโรคมะเร็งปอดชนิดเซลล์ตัวเล็ก และโรคมะเร็งต่อมน้ำเหลืองฮอดจ์กิน ได้บ้าง

2.11.1.3 โรคเอ็มจีมีอาการอย่างไร?

อาการสำคัญของโรคเอ็มจี คือ มีอาการเหนื่อย อ่อนล้า กล้ามเนื้ออ่อนแรง และจะอ่อนแรงเพิ่มมากขึ้นเมื่อออกแรงมากขึ้น แต่อาการจะดีขึ้นเมื่อกำลังกล้ามเนื้อหยุดพักการออกแรง นอกจากนั้น อาจมีอาการอื่นๆร่วมด้วย โดยจะขึ้นกับว่า โรคเกิดกับกล้ามเนื้อส่วนไหนของร่างกาย ทั้งนี้ ประมาณ 85% ของผู้ป่วยจะมีอาการกล้ามเนื้ออ่อนแรงในทุกมัดของกล้ามเนื้อลาย เมื่อเกิดกับกล้ามเนื้อดวงตา มักมีอาการหนังตาตก อาจเกิดข้างเดียว (ซ้าย หรือขวามีโอกาสเกิดได้เท่ากัน) หรือตกทั้งสองตา และมักมองเห็นภาพซ้อน ซึ่งกล้ามเนื้อดวงตาเป็นกล้ามเนื้อที่พบมีอาการได้บ่อยที่สุด มักเกิดตั้งแต่ระยะแรกของโรค และเป็นอาการที่มักนำผู้ป่วยมาพบแพทย์ ทั้งนี้ประมาณ 15% ของผู้ป่วยโรคเอ็มจี จะมีอาการเกิดกับเฉพาะกล้ามเนื้อดวงตา

เมื่อเกิดกับกล้ามเนื้อช่องปาก จะมีอาการ กลืนลำบาก พูดไม่ชัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเกิดกับกล้ามเนื้อแขนขา แขน ขาจะอ่อนแรง
เมื่อเกิดกับกล้ามเนื้อกระดูกซี่โครง ซึ่งเป็นอาการรุนแรงที่สุด ผู้ป่วยจะหายใจไม่ได้
ร่างกายจะขาดอากาศ เป็นอาการที่ต้องไปโรงพยาบาลฉุกเฉิน

หนึ่ง ประมาณ 10% ของโรคเอมจี จะมีโรคของต่อมไทรอยด์ร่วมด้วย ผู้ป่วยจึงอาจมีอาการ
จากโรคของต่อมไทรอยด์ร่วมด้วยได้ อาจจากโรคต่อมไทรอยด์เป็นพิษ หรือจากภาวะขาดไทรอยด์
ฮอร์โมน

2.11.1.4 ปัจจัยที่อาจทำให้โรคเอมจีมีอาการรุนแรงขึ้น

ได้แก่ ผลข้างเคียงจากยาบางชนิด เช่น ยาปฏิชีวนะบางชนิด ยาทางจิตเวชบางชนิด
ยาลดความดันโลหิตบางชนิด เป็นต้น

- ภาวะติดเชื้อ
- ความเครียด
- การดื่มแอลกอฮอล์
- อากาศร้อน และการถูกแสงแดดจัด
- การตั้งครรภ์

2.11.1.5 แพทย์วินิจฉัยโรคเอมจีได้อย่างไร?

แพทย์วินิจฉัยโรคเอมจีได้จาก ประวัติอาการ ประวัติการเจ็บป่วยทั้งในอดีตและ
ปัจจุบัน ประวัติกินยาต่างๆ การตรวจร่างกาย และการตรวจวิธีเฉพาะต่างๆทางโรคเอมจี เช่น การ
ตรวจเบื้องต้นที่เรียกว่า Ice pack test (การวางน้ำแข็งบนหนังตาด้านเกิดอาการในขณะหลับตา
ประมาณ 2 นาที เมื่อลืมตาอีกครั้ง หนังตาที่ตกจะดีขึ้น หรือกลับเป็นปกติ) การตรวจเลือดดูค่าสารภูมิ
ต้านทานต่างๆต่อโรคนี้ (เช่น ภูมิต้านทานต่อตัวรับ Acetylcholine receptor) การตรวจการทำงานของ
ของกล้ามเนื้อ (Electrodiagnosis) และการตรวจภาพต่อมไทมัสด้วยเอกซเรย์คอมพิวเตอร์

2.11.1.6 รักษาโรคเอมจีอย่างไร?

แนวทางการรักษาโรคเอมจี ได้แก่ การให้ยาต้านการทำงานของเอนไซม์ที่ทำให้
กล้ามเนื้ออ่อนแรง (Acetylcholinesterase inhibitors) และยากดภูมิคุ้มกันต้านทานของร่างกาย
เช่น ยาในกลุ่มสเตียรอยด์ หรือในรายที่มีอาการรุนแรง อาจให้การรักษาโดยวิธีการกรองสารภูมิต้านทาน
ออกจากเลือด (Plasmapheresis) หรือการผ่าตัดต่อมไทมัสเมื่อตรวจพบมีต่อมไทมัสโต หรือบางครั้ง
เมื่อผู้ป่วยไม่สามารถผ่าตัดต่อมไทมัสได้จากสุขภาพไม่อำนวย และ/หรือการรักษาด้วยวิธีต่างๆ
ดังกล่าวแล้วไม่ได้ผล อาจใช้การฉายรังสีรักษาต่อมไทมัสเพื่อให้ต่อมไทมัสฝ่อลง เพื่อลดการทำงานของ
ของต่อมไทมัส

2.11.1.7 โรคเอมจีรุนแรงไหม? มีผลข้างเคียงไหม?

โรคเอมจี เป็นโรคเรื้อรังที่รักษาไม่หาย ต้องกินยาตลอดชีวิต หรือกินๆหยุดๆในช่วงที่
โรคสงบ (ขึ้นกับคำแนะนำของแพทย์) แต่เมื่อได้รับการรักษาตั้งแต่แรก โรคจะไม่รุนแรง รักษาเป็น
แบบผู้ป่วยนอกได้เสมอ และผู้ป่วยมักสามารถใช้ชีวิตปกติได้ด้วยการกินยาควบคุม และสามารถมีชีวิตร
ยืนยาวได้เท่ากับคนทั่วไป แต่ในผู้ป่วยบางราย (พบได้น้อย) ที่มีอาการรุนแรง เรียกว่า Myasthenic
เอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

crisis จะส่งผลให้กล้ามเนื้อช่วยการหายใจไม่ทำงาน ผู้ป่วยจำเป็นต้องไปโรงพยาบาลฉุกเฉินและได้รับการรักษาเป็นผู้ป่วยใน อาจต้องเจาะคอใส่ท่อช่วยหายใจ และใช้เครื่องช่วยหายใจ รวมทั้งการรักษาด้วยยา และการกรองสารภูมิต้านทานออกจากเลือด กรณีนี้ มีโอกาสเสียชีวิตได้ประมาณ 2-5% โดยมักพบในผู้สูงอายุ จากภาวะระบบทางเดินหายใจล้มเหลว (ภาวะหายใจล้มเหลว)

2.11.1.8 ดูแลตนเองอย่างไร? ควรพบแพทย์เมื่อไร?

การดูแลตนเองและการพบแพทย์เมื่อเป็นโรคเอดส์ ได้แก่

- กินยาตามแพทย์แนะนำให้ครบถ้วน ถูกต้อง ไม่ขาดยา
- ใช้ชีวิตประจำวันในการออกแรงให้สม่ำเสมอ เหมือนๆกันในทุกๆวันเพื่อแพทย์จะได้จัดปริมาณยา (Dose) ที่กินได้อย่างถูกต้อง
 - กินอาหารค่าน้อยๆ เป็นอาหารอ่อน (ประเภทอาหารทางการแพทย์) เพื่อลดการทำงานของกล้ามเนื้อช่องปาก จะได้เคี้ยวอาหารได้ ไม่ลำบาก ระหว่างกิน
 - มีที่ยึดจับในบ้าน เพื่อช่วยในการลุก นั่ง ยืน เดิน ร่วมกับจัดบ้านให้ปลอดภัย ง่ายแก่การใช้ชีวิตที่ไม่ต้องออกแรงมาก รวมทั้งเพื่อการใช้ชีวิตประจำวัน เช่น ใช้ประตูปั่นไฟฟ้า ใช้เครื่องซักผ้า ใช้เครื่องปั่นแทนการตักน้ำพริก เป็นต้น
 - เมื่อออกนอกบ้านต้องวางแผนล่วงหน้า ไม่รีบร้อน ไม่ออกแรงมากเกินไป
 - เมื่อเห็นภาพซ้อน ควรปิดตาข้างที่เกิดอาการ จะช่วยให้การมองเห็นดีขึ้น ลดโอกาสเกิดอุบัติเหตุ
 - มีป้ายติดตัวเสมอว่าเป็นโรคอะไร กินยาอะไร รักษาโรงพยาบาลไหน เลขประจำตัวโรงพยาบาล คืออะไร เพื่อมีอาการฉุกเฉิน คนจะได้ช่วยได้ถูกต้องรวดเร็ว
 - หลีกเลี่ยงปัจจัยเสี่ยงที่กระตุ้นให้อาการรุนแรง ดังกล่าวแล้วในหัวข้อ ปัจจัยเสี่ยงรักษาสุขภาพพื้นฐาน (สุขบัญญัติแห่งชาติ) เพื่อสุขภาพกาย สุขภาพจิตที่แข็งแรง ลดปัจจัยเสี่ยงที่จะทำให้มีอาการรุนแรง
 - กินอาหารมีประโยชน์ 5 หมู่ให้ครบถ้วนในทุกวัน ในปริมาณที่ไม่ทำให้เกิดโรคอ้วนและน้ำหนักตัวเกิน เพื่อลดการแบกน้ำหนักของกล้ามเนื้อและเพื่อความแข็งแรงของกล้ามเนื้อ
 - ก่อนการตั้งครรภ์ ต้องปรึกษาแพทย์ก่อนเสมอ เพื่อการปรับยา และเพื่อการดูแลตนเองและทารกในครรภ์
 - พบแพทย์ตามนัดเสมอ และรีบพบก่อนนัดเมื่อมีอาการต่างๆเลวลง หรือผิดไปจากเดิม หรือเมื่อกังวลในอาการ
 - รีบไปโรงพยาบาลฉุกเฉินเมื่อมีอาการทางการแพทย์ เช่น หายใจไม่ออก/หายใจลำบาก

2.11.1.9 ป้องกันโรคเอดส์ได้อย่างไร?

เนื่องจากโรคเอดส์ยังเป็นโรคที่ไม่รู้สาเหตุ และโรคที่สัมพันธ์กันก็ยังเป็นโรคที่ส่วนใหญ่ไม่รู้สาเหตุ เช่น โรคของต่อมไทมัส และโรคของต่อมไทรอยด์ ดังนั้น ปัจจุบัน จึงยังไม่มีวิธีป้องกันโรคเอดส์

อย่างไรก็ตาม โรคเอ็มจี เป็นโรคที่รักษาและควบคุมได้ผลสูง ดังนั้นเมื่อมีอาการดังกล่าวแล้วในหัวข้ออาการ โดยเฉพาะเมื่อมีหนังตาคก จึงควรรีบพบแพทย์เสมอ เพื่อการวินิจฉัยและการรักษาที่รวดเร็วเหมาะสม ซึ่งจะส่งผลให้การรักษาควบคุมโรคได้ผลดีจนสามารถใช้ชีวิตประจำวันได้อย่างปกติ

2.11.2. โรคกล้ามเนื้ออ่อนแรงเอแอลเอส (Amyotrophic lateral sclerosis หรือเรียกว่า ALS)

โรคเอแอลเอส (ALS) หรือโรคกล้ามเนื้ออ่อนแรงเอแอลเอส ย่อมาจาก Amyotrophic lateral sclerosis เป็นโรคที่เกิดจากการเสื่อมของเซลล์ประสาทสั่งการ (Motor neuron disease : MND) ไม่ใช่โรคกล้ามเนื้อโดยตรง แต่เนื่องจากเซลล์ประสาทนำคำสั่งในไขสันหลังส่วนหน้า และมีส่วนของเนื้อสมองเสื่อม จึงสูญเสียการนำคำสั่งในการทำงานมายังกล้ามเนื้อ จึงก่อให้เกิดอาการผิดปกติของกล้ามเนื้อขึ้น

2.11.2.1 โรคเอแอลเอสเกิดขึ้นได้อย่างไร?



รูปที่ 2.29 โรคเอแอลเอส

ปัจจุบันยังไม่ทราบแน่ชัดว่า เกิดการเสื่อมของเซลล์นำคำสั่งได้อย่างไร เชื่อว่าอาจเกิดจากหลายๆปัจจัย เช่น สารพิษ ยาฆ่าแมลง โลหะหนัก สารรังสี การติดเชื้อไวรัส หรืออาจเกิดจากการเสื่อมของเซลล์เอง (อาจจากอายุ และ/หรือ มีจีน/ยีน/Gene บางชนิดผิดปกติ)

2.11.2.2 ใครมีปัจจัยเสี่ยงเป็นโรคเอแอลเอสบ้าง?

โรคเอแอลเอสนี้ พบได้น้อยมาก ในประเทศไทยไม่มีข้อมูลว่าพบมากน้อยเพียงใด ในยุโรปพบประมาณ 2 รายต่อประชากร 100,000 คน ส่วนใหญ่เป็นผู้สูงอายุมากกว่า 60 หรือ 65 ปีขึ้นไป ผู้ชายพบบ่อยกว่าผู้หญิง ไม่พบการถ่ายทอดทางพันธุกรรมที่แน่ชัด

2.11.2.3 โรคเอแอลเอสมีอาการอย่างไร?

ผู้ป่วยโรคเอแอลเอส จะมีอาการกล้ามเนื้ออ่อนแรงบริเวณ แขน มือ ขา ข้างหนึ่งข้างใดก่อน โดยอาการค่อยๆเป็นค่อยๆไป ต่อมามีอาการรุนแรงมากขึ้น ร่วมกับมีอาการ พูดไม่ชัด กลืน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อาหารลำบาก กล้ามเนื้อต่างๆลีบ และมีอาการเต้นกระตุกของกล้ามเนื้อด้วย ต่อมาเป็นทั้ง 2 ข้างของร่างกาย อาการในช่วงแรกๆจะไม่ค่อยชัดเจน ผู้ป่วยอาจบ่นว่าอ่อนเพลีย ไม่ค่อยมีแรงเดิน หรือหิบบีบจับของไม้ถนัด ร่วมกับมีกล้ามเนื้อเต้นกระตุกด้วย การดำเนินโรคจะใช้เวลาหลายเดือนถึงเป็นปี จึงมีอาการชัดเจน ทั้งนี้ผู้ป่วยจะไม่มีอาการผิดปกติด้านความจำ ระดับความรู้สึก ตัวใดๆ รวมทั้งในการขับถ่าย แม้จะเป็นในระยะสุดท้ายของโรคก็ตาม

2.11.2.4 โรคเอแอลเอสวินิจฉัยได้อย่างไร?

แพทย์วินิจฉัยโรคเอแอลเอสได้โดย จะพิจารณาจากลักษณะอาการผู้ป่วย คือ กล้ามเนื้ออ่อนแรง โดยการรับรู้ความรู้สึกเป็นปกติ กล้ามเนื้อลีบฝ่อ ไม่ปวด ไม่ชา มีการเต้นกระตุกของกล้ามเนื้อ พุดไม่ชัด กล้ามเนื้อลึบลิบ และมีการเต้นของกล้ามเนื้อลิ้น

ตรวจร่างกายพบกล้ามเนื้อลีบฝ่อทั่วไป ลิ้นลีบ มีการเต้นกระตุกของลิ้น และพบรีเฟล็กซ์ผิดปกติ ตรวจการทำงานของกล้ามเนื้อด้วยกระแสไฟฟ้า (Electromyography) พบความผิดปกติที่เข้าได้กับโรคนี้

นอกจากนั้นคือ ต้องตรวจประเมินไม่พบโรคอื่นๆที่ให้อาการคล้ายกัน และ โรคนี้ไม่จำเป็น ต้องเอกซเรย์คอมพิวเตอร์ หรือเอ็มอาร์ไอสมอง ซึ่งการตรวจสปีคันเพิ่มเติมเหล่านี้ขึ้นกับดุลพินิจของแพทย์ผู้รักษา

โรคที่มีอาการคล้ายกับโรคเอแอลเอส เช่น

- โรคไทรอยด์เป็นพิษ
- โรคซีฟิลิส ระยะเข้าไขสันหลัง และ/หรือเข้าสมอง
- โรคกระตุกสันหลังเสื่อม ในบริเวณกระตุกสันหลังส่วนคอ

2.11.2.4 รักษาโรคเอแอลเอสอย่างไร?

ปัจจุบัน การรักษาและการพยากรณ์โรคของโรคเอแอลเอส คือ โรคเอแอลเอส ยังไม่มียา หรือวิธีใดๆที่จะรักษาให้หายได้ มีเพียงยาชื่อ ริลูซอล (Riluzole) ที่พอจะชะลออาการของโรคได้บ้าง โดยสามารถชะลอภาวะหายใจล้มเหลวได้ระยะเวลาหนึ่งเมื่อเทียบกับไม่ใช้ยา แต่สุดท้ายผู้ป่วยก็จะมีอาการรุนแรงขึ้น หายใจเองไม่ไหว ต้องใส่ท่อและเครื่องช่วยหายใจตลอดไป โดยส่วนใหญ่ อาการจะค่อยๆเลวลงหลังจากวินิจฉัยได้แล้วประมาณ 2-5 ปี ก็จะมีภาวะหายใจล้มเหลว และเป็นสาเหตุของการเสียชีวิตจากระบบหายใจล้มเหลว และการติดเชื้อในระบบต่าง ๆ โดยเฉพาะในทางเดินหายใจ ทั้งนี้ผู้ป่วยจะค่อยๆไม่สามารถเคลื่อนไหวได้ เพราะกล้ามเนื้อทุกส่วนจะอ่อนแรง และฝ่อลีบไปเรื่อยๆจนไม่มีแรงเลย ยกเว้นที่พอทำงานได้ คือ การกลอกตาไปมา และการหลับตา สัมผัส

การรักษาที่ดีที่สุดคือ การรักษาประคับประคองตามอาการ ร่วมกับการทำกายภาพบำบัดเพื่อให้กล้ามเนื้อแข็งแรง ชะลอการฝ่อลีบให้ช้าลง และเป็นการป้องกันภาวะแทรกซ้อน/ผลข้างเคียงที่จะเกิดขึ้น เช่น แผลกดทับ และการสำลักอาหารและน้ำลาย ที่จะส่งผลถึงการติดเชื้อในระบบทางเดินหายใจ เป็นต้น

ผู้ป่วยที่มีภาวะหายใจล้มเหลว ก็อาจต้องเข้ารับการรักษาในโรงพยาบาล เพราะต้องใส่ท่อช่วยหายใจและใช้เครื่องช่วยหายใจ ซึ่งการดูแลในโรงพยาบาลคือ การดูแลการใช้เครื่องช่วยหายใจ และการป้องกันการติดเชื้อ ร่วมกับการทำกายภาพบำบัด ดังนั้นถ้าทางครอบครัวมีความพร้อมก็สามารถให้การดูแลที่บ้านได้ ซึ่งจะปลอดภัยจากภาวะติดเชื้อในโรงพยาบาล แต่ค่าใช้จ่ายจะสูงมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และต้องลงทุนซื้อเครื่องช่วยหายใจ และติดตั้งระบบออกซิเจนที่บ้าน ปัจจุบันก็มีบางครอบครัวที่สามารถทำแบบนี้ได้

2.11.2.5 ครอบครัวและผู้ป่วยควรทำอะไรเมื่อทราบว่าเป็นโรคนี?

เมื่อเป็นโรคนี กาลังใจเป็นสิ่งสำคัญที่สุด ทุกคนต้องมีกำลังใจที่ดี ยอมรับความเป็นจริง ปรับตัวอยู่กับโรคให้ได้ ไม่ควรท้อแท้ หรือหมดกำลังใจ ในระยะแรกของโรค การออกกำลังกายสม่ำเสมอเพื่อให้กล้ามเนื้อแข็งแรง ก็เป็นการช่วยชะลอโรคได้

การดูแลรักษาที่บ้านนั้น ในระยะแรกสามารถทำได้ แต่ถ้าเกิดภาวะหายใจล้มเหลวแล้ว การดูแลที่บ้าน ทางครอบครัวต้องมีความพร้อมทุกอย่างดังที่กล่าวแล้วข้างต้น รวมทั้งการให้อาหารทางสายยาง ไม่ว่าจะผ่านทางจมูกหรือทางหน้าท้อง การดูแลการขับถ่าย และการเตรียมญาติและผู้ดูแลตลอด 24 ชั่วโมง ซึ่งเมื่ออยู่บ้าน ผู้ป่วยจะมีความสุขมากกว่าอยู่โรงพยาบาล

เมื่อผู้ป่วยเกิดการสำลักอาหารบ่อยขึ้น มีไข้ หรือมีการติดเชื้อในระบบต่างๆ เช่น ในทางเดินหายใจ หรือในทางเดินปัสสาวะ หรือเมื่อมีปัญหาต่างๆในการดูแลผู้ป่วย ก็ควรต้องปรึกษาแพทย์ พยาบาลก่อนนัด ซึ่งเมื่อป่วยเป็นโรคนี ควรต้องสอบถามแพทย์/พยาบาลล่วงหน้าถึงวิธี การดูแลผู้ป่วยที่บ้าน การปรึกษาแพทย์/พยาบาลทางโทรศัพท์ หรือการมาโรงพยาบาลก่อนนัดว่า ควรทำอย่างไร จะได้เหมาะสมและสะดวกทั้งกับผู้ป่วยและผู้ดูแล

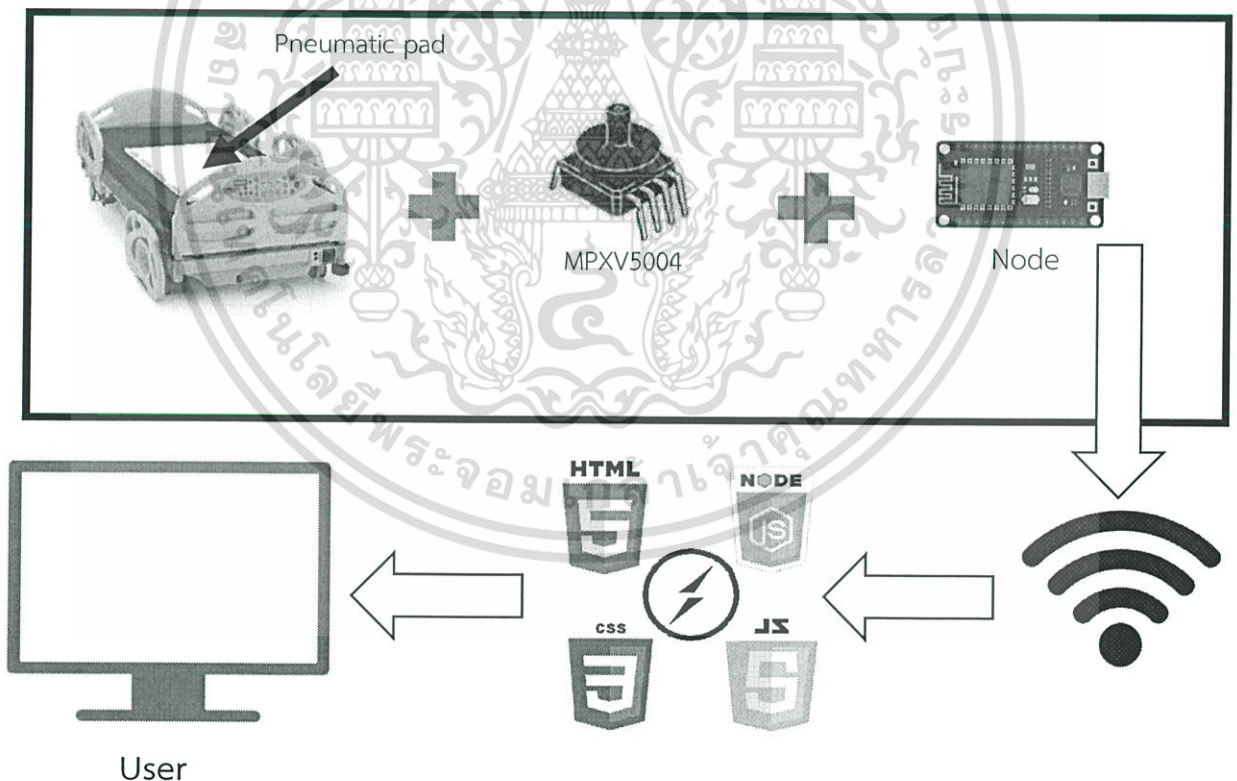


บทที่ 3

วิธีดำเนินการวิจัย

(RESEARCH METHODOLOGY)

การออกแบบระบบการตรวจจับสัญญาณว่าในขณะที่นั้นมีผู้ป่วยอยู่บนเตียงหรือไม่ ซึ่งแบ่งลักษณะการทำงานออกเป็น 2 ส่วน ได้แก่ ส่วนที่ทำหน้าที่เป็นตัวรับสัญญาณจากผู้ป่วยแล้วส่งไปยังเว็บเซิร์ฟเวอร์ และส่วนที่ทำหน้าที่แสดงผลข้อมูล ซึ่งสามารถแสดงเป็นไดอะแกรม ได้ดังภาพที่ 3.1 โดยการทำงานจะเริ่มจากในส่วนของตัวรับสัญญาณจากผู้ป่วยนั่นก็คือหมอนลมซึ่งจะมี Pressure Sensor ทำหน้าที่วัดแรงดันจากการนอนหรือขยับตัวของผู้ป่วย แล้วเก็บค่าข้อมูลที่ได้ไว้ในฐานข้อมูลเมื่อมีผู้เรียกใช้งานเว็บแอปพลิเคชัน ส่วนของเว็บเซิร์ฟเวอร์จะทำการรับส่งข้อมูลจากฐานข้อมูลมาทำการแสดงผลหรือส่งค่าข้อมูลแบบเรียลไทม์ผ่าน NodeMCU ไปยังหน้าเว็บแอปพลิเคชันเพื่อการแจ้งเตือนโดยเฉพาะ และจะส่งข้อมูลผ่านระบบสื่อสารแบบไร้สาย (Wi-Fi) ค่าที่ได้จะแสดงออกมาในรูปแบบของกราฟ ซึ่งสามารถนำค่าที่ได้จากกราฟมาดูสถานะของผู้ป่วยได้ว่าการขยับตัว เคลื่อนไหวอย่างไรบ้าง



รูปที่ 3.1 บล็อกไดอะแกรมระบบการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 การทำงานส่วนของเซิร์ฟเวอร์ (Server)

เซิร์ฟเวอร์ (Server) คือคอมพิวเตอร์ที่ลงโปรแกรมระบบปฏิบัติการหรือโปรแกรมอื่นๆที่ทำให้สามารถให้บริการข้อมูลแก่คอมพิวเตอร์เครื่องอื่นๆภายในระบบเครือข่ายได้ ด้วยความสามารถนี้จึงเหมาะสมที่จะนำมาใช้ในการพัฒนาระบบรักษาความปลอดภัยบนเตียงผู้ป่วยให้สามารถเข้าถึงข้อมูลได้ง่ายมากขึ้น

3.1.1 การทำ Real-Time web application

โดยก่อนที่เราจะทำ real-time web application เราจำเป็นจะต้องมีสิ่งต่างๆเหล่านี้

Node.js (<https://nodejs.org/en/>)

Socket.io (<http://socket.io/>)

Text editor (Notepad++)

3.1.1.1 Framework

3.1.1.1.1 Node.js เป็น JavaScript runtime ที่สร้างบน Chrome's V8 JavaScript engine พัฒนาโดย Google (ตามอ่านรายละเอียดได้ [ที่นี่](#)) การทำงานของ Node.js จะเป็นในลักษณะ event-driven และ non-blocking ซึ่งทำให้มันมีประสิทธิภาพและไม่หนักเครื่อง โดยมี library package มากมายที่ผู้พัฒนาจากทั่วโลกทำขึ้นมา support ในงานต่าง เราสามารถติดตั้งด้วยคำสั่ง npm และที่สำคัญคือเป็น open source

3.1.1.1.2. Socket.io เป็น JavaScript framework ที่ใช้ในการส่งข้อมูลแบบ real-time แบบ bidirectional และ เป็น event-based ที่ค่อนข้างเร็วและเสถียร แต่ Socket.io อาจจะไม่เหมาะสมกับการเชื่อมต่อหลายๆ connection (1,000+ connection) หากจำเป็นต้องใช้ WebSocket ในงานของคุณจริงๆ อาจต้อง config เพิ่มเติมหรือใช้งาน framework หรือ engine ตัวอื่นมา support

เมื่อโหลด Node.js ได้เราทำการ check ว่าสามารถใช้งานได้ไหมด้วยการพิมพ์คำว่า
 node -v จะเป็นการ check version

```

Node.js command prompt
Your environment has been set up for using Node.js 4.5.0 (ia32) and npm.
C:\Users\SUNDAY>node -v
v4.5.0
C:\Users\SUNDAY>
  
```

รูปที่ 3.2 การ check version

จากนั้นเราจะไปเรียกไฟล์งานของเราในไดร์ฟ D และ run ไฟล์งานของเรา
 ที่ชื่อว่า test2.js ถ้า run สำเร็จจะขึ้น port ที่เราเชื่อมต่อไว้ในที่นี้เราเขียน code ไว้ที่ port: 1337

```

Node.js command prompt - node test2.js
Your environment has been set up for using Node.js 4.5.0 (ia32) and npm.
C:\Users\SUNDAY>d:
D:\>cd My project
D:\My project>node test2.js
listening on *:1337
  
```

รูปที่ 3.3 ทดลอง run ไฟล์งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในไฟล์งาน tet2.js เราจะเขียนโค้ดได้ดังนี้ ซึ่งจะทำหน้าที่เป็นเซิร์ฟเวอร์

```

var express = require('express');
var fs = require('fs');
var app = express();
var http = require('http').Server(app);
var io = require('socket.io')(http);
var port = 1337;
var log = require('./log.js');
var db = require('./db.js');

app.use('/static', express.static(__dirname + '/public'));
app.get("/login", function(req,res){
    res.sendFile(__dirname + '/public/login.html');
});
app.get("/register", function(req,res){
    res.sendFile(__dirname + '/public/register.html');
});
app.get("/Forget", function(req,res){
    res.sendFile(__dirname + '/public/updatedata.html');
});
app.get("/aboutus", function(req,res){
    res.sendFile(__dirname + '/public/aboutus.html');
});
app.get("/contact", function(req,res){
    res.sendFile(__dirname + '/public/contact.html');
});
app.get("/beforehome2", function(req,res){
    res.sendFile(__dirname + '/public/beforehome2.html');
});
app.get("/home", function(req, res) {
    res.sendFile(__dirname + '/public/home.html');
});

app.get('/:id/:pwd', function (req, res) {
    var userInfo = getInfoByUsername(req.params.id);
    if(userInfo.isSuccess==true){
        var userObj = userInfo.info.split(',');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if(userObj[1]==req.params.pwd){
            console.log('ID:', req.params.id);
            console.log('Password:', req.params.pwd);
            res.send('./beforehome2');
        }
    } else {
        res.send('./error');
    }
});
function recordToDatabase(content){
    if(getInfoByUsername(content.username).isSuccess==false){
        db.append('./', 'database.txt', content, true);
    }
}
function readDatabase(){
    return fs.readFileSync('./database.txt','utf8');
}
function getInfoByUsername(name){
    var objInfo = { info:"", isSuccess:false };
    var contents = readDatabase();
    if(contents==null || contents==""){
    } else {
        var rows = contents.split('\n');
        for(var r=0;r<rows.length-1;r++){
            if(rows[r]!=null || rows[r]!=""){
                var userInfo = rows[r].split('|')[1];
                var uname = userInfo.split(',')[0];
                if(uname==name){
                    objInfo.info = userInfo;
                    objInfo.isSuccess = true;
                    break;
                }
            }
        }
    }
}
return objInfo;
}
io.on('connection', function(socket){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

socket.on('A', function(data) {
    console.log('Got message : ' + data);
    io.emit('x', data);
});
socket.on('connection',function(data){
    console.log('Data from NodeMCU :');
    console.log(data);
    io.emit('update_chart', data.message);
});
socket.on('save',function(data){
    log.append('./', (data.room+'.txt'),
(data.name+", "+data.lastname+", "+data.weight+", "+data.height+", "+data.comment),
false);
    io.emit('logdata',(data.name+", "+data.lastname+", "+data.weight+", "+data.height
+", "+data.comment));
});
socket.on('info',function(data){
    if(data=="room"){
        var content = fs.readFileSync('./305.txt','utf8');
        io.emit('all_room',content);
    }
});
socket.on('register', function(data) {
    if(getInfoByUsername(data.username).isSuccess==true){
        socket.emit('userinfo', 'Duplicate info');
    } else {

recordToDatabase((data.username+", "+data.password+", "+data.email));
    }
});
});
http.listen(port, function(){ console.log('listening on *:' + port); });

```

ในส่วนของ code server จะประกอบไปด้วยส่วน login โดยเก็บข้อมูลลงไปไว้ใน text file และยังมีส่วนที่จะแสดงผลไฟล์ HTML ตามที่เราร้องขอก็จะสามารถลิงค์หากันได้ตามที่เราต้องการ โดยจะสังเกตได้จากส่วนของ app.get

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

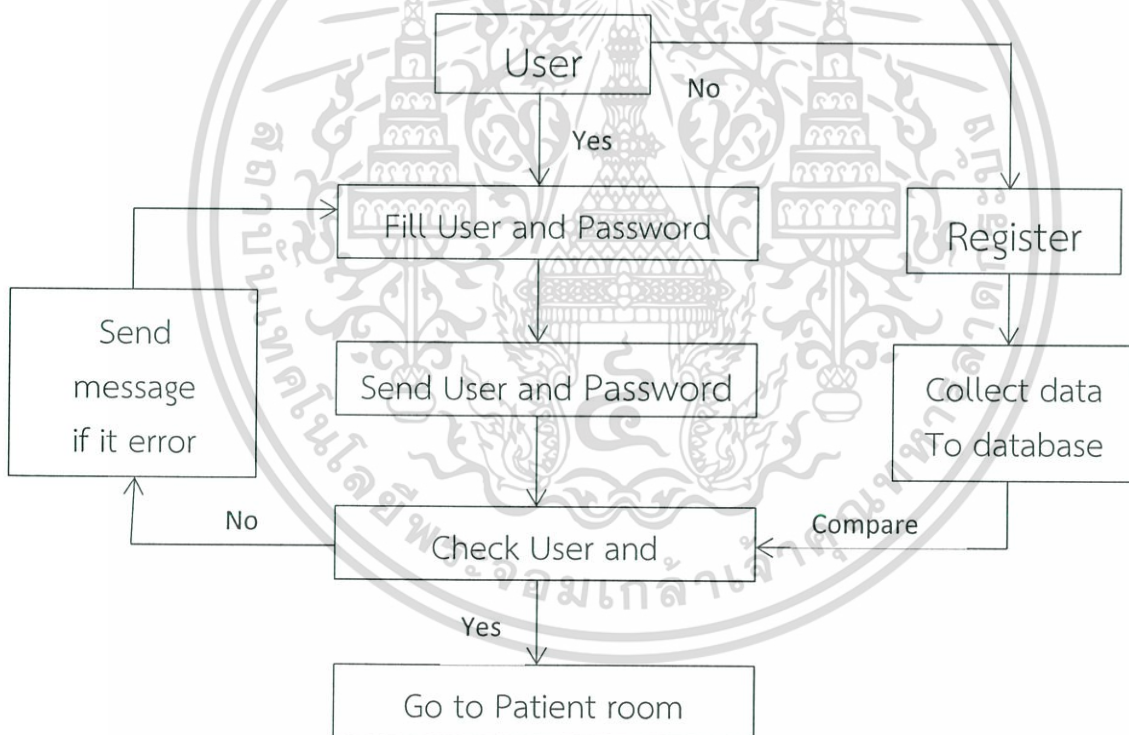
3.1.1.2. การออกแบบหน้าเว็บเพจ

การทำหน้าเว็บจะใช้ JavaScript, HTML, CSS เพื่อออกแบบหน้าเว็บเพจและตกแต่ง โดยจะแบ่งออกเป็น 4 หน้าเว็บเพจหลักๆคือส่วนของ

1. หน้าเพจล็อกอินเข้าสู่ระบบ
2. หน้าเพจหลักหรือเลือกห้องผู้ป่วย
3. หน้าเพจแสดงผลกราฟ
4. หน้าเพจเพิ่มเติม

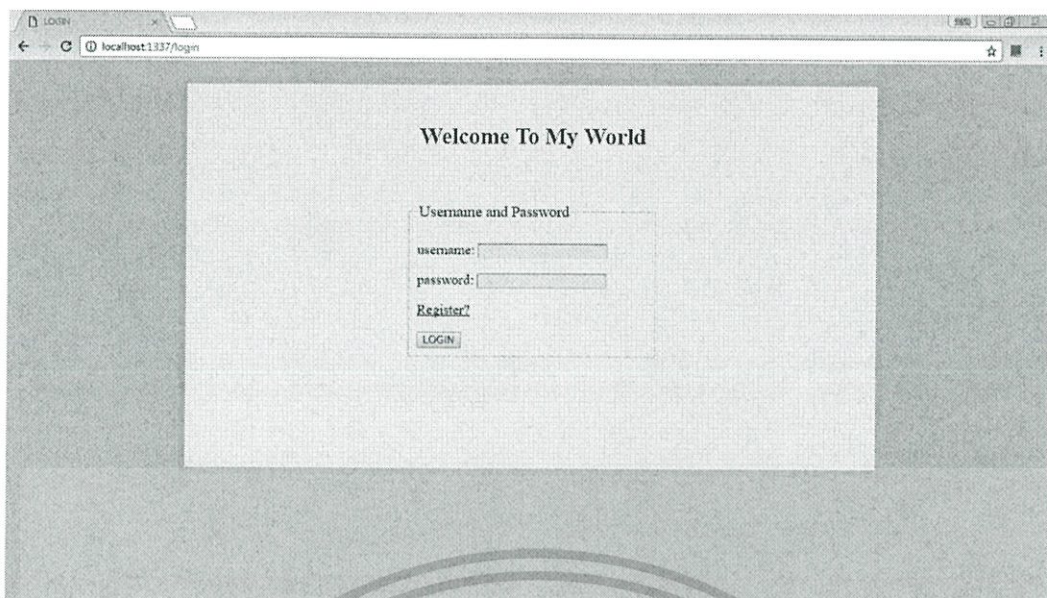
3.1.1.2.1. หน้าเพจล็อกอินเข้าสู่ระบบ

เริ่มจากถ้าผู้ใช้งานเพิ่งเคยเข้ามาใช้หน้าเว็บเป็นครั้งแรก ผู้ใช้งานจำเป็นต้องสมัครสมาชิกโดยกดเข้าไปที่คำว่า Register? หน้าเว็บก็จะแสดงส่วนที่ต้องกรอกเพื่อสมัครสมาชิก จากนั้น กด Back ก็จะกลับมาที่หน้า Login ข้อมูลจะไปปรากฏที่ 'database.txt' ส่วน server จะทำการตรวจสอบข้อมูลถ้าข้อมูลที่เข้ามาตรงกับที่ Register ไว้ก็จะใช้งานเว็บได้

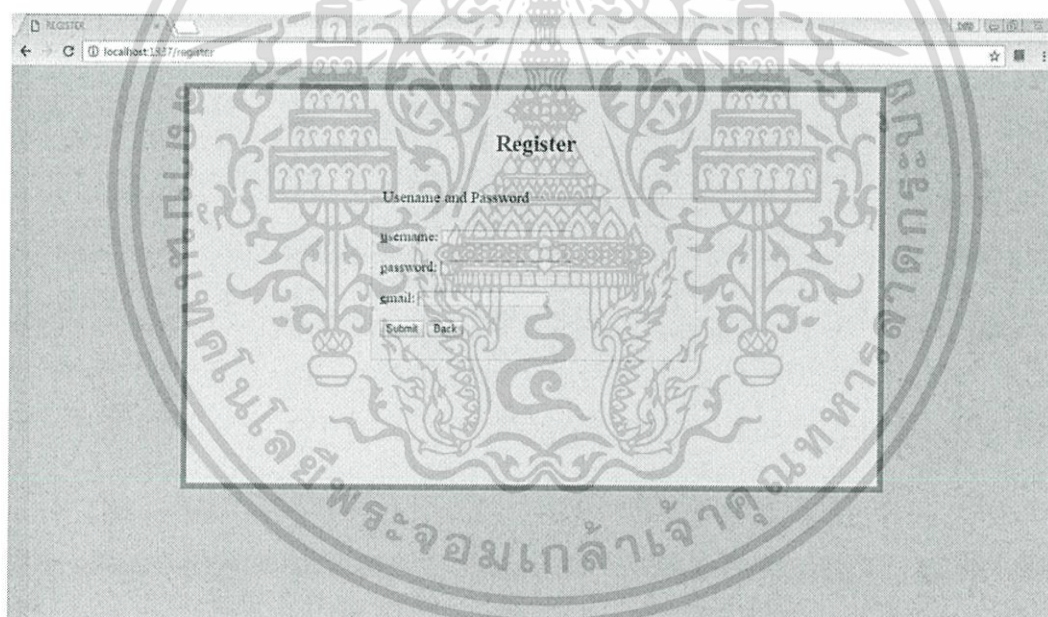


รูปที่ 3.4 แสดงผู้ใช้งานล็อกอินเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

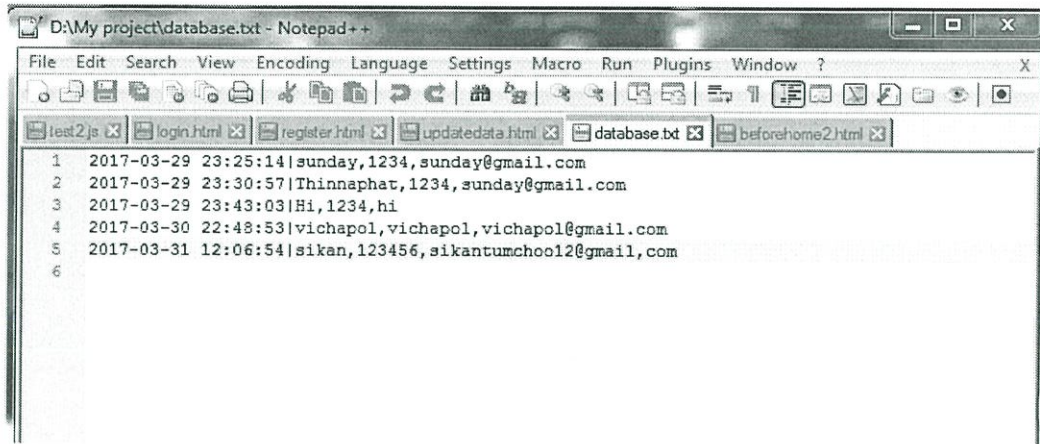


รูปที่ 3.5 หน้าเพจล็อกอิน



รูปที่ 3.6 หน้าเพจ Register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

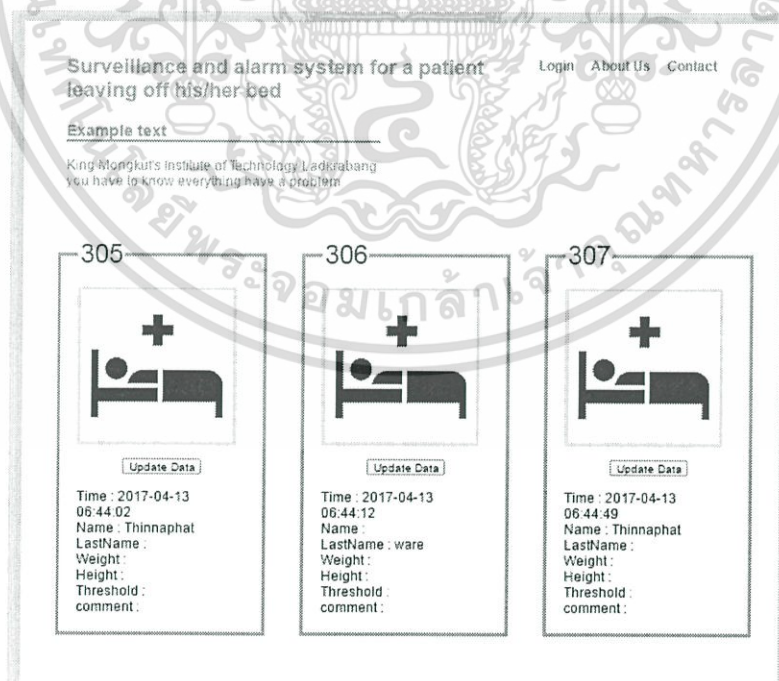
D:\My project\database.txt - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
test2.js login.html register.html updatedata.html database.txt beforehome2.html
1 2017-03-29 23:25:14|sunday,1234,sunday@gmail.com
2 2017-03-29 23:30:57|Thinnaphat,1234,sunday@gmail.com
3 2017-03-29 23:43:03|Hi,1234,hi
4 2017-03-30 22:48:53|vichapol,vichapol,vichapol@gmail.com
5 2017-03-31 12:06:54|sikan,123456,sikantumchoo1@gmail.com
6

```

รูปที่ 3.7 ส่วนการเก็บข้อมูลหรือ Database

3.1.1.2.2. หน้าเพจหลักหรือเลือกห้องผู้ป่วย

หลังจากที่ผู้ใช้งานเข้าสู่ระบบเรียบร้อยแล้วจะเข้าสู่หน้าเว็บหลักหรือหน้าเลือกห้องผู้ป่วยโดยหน้านี้จะแสดงห้องของผู้ป่วย สถานะและข้อมูลในเบื้องต้นของผู้ป่วย กรณีที่เกิดเหตุต่อผู้ป่วยเช่น การตกเตียงหน้าเว็บนี้จะแจ้งเตือนพร้อมกับส่งเสียงเพื่อให้พยาบาลหรือหมอเข้าช่วยเหลือได้ทันทีโดยในแต่ละห้องผู้ป่วยสามารถ Update ข้อมูลของผู้ป่วยได้ตลอดเวลาข้อมูลทั้งหมดจะแสดงที่แต่ละห้องของผู้ป่วยเมื่อทำการกดปุ่ม Save และค่า Threshold จะเปลี่ยนไปตามค่าน้ำหนักของผู้ป่วยที่กรอกไว้เพราะผู้ป่วยแต่ละคนน้ำหนักไม่เท่ากันค่า Threshold จึงต่างกัน



รูปที่ 3.8 หน้าเพจหลักหรือหน้าเลือกห้องผู้ป่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Update Patient Data

localhost:1337/Forget

Patient Data

Patient Data

Name :

Last name :

Weight :

Height :

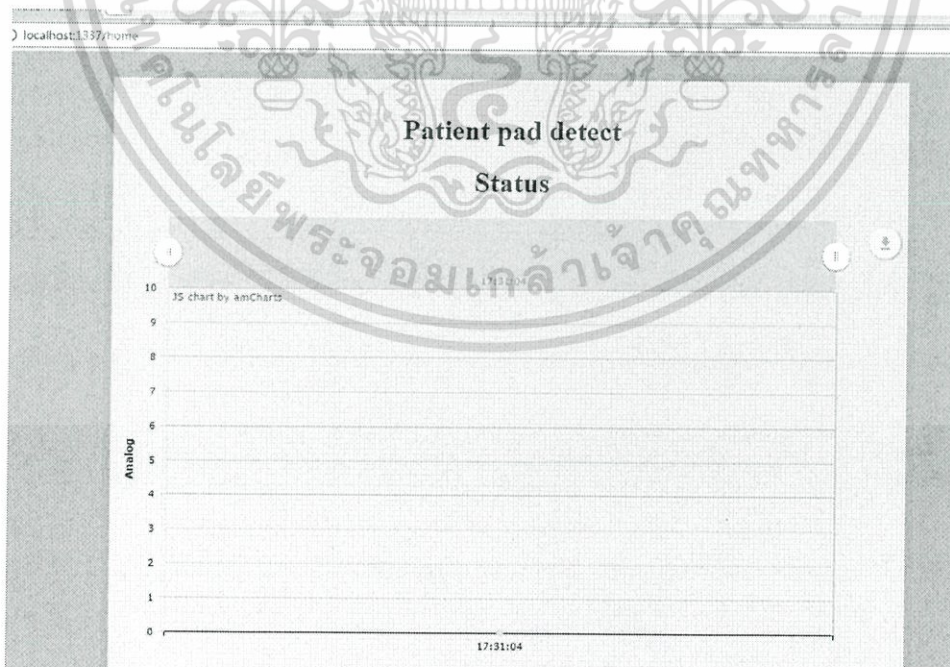
Comments

SAVE Cancel

รูปที่ 3.9 หน้า Update ข้อมูลของผู้ป่วย

3.1.1.2.3. หน้าเพจแสดงผลกราฟ

เมื่อทำการเลือกห้องของผู้ป่วยแต่ละคนได้แล้วเราสามารถเข้ามาดูกราฟของแต่ละห้องได้แบบตามเวลาจริงหรือ Real time ได้ตลอดเวลาการส่งข้อมูลจะเข้ามาอย่างต่อเนื่องในแต่ละห้อง

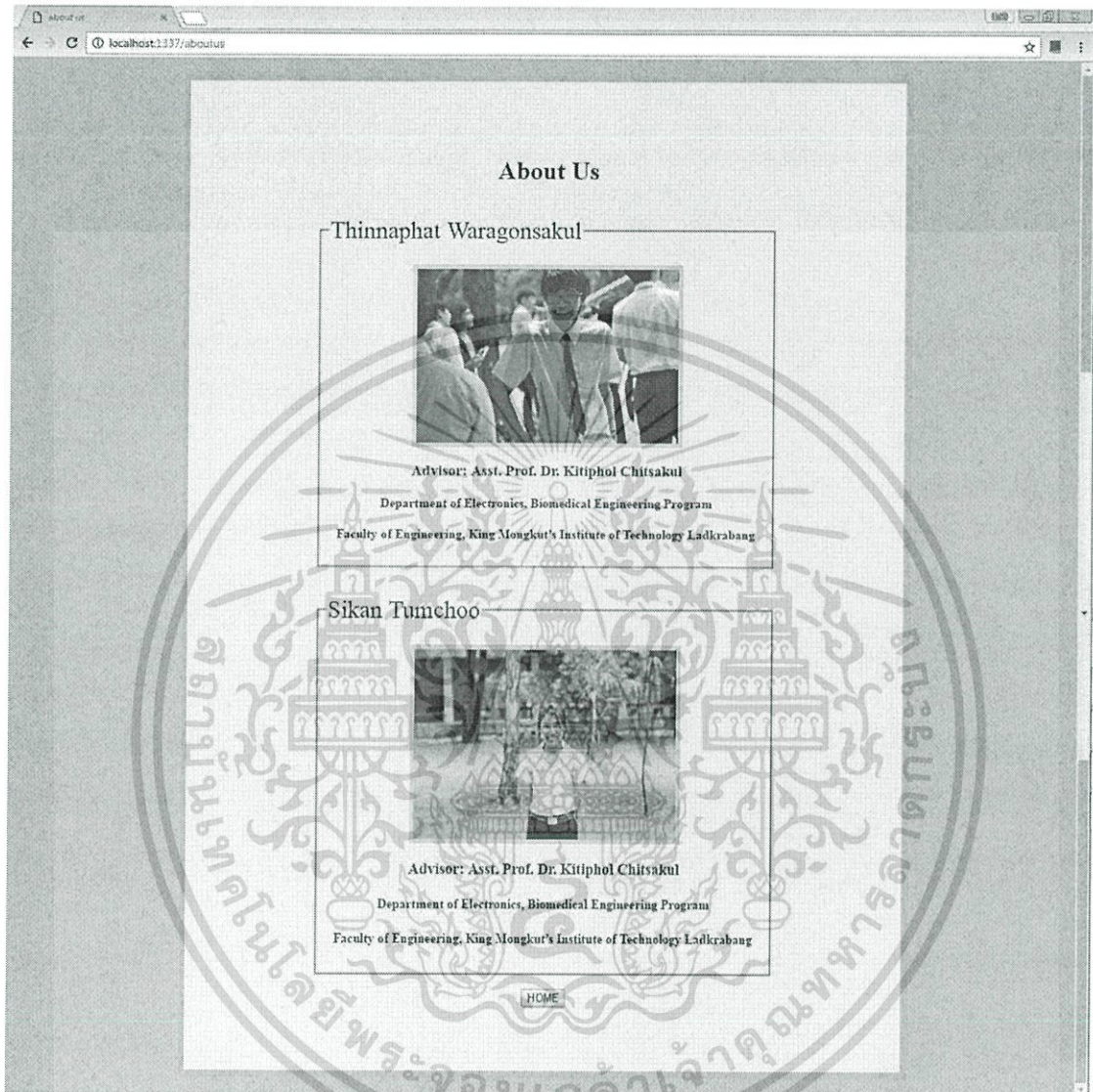


รูปที่ 3.10 หน้าเพจแสดงผลกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

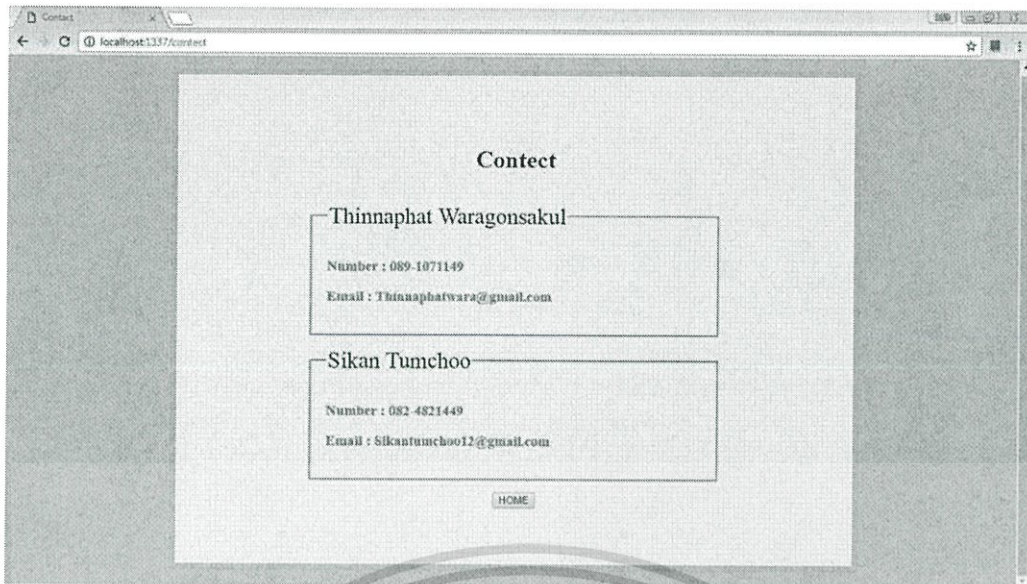
3.1.1.2.4. หน้าเพจเพิ่มเติม

จะเป็นหน้าเว็บในส่วนต่างๆที่เป็นองค์ประกอบย่อยๆของหน้าเว็บโดยจะประกอบด้วย หน้าเพจเกี่ยวกับผู้จัดทำและหน้าเพจเพื่อสำหรับติดต่อ



รูปที่ 3.11 หน้าเพจเกี่ยวกับผู้จัดทำ

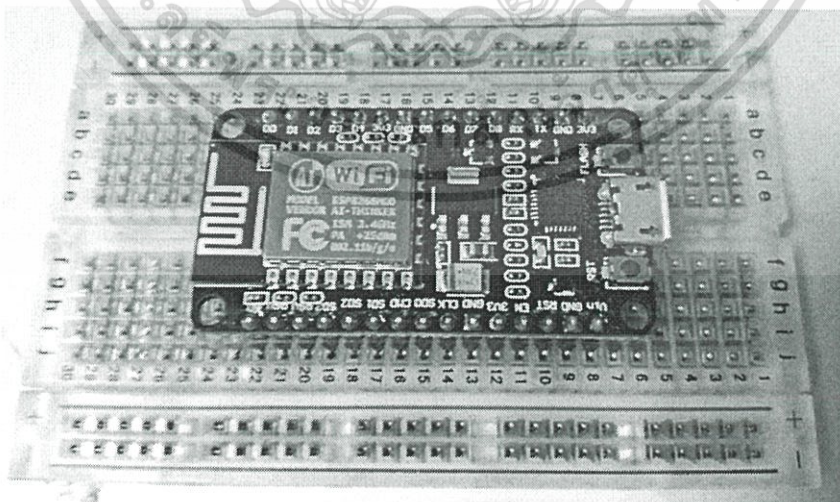
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 หน้าเพจสำหรับติดต่อ

3.2 การรับส่งสัญญาณผ่าน NodeMCU

เมื่อเซนเซอร์ได้รับสัญญาณสถานะจากผู้ป่วยแล้ว สัญญาณจะถูกส่งไปยัง NodeMCU เพื่อแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลและเนื่องจาก NodeMCU ESP8266 มีโมดูล Wi-Fi ในตัวจึงทำให้สามารถเชื่อมต่อกับระบบ Wi-Fi ได้อย่างง่ายดายผ่าน AT Command สามารถรับค่าจากเซนเซอร์ประมวลผล และรับส่งข้อมูลได้ด้วยตัวมันเองเพียงลำพัง ในที่นี้ NodeMCU จะทำการเชื่อมเข้ากับตัวเซิร์ฟเวอร์ ซึ่งตัวเซิร์ฟเวอร์จะรับค่าที่วัดได้จาก pressure sensor จากนั้นเซิร์ฟเวอร์จะนำค่าที่ได้มาเก็บข้อมูลทั้งหมดลงdatabaseและนำข้อมูลที่เก็บมาจากdatabaseแสดงผลบนweb browser

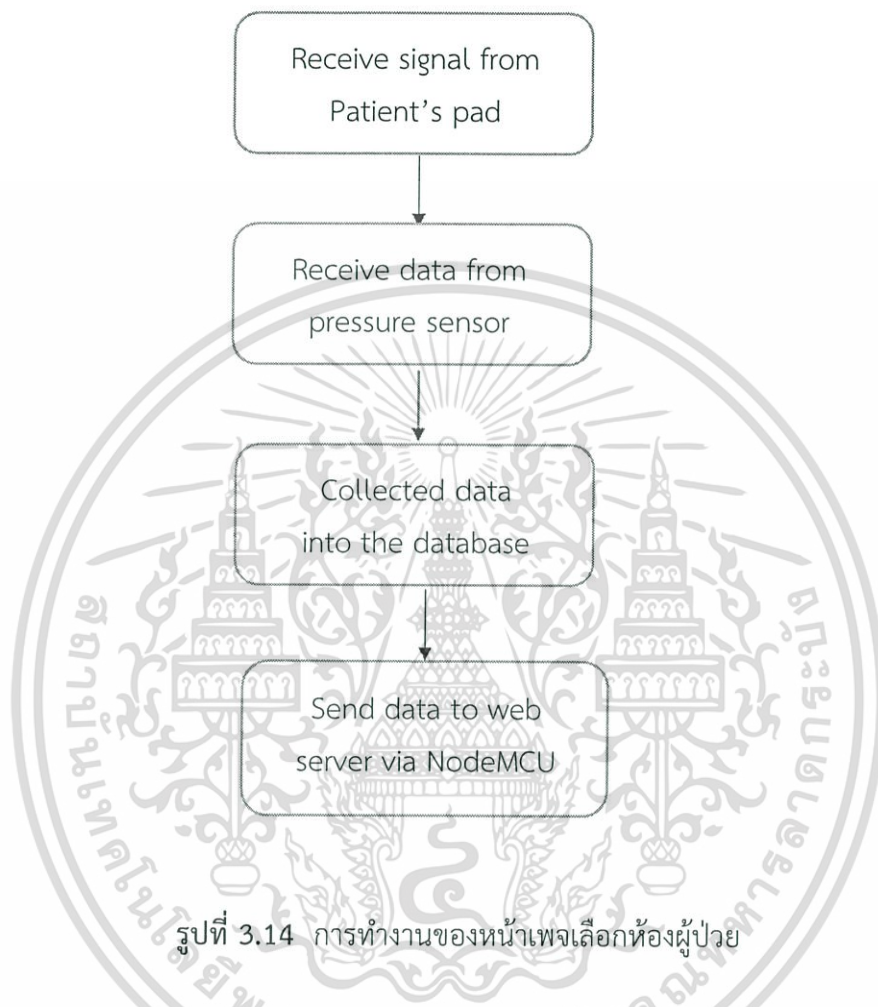


รูปที่ 3.13 NodeMCU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การทำงานส่วนของฮาร์ดแวร์ (Hardware)

ส่วนฮาร์ดแวร์ทำหน้าที่ในการรับความดันจากเซนเซอร์วัดแรงกด จากนั้นนำมาคำนวณหาปริมาณแรงดันต่อปริมาตรแล้วส่งค่าผ่าน NodeMCU แล้วไปเก็บไว้ในฐานข้อมูล ก่อนจะส่งไปยังเซิร์ฟเวอร์



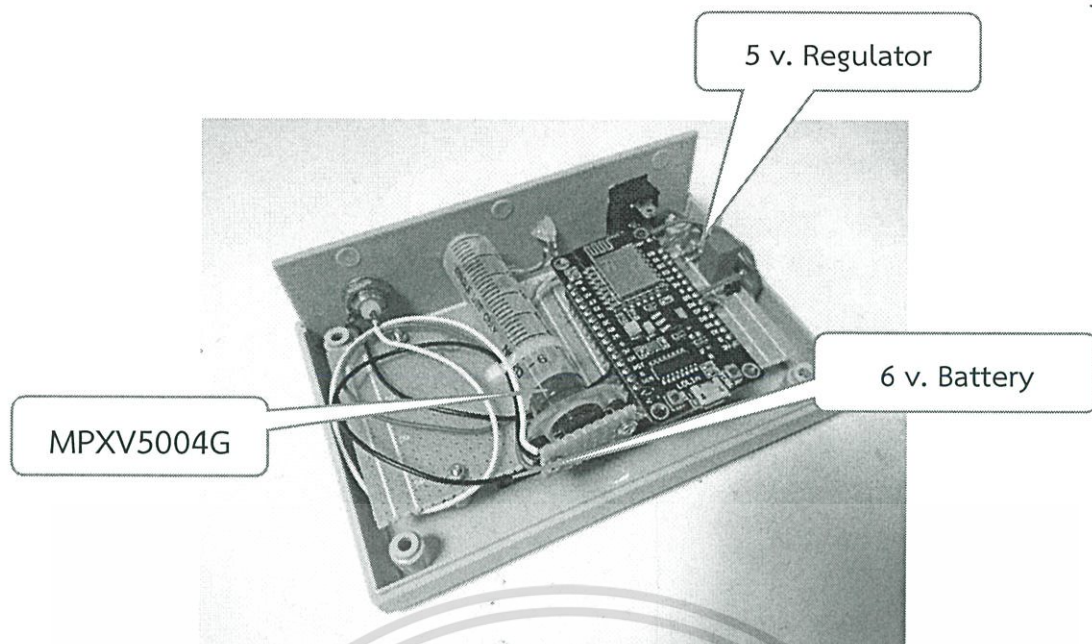
รูปที่ 3.14 การทำงานของหน้าเพจเลือกห้องผู้ป่วย

ฮาร์ดแวร์ประกอบด้วย 2 ส่วนหลักๆ คือ 1.กล่องเซนเซอร์ และ 2.หมอนลม โดยที่กล่องเซนเซอร์จะติดอยู่ที่เตียงของผู้ป่วยและต่อสายเชื่อมมาถึงหมอนลม ซึ่งจะวางอยู่บนเตียงของผู้ป่วย โดยวางอยู่บริเวณแผ่นหลังของผู้ป่วย

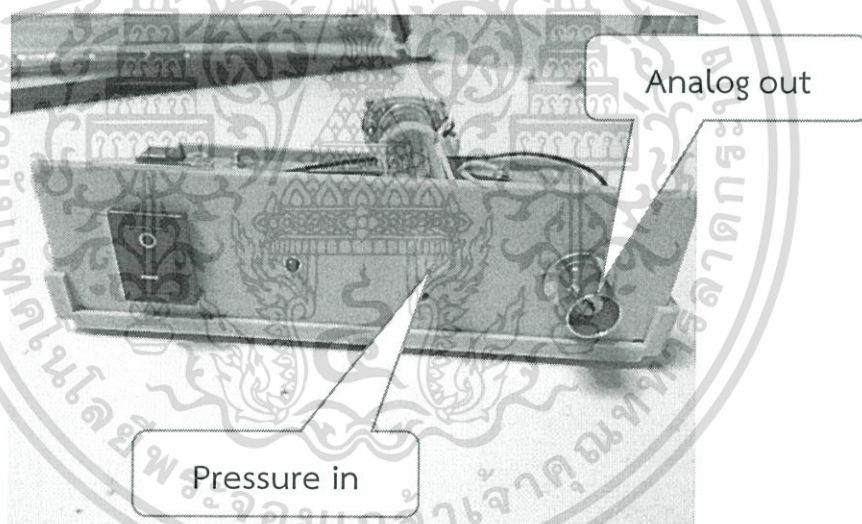
3.3.1 กล่องเซนเซอร์ (Sensor Box)

ภายในกล่องจะประกอบด้วย เซนเซอร์วัดแรงกด (pressure sensor) ซึ่งจะอยู่ภายในหลอดฉีดยา NodeMCU แบตเตอรี่ อนุาล็อกเอาต์พุท อุปกรณ์ลดแรงดัน (Regulator) หลอด LED เพื่อบอกสถานะการใช้งาน, สวิตช์เปิดปิด โดยเซนเซอร์วัดความดันต่ออยู่กับแหล่งจ่ายไฟ 6 โวลต์ ต่อกับอุปกรณ์ลดแรงดัน 5 โวลต์และต่อกับขานาฬิกาเอาต์พุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 ส่วนประกอบภายในกล่องเซนเซอร์



รูปที่ 3.16 ส่วนประกอบภายนอกกล่องเซนเซอร์

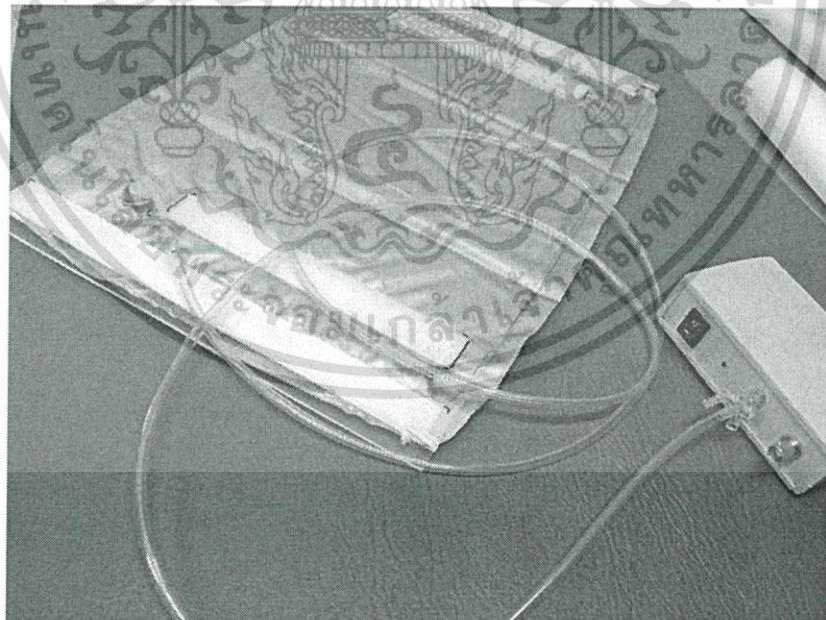
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 หมอนลม (Pneumatic Pad)

หมอนลมจะเป็นตัวเชื่อมต่อระหว่างตัวผู้ป่วยและเซนเซอร์ โดยเชื่อมสายจากหมอนลมเข้ากับกล่องของเซนเซอร์ที่เตียงของผู้ป่วย



รูปที่ 3.17 หมอนลม



รูปที่ 3.18 การเชื่อมสายระหว่างหมอนลมและกล่องเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองและผลการทดลอง (EXPERIMENT AND RESULTS OF DATA)

4.1 การทดลองในส่วนการรับส่งสัญญาณผ่าน NodeMCU

เมื่อทำการเชื่อมต่อ NodeMCU กับอินเทอร์เน็ตได้แล้ว ในกรณีที่ผู้ใช้ทำการเชื่อมต่อเป็นครั้งแรกหรือมีการเปลี่ยนการใช้งาน Wi-Fi เป็นชื่ออื่นๆ จะต้องทำการเลือกชื่อ Wi-Fi ให้ตรงตามที่ต้องการ ดังรูป

```

COM4
[SETUP] BOOT WAIT 1...

---- File system ----
FS File: /config.txt, size: 57B

WiFi status : 0
Scanning available networks...

---- Scan Networks ----
number of available networks:14
ssid:@KMITL Signal: -78 dBm Encryption: ssid:Connectify-free Signal: -50 dBm Encryption: WPA2
ssid:@KMITL Signal: -82 dBm Encryption: ssid:Connectify-3d Signal: -35 dBm Encryption: WPA2
ssid:@KMITL Signal: -74 dBm Encryption: ssid:.@ TRUEWIFI Signal: -73 dBm Encryption: None
ssid:true_home2G_kSV Signal: -85 dBm Encryption: Auto
ssid:.@ TRUEWIFI Signal: -85 dBm Encryption: None
ssid:.@ TRUEWIFI Signal: -74 dBm Encryption: None
ssid:DSS-5 Signal: -61 dBm Encryption: WPA2
ssid:@KMITL Signal: -76 dBm Encryption: ssid:KMITL-WIFI Signal: -76 dBm Encryption: None
ssid:.@ TRUEWIFI Signal: -79 dBm Encryption: None
ssid:PicassoLab Signal: -76 dBm Encryption: WPA

---- Local config ----
SSID :
Password :
Connecting...
    
```

รูปที่ 4.1 แสดงการเชื่อมต่อNodeMCU

เมื่อเลือกชื่อ Wi-Fi ได้แล้ว จะทำการเชื่อมต่อข้อมูลจาก NodeMCU ส่งไปยัง server ถ้าไม่มีการเปลี่ยนชื่อ Wi-Fi ใดๆทุกครั้งที่จ่ายไฟให้กับ NodeMCU ข้อมูลจะถูกเชื่อมต่อโดยอัตโนมัติและทำการส่งข้อมูลให้ในทันที ดังรูป

```

Scanning available networks...

----- Scan Networks -----
number of available networks:14
ssid:Connectify-sd      Signal: -27 dBm Encryption: WPA2
ssid:8 TRUEWIFI        Signal: -68 dBm Encryption: None
ssid:Connectify-free   Signal: -54 dBm Encryption: WPA2
ssid:@KMITL            Signal: -77 dBm Encryption: ssid:@KMITL Signal: -72 dBm Encryption: ssid:KMITL-WIFI Signal: -85 dBm Encryption: None
ssid:Sunlight          Signal: -75 dBm Encryption: Auto
ssid:Rugby             Signal: -86 dBm Encryption: Auto
ssid:DSS-5             Signal: -57 dBm Encryption: WPA2
ssid:@KMITL            Signal: -71 dBm Encryption: ssid:PicassoLab Signal: -69 dBm Encryption: WPA
ssid:Sono2ASitch       Signal: -82 dBm Encryption: Auto
ssid:HUAWEI_F9         Signal: -37 dBm Encryption: WPA2
ssid:BURN_LAB         Signal: -88 dBm Encryption: WPA2

----- Local config -----
SSID : HUAWEI_F9
Password : 12112537
Connecting...

----- Network Info -----
Connected using IP : 192.168.43.164
And MAC address : 5C:C7:F7:88:DA:E1
HTTP/1.1 200 OK
Content-Type: text/plain; charset=UTF-8
Content-Length: 100
Access-Control-Allow-Credentials: true
Access-Control-Allow-Origin: Arduino
Set-Cookie: io=VigNtKo9ReqlzvWAA86; Path=/; HttpOnly
Date: Thu, 06 Apr 2017 05:59:38 GMT
Connection: keep-alive

97:0["sid":"VigNtKo9ReqlzvWAA86","upgrades":["websocket"],"pingInterval":25000,"pingTimeout":60000]
Connected. SID=VigNtKo9ReqlzvWAA86
Connecting via WebSocket
HTTP/1.1 101 Switching Protocols
Upgrade: websocket

```

รูปที่ 4.2 การส่งข้อมูลของ NodeMCU

ข้อมูลที่ได้อาจจะถูกส่งออกมาในรูปแบบของตัวเลข ซึ่งจะใช้ตัวเลขเป็นมาตรฐานในการบ่งบอกถึงความดันภายในหมอนลม

```

COM3
|
|
[]42["update_chart","412"][]42["update_chart","412"]
Message size = 24
Received message = 42["update_chart","412"]
42["update_chart","412"]
[]42["update_chart","445"][]42["update_chart","445"]
Message size = 24
Received message = 42["update_chart","445"]
42["update_chart","445"]
[]42["update_chart","463"][]42["update_chart","463"]
Message size = 24
Received message = 42["update_chart","463"]
42["update_chart","463"]
[]42["update_chart","477"][]42["update_chart","477"]
Message size = 24
Received message = 42["update_chart","477"]
42["update_chart","477"]
[]42["update_chart","479"][]42["update_chart","479"]
Message size = 24
Received message = 42["update_chart","479"]
42["update_chart","479"]
[]42["update_chart","479"][]42["update_chart","479"]
Message size = 24
Received message = 42["update_chart","479"]
42["update_chart","479"]
[]42["update_chart","479"][]42["update_chart","479"]
Message size = 24
Received message = 42["update_chart","479"]
42["update_chart","479"]
[]42["update_chart","479"][]42["update_chart","479"]
Message size = 24
Received message = 42["update_chart","479"]
42["update_chart","479"]
[]42["update_chart","479"][]42["update_chart","479"]
Message size = 24
Received message = 42["update_chart","479"]
42["update_chart","479"]
[]42["update_chart","479"][]42["update_chart","479"]
Message size = 24
Received message = 42["update_chart","479"]
42["update_chart","479"]
[]42["update_chart","478"][]42["update_chart","478"]
Message size = 24
Received message = 42["

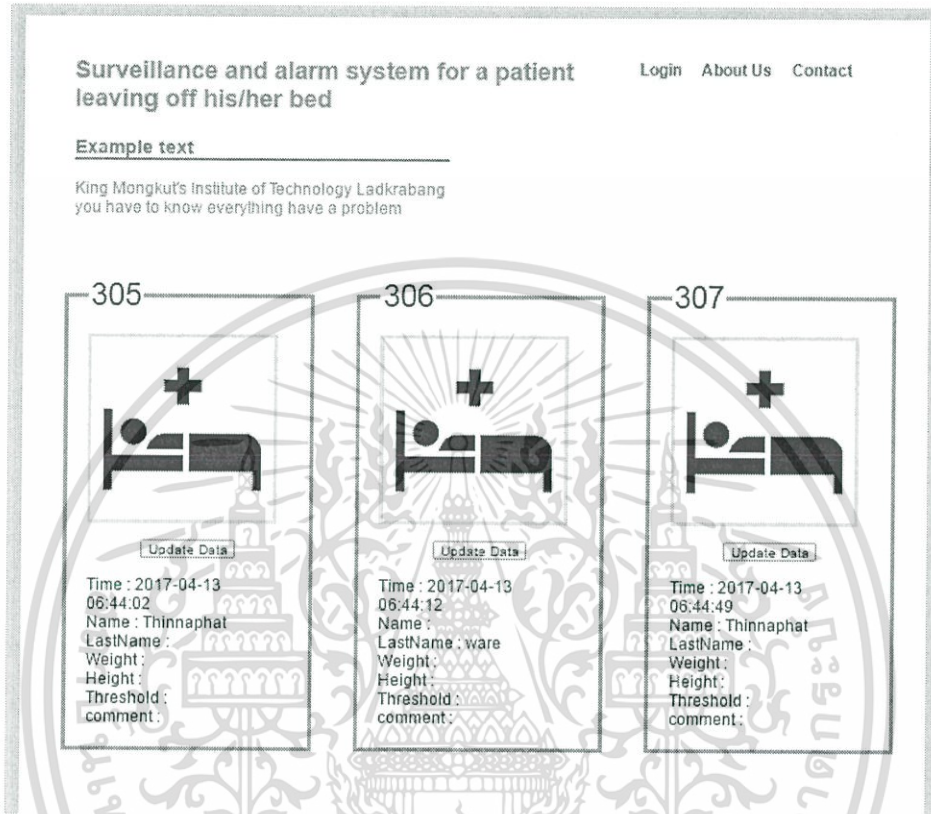
```

รูปที่ 4.3 ข้อมูลของ NodeMCU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองในส่วนของเซิร์ฟเวอร์ (Server)

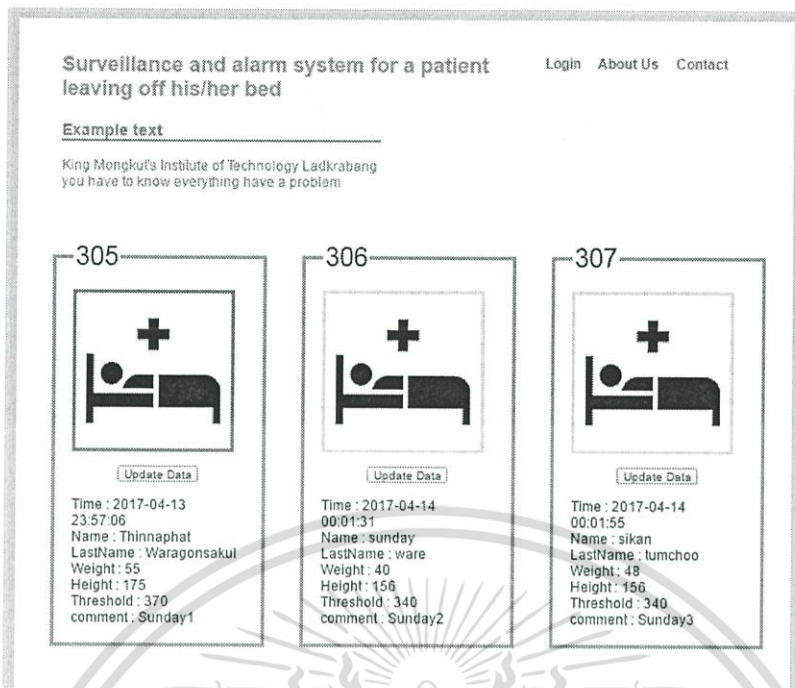
เมื่อผู้ใช้งานทำการ Register และ Login เสร็จเรียบร้อยแล้วจะปรากฏข้อมูลขึ้นใน Database.txt ก็จะสามารถเข้าใช้งานหน้าเว็บได้ Login แล้วก็จะพบกับหน้าเลือกห้องของผู้ป่วยจะมีข้อมูลของผู้ป่วยและเลขห้องแสดงอยู่



รูปที่ 4.4 หน้าเลือกห้องข้อมูลของผู้ป่วย

กรณีที่ผู้ป่วยไม่อยู่บนเตียงหน้าเว็บจะทำการส่งเสียงเตือนและจะเกิดสัญญาณไฟกระพริบที่ห้องผู้ป่วยที่มีค่าต่ำกว่าค่า Threshold นั้นเป็นสัญญาณว่าเกิดปัญหากับผู้ป่วยห้องนั้น หมอและพยาบาลควรรีบเข้าไปให้การช่วยเหลือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 ตัวอย่างหน้าเว็บเมื่อผู้ป่วยไม่อยู่บนเตียงจะมีสัญญาณเตือนและเสียงดังออกมา

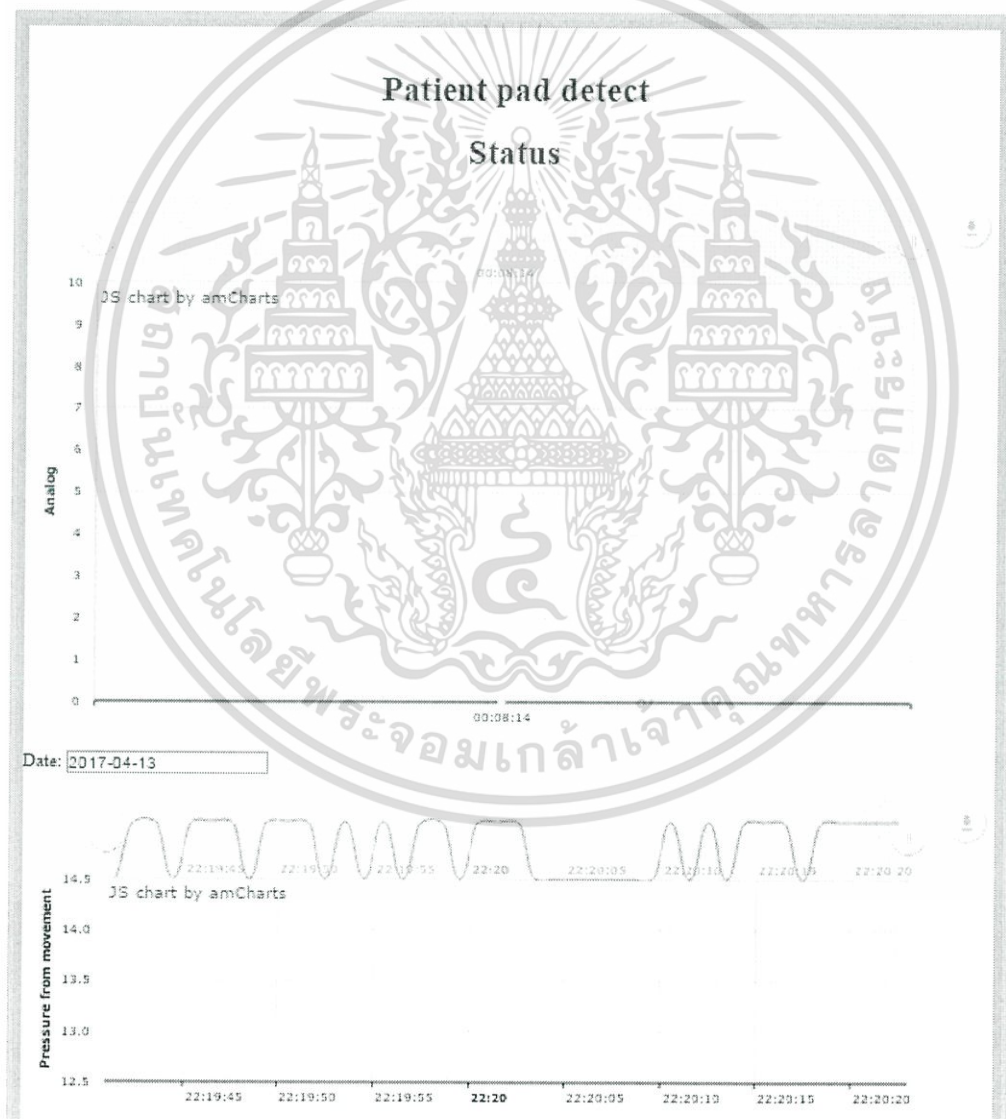


รูปที่ 4.6 ตัวอย่างหน้าเลือกห้องผู้ป่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของแต่ละห้องของผู้ป่วยจะประกอบไปด้วยปุ่ม Update Data จะเป็นปุ่มที่ไว้ใช้สำหรับแก้ไขหรือเพิ่มเติมข้อมูลของผู้ป่วยเมื่อคลิกเข้าไปจะแสดงหน้าอัพเดทข้อมูลผู้ป่วยให้ทำการกรอก จากนั้นกดปุ่ม Save และปุ่ม Back ตามลำดับ ข้อมูลทั้งหมดจะปรากฏที่หน้าห้องของผู้ป่วย โดยจากรูปภาพจะเป็นการจำลองการใช้งานของห้อง 305 จะแสดงเวลาที่เรากำหนดข้อมูล ชื่อ นามสกุล น้ำหนัก ส่วนสูง และค่า Threshold ซึ่งค่านี้อาจเปลี่ยนแปลงตามน้ำหนักของผู้ป่วยที่กรอกตอนอัพเดทข้อมูล

เราจะสามารถดูข้อมูลกราฟแบบเรียลไทม์ ได้โดยดูได้จากหน้า home หรือจากการคลิกเข้าไปที่รูปภาพในแต่ละห้องของผู้ป่วยจะแสดงกราฟที่รับค่ามาจาก NodeMCU โดยจะสังเกตเห็นถึงลักษณะของกราฟที่เปลี่ยนแปลงไปในการนำข้อมูลมาศึกษา นอกจากนี้ยังสามารถดูข้อมูลย้อนหลังได้จากกราฟข้างล่าง โดยจะแสดงค่าที่เก็บไว้ในครั้งก่อนๆ



รูปที่ 4.7 หน้ากราฟตอนยังไม่ได้รับข้อมูลข้อมูลแต่จะแสดงข้อมูลย้อนหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Node.js command prompt - node test2
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
< message: '311' >
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
< message: '311' >
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
< message: '311' >
Data from NodeMCU :
< message: '310' >
Data from NodeMCU :
```

รูปที่ 4.8 Server รับข้อมูล

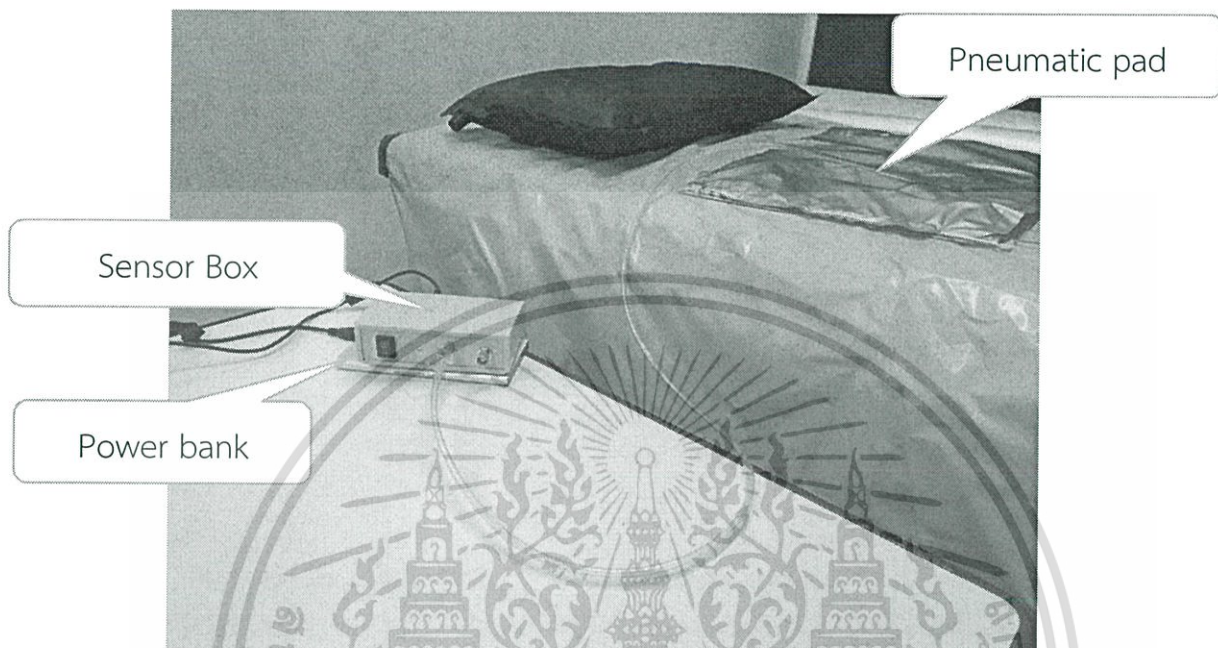


รูปที่ 4.9 กราฟแสดงผลเมื่อได้รับข้อมูลแบบเรียลไทม์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองในส่วนของฮาร์ดแวร์ (Hardware)

· โดยวางหมอนลมไว้บริเวณแผ่นหลังของผู้ป่วย จากนั้นจ่ายไฟให้กับ NodeMCU และ pressure sensor เชื่อมต่อWi-Fi ให้กับ NodeMCU เพื่อทำการส่งข้อมูลการนอนของผู้ป่วยไปยัง server



รูปที่ 4.10 การวางอุปกรณ์



รูปที่ 4.11 การทดลองวัดค่าจากเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทดลองหรือการนำไปใช้จริงทุกครั้งจะต้องทำการปรับแก้ค่าน้ำหนักของผู้ป่วย เพราะผู้ป่วยแต่ละคนมีน้ำหนักหนักแตกต่างกันไป จึงต้องตั้งค่า Threshold เพื่อป้องกันการผิดพลาดทางการวัดค่าข้อมูลผู้ป่วย ซึ่งสามารถสรุปได้เป็นตารางตัวอย่างค่าน้ำหนัก ดังนี้

ชื่อ	น้ำหนัก	ค่าเริ่มต้น	Average	นอนปกติ	Average	นอน ตะแคง ซ้าย	Average	นอน ตะแคง ขวา	Average
Aum	48.2	299 - 300	299.5	497 - 502	499.5	559 - 565	562	568 - 574	571
Sunday	56.3	299 - 302	300.5	746 - 760	753	617 - 634	625.5	636 - 643	639.5
Jack	72.4	300 - 302	301	858 - 900	879	722 - 779	750.5	572 - 603	587.5
Thum	67.6	302	302	838 - 849	843.5	719 - 735	727	723 - 769	746
Tong	71.6	302	302	1007 - 1020	1013.5	726 - 741	733.5	604 - 777	690.5
Nut	57.4	303 - 304	303.5	662 - 668	665	543 - 602	572.5	645 - 671	658
Ping	99.2	301 - 302	301.5	990 - 1024	962	745 - 764	754.5	913 - 925	919
James	53.3	299 - 300	299.5	647 - 658	652.5	454 - 465	459.5	497 - 513	505
Pae	65.9	296 - 298	297	545 - 550	547.5	423 - 430	426.5	412 - 450	431
Pair	70.4	298 - 299	298.5	846 - 900	873	676 - 686	681	645 - 683	664
Stop	71.6	299 - 300	299.5	948 - 962	955	725 - 745	735	550 - 642	596
Net	95.1	298 - 299	298.5	945 - 1010	977.5	722 - 779	750.5	772 - 803	787.5
Mel	57.2	299 - 300	299.5	654 - 662	658	551 - 567	559	452 - 463	457.5
Tanu	59.9	299 - 300	299.5	704 - 745	724.5	603 - 654	628.5	612 - 645	628.5
Phem	70	300 - 301	300.5	808 - 835	821.5	578 - 614	596	531 - 548	539.5
Fai	44.2	298 - 300	299	967 - 984	975.5	576 - 586	581	535 - 553	544
Mos	61.7	300 - 301	300.5	565 - 580	572.5	443 - 460	451.5	402 - 410	406
Mochu	50.7	300 - 301	300.5	547 - 558	552.5	414 - 435	424.5	457 - 543	500
Bell	53.1	299 - 300	299.5	607 - 628	617.5	454 - 465	459.5	487 - 521	504
Fame	56.1	299 - 300	299.5	537 - 578	557.5	484 - 525	504.5	435 - 467	451
Manao	52.7	298 - 299	298.5	634 - 652	643	474 - 512	493	501 - 530	515.5
Orn	49.3	300 - 301	300.5	497 - 502	499.5	559 - 565	562	568 - 574	571
Tam	47.5	299 - 300	299.5	467 - 482	474.5	549 - 565	557	558 - 574	566
Pond	66.1	300 - 301	300.5	535 - 561	548	413 - 452	432.5	461 - 472	516.5
Jab	58.3	299 - 300	299.5	684 - 735	709.5	561 - 589	575	582 - 621	601.5
Bright	74.3	300 - 301	300.5	882 - 920	901	752 - 781	766.5	561 - 651	606
Earth	62.3	299 - 300	299.5	485 - 520	502.5	417 - 457	437	418 - 481	449.5
Nick	42.9	300 - 301	300.5	528 - 546	537	509 - 516	512.5	548 - 574	561
Nut	45.5	299 - 300	299.5	507 - 512	509.5	514 - 536	525	516 - 544	530
Palm	61.4	298 - 300	299	562 - 570	566	445 - 465	455	412 - 441	426.5
Fin	75.2	300 - 301	300.5	848 - 864	856	618 - 645	631.5	561 - 572	566.5
Fifa	75.4	300 - 301	300.5	846 - 858	852	575 - 604	589.5	521 - 582	551.5
Bhum	49.7	299 - 300	299.5	467 - 482	474.5	526 - 531	528.5	549 - 561	555
Jeffy	55.1	298 - 300	299	726 - 741	733.5	637 - 646	641.5	546 - 587	566.5
Gift	52.3	300 - 301	300.5	624 - 682	653	454 - 462	458	481 - 519	500
Gift	59.4	299 - 300	299.5	614 - 658	636	571 - 602	586.5	562 - 603	582.5
Mew	46.2	300 - 301	300.5	467 - 471	469	554 - 561	557.5	461 - 503	482
Top	50.5	299 - 300	299.5	537 - 578	557.5	474 - 482	478	428 - 463	445.5
Walee	54.4	300 - 301	300.5	584 - 618	601	434 - 465	449.5	456 - 481	468.5
Pair	46.6	299 - 300	299.5	495 - 512	503.5	549 - 565	557	538 - 554	546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tua	50.4	300 - 301	300.5	527 - 538	532.5	454 - 472	463	447 - 463	455
James	89.9	299 - 300	299.5	975 - 1003	989	714 - 763	738.5	689 - 725	707
Pond	65.4	298 - 300	299	551 - 570	560.5	413 - 420	416.5	476 - 492	484
David	61.2	298 - 299	299	565 - 580	572.5	443 - 460	451.5	402 - 410	406
Flook	80.1	299 - 300	299.5	945 - 983	964	735 - 752	743.5	718 - 767	742.5
Nut	83.7	299 - 300	299.5	968 - 993	980.5	718 - 787	752.5	704 - 741	722.5
Ta	68.3	300 - 301	300.5	868 - 901	884.5	756 - 785	770.5	723 - 769	746
Kor	51.8	298 - 300	299	547 - 558	552.5	414 - 435	424.5	457 - 543	500
Gear	65.8	300 - 301	300.5	545 - 550	547.5	423 - 430	426.5	412 - 450	431
Toto	70.2	300 - 301	300.5	808 - 835	821.5	578 - 614	596	531 - 548	539.5

ตารางที่ 4.1 การทดลองเทียบค่าน้ำหนัก

Weight	Threshold	Average Normal	Average นอนปกติ	Average ตะแคงซ้าย	Average ตะแคงขวา
40 - 49	340	299.875	495.875	528.75	547.75
50 - 59	370	299.911	635.235	517.794	528.147
60 - 69	400	299.708	614.5	499.45	504.25
70 - 79	430	300.25	894.8	666.05	588.5
80 - 89	460	299.5	977.833	744.833	724
90 - 99	490	300	969.75	752.5	655.375

ตารางที่ 4.2 สรุปค่าน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ (CONCLUSION AND SUGGESTION)

5.1 สรุปผลการวิจัย

ระบบการทำงานของแผ่นแจ้งเตือนสามารถแบ่งการทำงานออกเป็น 3 ส่วนดังนี้

5.1.1 ส่วนของเซ็นเซอร์

ส่วนของเซ็นเซอร์ประกอบด้วย Pressure sensor มีขนาดและความสามารถในการรับค่าที่หลากหลายต้องคำนึงถึงแรงดันและปริมาตรที่สามารถรับได้ให้เหมาะสมเพื่อใช้ส่งสัญญาณที่ตีส่งส่วนส่งข้อมูล

5.1.2 ส่วนของตัวส่งข้อมูล

ส่วนของตัวส่งข้อมูลประกอบด้วย NodeMCU เป็นโมดูลซึ่งใช้ภาษาเดียวกับ Arduino และยังมีโมดูล Wi-Fi อยู่ในตัว สามารถรับค่าข้อมูลจาก Pressure sensor และส่งค่าข้อมูลนั้นสู่ NodeMCU ได้ตลอดเวลาโดยไม่เกิดปัญหาในการทำงาน

5.1.3 ส่วนของเว็บเซิร์ฟเวอร์

ส่วนของเว็บเซิร์ฟเวอร์ ประกอบด้วย เว็บโฮสต์หรือคอมพิวเตอร์ และฐานข้อมูล ส่วนของเว็บเซิร์ฟเวอร์สามารถตอบสนองกับผู้ใช้งานได้ ในส่วนการส่งและรับค่าข้อมูลถือว่าเป็นที่น่าพอใจสามารถแสดงผลเป็นกราฟและค่าเฉลี่ยเพื่อเก็บข้อมูลไว้ดูในภายหลังได้ แต่ยังคงมีปัญหาในส่วนฐานข้อมูล ซึ่งใช้การเก็บข้อมูลเป็น text file แต่ระบบฐานข้อมูลที่เสถียรควรจะเก็บอยู่ใน MySQL หรือ MongoDB

5.2 ข้อเสนอแนะ

5.2.1 สมาชิกในกลุ่มโปรเจกต์ขาดความรู้ทางด้านระบบการเชื่อมต่อ ระบบเครือข่ายและภาษาทางคอมพิวเตอร์ต่างๆทำให้การใช้งาน การพัฒนาใช้เวลาค่อนข้างนานในการค้นคว้าหาข้อมูลเพิ่มเติมและแก้ไขปัญหา

5.2.2 การเลือกซื้ออุปกรณ์สำหรับทำโครงการต้องมีความละเอียดถี่ถ้วน ทำให้การวางแผนที่จะซื้อต้องคิดให้ดีกว่าก่อน

5.2.3 ในส่วนของการเก็บข้อมูลลงในฐานข้อมูล ควรจะใช้เวลาในการศึกษาเพิ่มเติมในเรื่อง MySQL หรือ MongoDB เพื่อใช้เก็บข้อมูลและทำงานมีความน่าเชื่อถือมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางการพัฒนาโครงการ

5.3.1 ด้านความสวยงาม สามารถออกแบบเว็บเพจให้สวยงามขึ้นโดยเขียน Style sheet ใหม่และสามารถออกแบบให้ใช้งานได้สะดวกมากขึ้นตามลักษณะของผู้ใช้งาน

5.3.2 ด้านการทำงาน สามารถออกแบบให้ตัวโมดูลรับค่าจากหลายๆห้องผู้ป่วยได้โดยไม่ต้องเพิ่มโมดูลซึ่งจะเป็นการสิ้นเปลือง พัฒนาให้ระบบการเชื่อมต่ออินเทอร์เน็ตเสถียรมากขึ้น และรองรับการใช้งานเซิร์ฟเวอร์ได้หลากหลาย

5.3.3 ด้านฮาร์ดแวร์ สามารถพัฒนาตัวหมอนลม ให้สามารถมีขนาดตามที่ต้องการและสามารถป้องกันลมรั่วได้อย่างมีประสิทธิภาพ

5.3.4 ด้านการแสดงผล สามารถพัฒนาต่อยอดในส่วนของ android application เพื่อความสะดวกในการดูข้อมูลของผู้ป่วย

5.3.5 กรณีขาดการเชื่อมต่ออินเทอร์เน็ต ควรจะพัฒนาให้มีการรองรับการใช้งานในขณะที่ไม่มีการเชื่อมต่ออินเทอร์เน็ต เพื่อให้การทำงานครอบคลุมได้ทุกสถานการณ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] การประยุกต์ใช้ Internet of thing [Online].เข้าถึงได้จาก:
<http://www.bangkokgis.com/bangkokgis_2008/system_file/-t1439539799.pdf>. (6 กันยายน 2559)
- [2] การป้องกันผู้ป่วยหลบนี [Online]. เข้าถึงได้จาก:
<<http://www.suansaranrom.go.th/PhP/html/modules.php?name=Content&pa=showpage&pid=10>>. (6 กันยายน 2559)
- [3] การดูแลผู้ป่วยในเตียง [Online]. เข้าถึงได้
จาก: <<https://www.doctor.or.th/article/detail/5967>>. (6 กันยายน 2559)
- [4] กุลวดี โภคสวัสดิ์.สร้างสรรค์ บันทึกลง เว็บไซด์ให้สวยด้วย CSS.กรุงเทพฯ: ซีเอ็ดยูเคชั่น, 2554.
- [5] ณัฐนนท์ พรหมมา, อานนท์ กิระติวสิน และ ภัทรภร ยอดวันดี.ระบบเฝ้าระวังผู้ป่วย
ฉุกเฉินแบบไร้สาย. วิทยานิพนธ์ปริญญาบัณฑิต สาขาวิชาวิศวกรรมสารสนเทศ คณะ
วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,2556.
- [6] ทำความเข้าใจเรื่อง Internet of Things (IoT) [Online]. เข้าถึงได้
จาก: <<http://www.veedvil.com/news/internet-of-things-iot/>>. (6 กันยายน
2559)
- [7] ชีระศักดิ์ สุโชตินันท์.ทำไป ลองไป จาวาสคริปส์.กรุงเทพฯ: สำนักพิมพ์สวีสวีไอที, 2547.
บัณฑิต จามรภูติ.การประยุกต์ใช้ระบบ โคลเอนต์ เซิร์ฟเวอร์.กรุงเทพฯ:ว. เพ็ชรสกุล,
2542.
- [8] พิรพร หมุนสนธิ และ วันวิสาข์ เมฆฉาย.ใช้งาน JavaScript แบบมืออาชีพ.
กรุงเทพฯ:สำนักพิมพ์เคทีพี,2550
- [9] สุรเชษฐ์ วงศ์ชัยพรพงษ์.เทคนิคและการประยุกต์ใช้งาน สไตล์ชีท (CSS).กรุงเทพฯ:
สำนักพิมพ์เคทีพี, 2549.
- [10] รู้จักกับ Arduino ESP8266 (NodeMCU) [Online]. เข้าถึงได้จาก:
<<http://www.ayarafun.com/2015/08/introduction-arduino-esp8266-nodemcu/>>. (6 กันยายน 2559)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [11] รู้จักกับ Internet of Things [Online]. เข้าถึงได้จาก:
<<http://tormootech.blogspot.com/2015/11/intenet-of-things.html>>. (6 กันยายน 2559)
- [12] รู้จักกับ NODE MCU ESP8266 ตอนที่ 1 [Online]. เข้าถึงได้จาก:
<<http://wisdomgoody.blogspot.com/2015/06/node-mcu-esp8266.html>>. (6 กันยายน 2559)
- [13] BeYourCyber.เจาะลึกกับ node.js แบบเริ่มต้นทำความรู้จัก [Online].
เข้าถึงได้จาก: <<http://meewebfree.com/site/nodejs/441-learn-about-node-js-with-basic-of-node-js>>. (13 ตุลาคม 2559)
- [14] BeYourCyber.node.js กับ socket.io เพื่อทำ web socket [Online] เข้าถึงได้จาก:
<<http://meewebfree.com/site/nodejs/450-program-node-js-socket-io-web-socket>>. (12 ตุลาคม 2559)
- [15] CSS คืออะไร ซีเอสเอส คือ ภาษาที่ใช้ในการจัดรูปแบบเอกสาร HTML ให้มีความสวยงาม [Online]. เข้าถึงได้จาก: <<http://www.mindphp.com/คู่มือ/73-คืออะไร/2193-css-คืออะไร.html>>. (16 ตุลาคม 2559)
- [16] Internet of thing (IoT) [Online].เข้าถึงได้จาก:
<<http://www.sat2you.com/site/?p=2678>>. (6 กันยายน 2559)
- [17] IoT คืออะไร อะไรคือ Internet of thing (IoT) Part II [Online].
เข้าถึงได้จาก: <<http://www.zolkorn.com/article/knowledge/what-is-the-internet-of-things-iot-part-ii>>. (5 กันยายน 2559)
- [18] JavaScript คืออะไร จาวา สคริปต์ คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต [Online].เข้าถึงได้จาก:
<<http://www.mindphp.com/คู่มือ/73-คืออะไร/2187-java-javascript-คืออะไร.html>>. (12 ตุลาคม 2559)
- [19] Joe Salemi.Guide to Client/Server Databases.เรียบเรียงโดย โชคชัยเดชพรรุ่ง.
กรุงเทพฯ: ซีเอ็ดดูเคชั่น, 2538.
- [20] Ko Developer.สร้างแอปแชทด้วย Socket.io และ Node.js [Online]
เข้าถึงได้จาก: <<https://creatorthai.com/สร้างแอปแชทด้วย-socket-io-และ-mode-js/>>. (12 ตุลาคม 2559)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [21] PanJ. MEAN Stack คืออะไร? และทำไมเราถึงควรเปลี่ยนมาใช้ [Online].
เข้าถึงได้จาก: <<http://blog.panjmp.com/2014/04/what-is-mean-stack/>>. (13 ตุลาคม 2559)
- [22] PURITA.HTML คืออะไร? [Online].เข้าถึงได้จาก:
<<http://www.fusionidea.biz/html-คืออะไร/>>. (15 ตุลาคม 2559)
- [23] SIPA Programming.ทำความเข้าใจการเขียนโปรแกรมแบบ event-driven [Online].
เข้าถึงได้จาก: <<https://www.youtube.com/watch?v=4jvf4moRp2k>>. (15 ตุลาคม 2559)
- [24] WiFi คืออะไร มีประโยชน์อย่างไร [Online]. เข้าถึงได้จาก:
<<http://www.xn--12cg1cxchd0a2gzc1c5d5a.net/wifi/>>. (15 ตุลาคม 2559)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



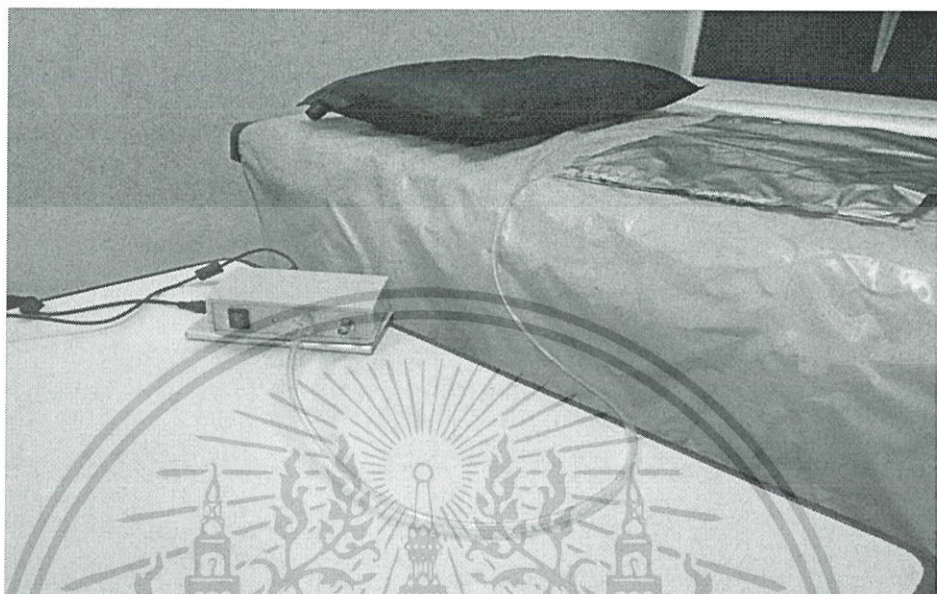
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



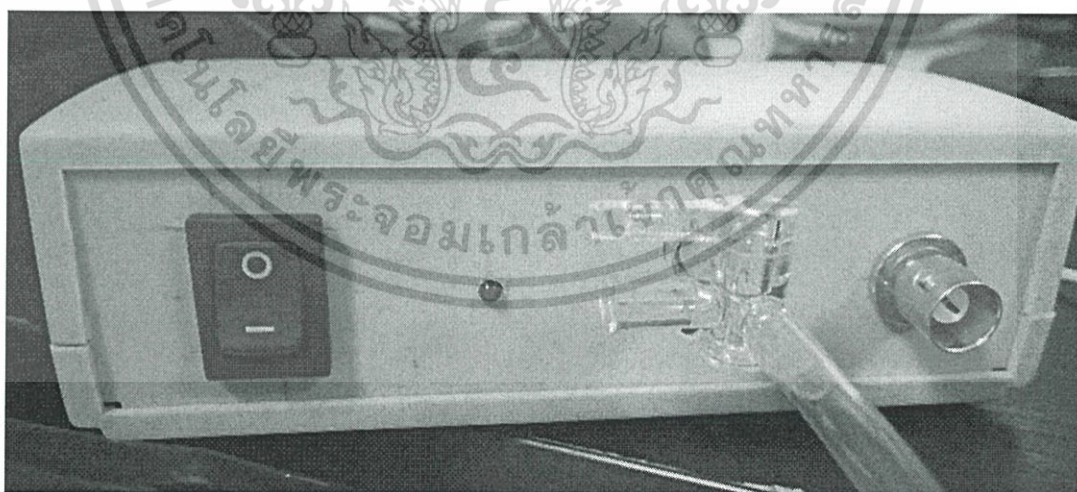
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้งานเบื้องต้น

1. เริ่มทำการติดตั้งอุปกรณ์โดยวางแผ่นตรวจจับไว้ที่หลังผู้ป่วยโดยตรงหรือสามารถวางไว้ใต้ผ้าปูเตียงก็ได้

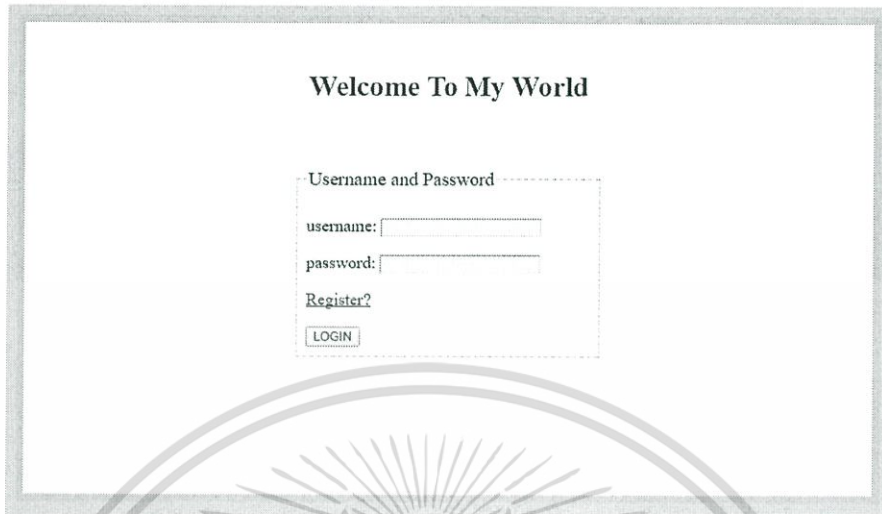


2. ทำการเปิดสวิตซ์ที่ตัวกล่อง พร้อมเสียบสาย USB เข้ากับแบตเตอรี่ชนิดพกพา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เซนเซอร์จะรับค่าแรงดันจากตัวเบาะลมแล้วทำการส่งข้อมูลไปแสดงผลผ่านหน้าเว็บโดยสามารถเข้าทางหน้าเว็บได้ที่ <http://52.221.55.254:8081/login> ทำการ Register ก็จะสามารถใช้งานได้



Welcome To My World

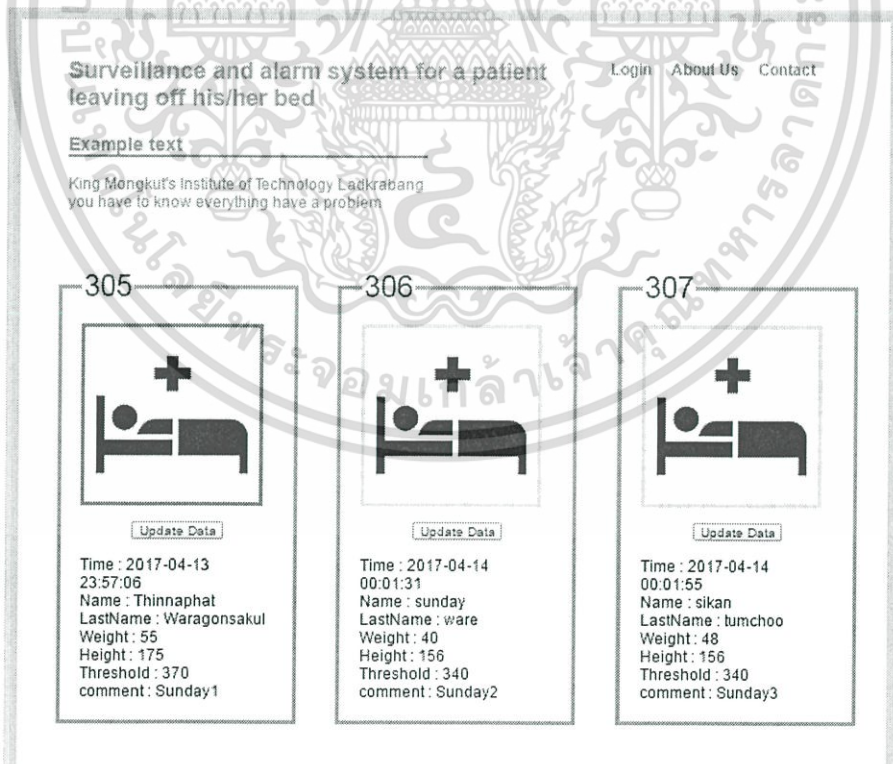
Username and Password

username:

password:

[Register?](#)

4. หน้าเว็บหน้า <http://52.221.55.254:8081/Beforehome2> จะแสดงสถานะและแจ้งเตือนเมื่อเกิดเหตุผู้ป่วยไม่อยู่บนเตียง



Surveillance and alarm system for a patient leaving off his/her bed

[Login](#) [About Us](#) [Contact](#)

Example text

King Mongkut's Institute of Technology Ladkrabang
you have to know everything have a problem

Room No.	Time	Name	Last Name	Weight	Height	Threshold	Comment
305	2017-04-13 23:57:06	Thinnaphat	Waragonsakul	55	175	370	Sunday1
306	2017-04-14 00:01:31	sunday	ware	40	156	340	Sunday2
307	2017-04-14 00:01:55	sikan	tumchoo	48	156	340	Sunday3

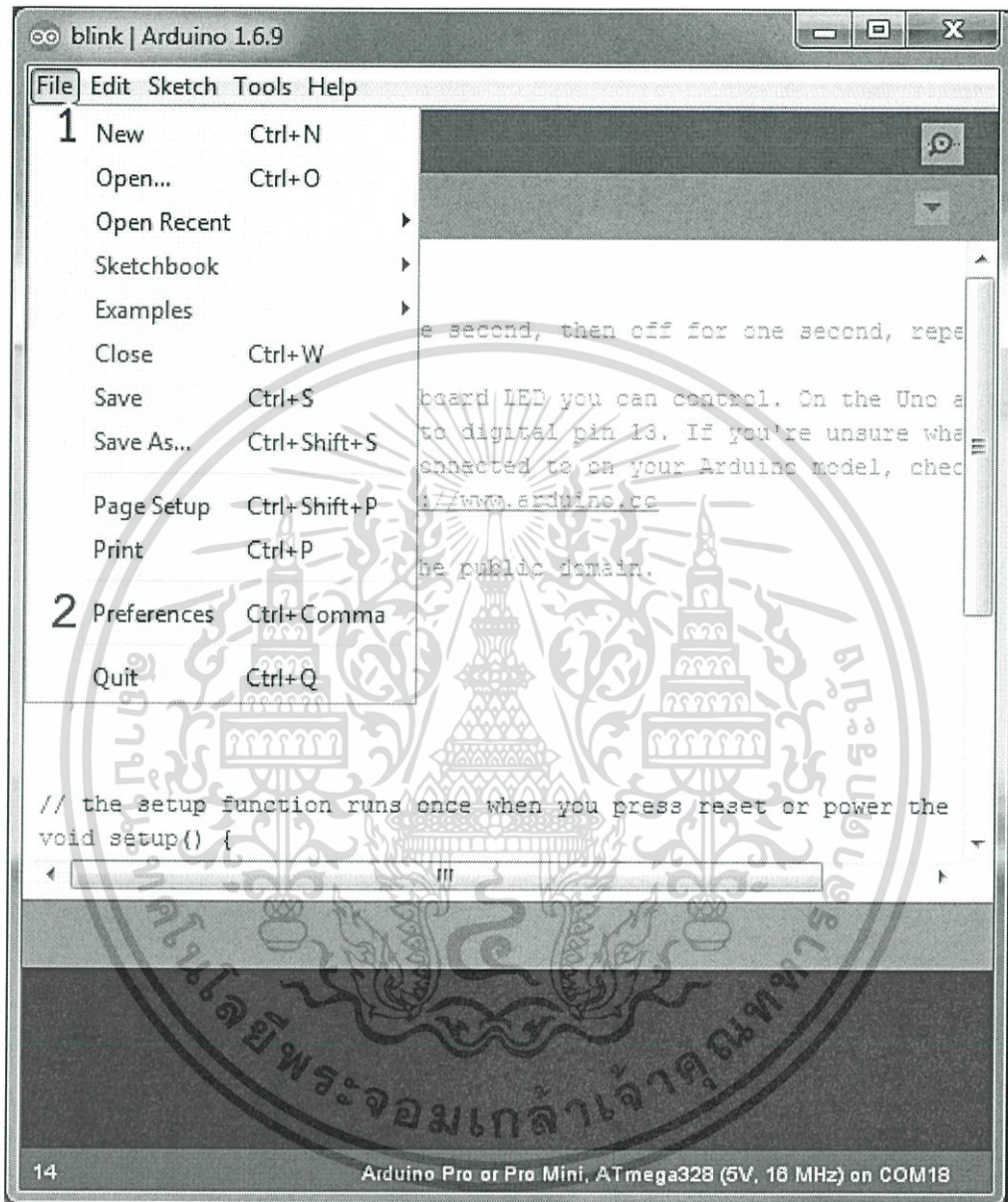
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การติดตั้ง Eps8266 Nodemcu เพื่อใช้กับ Arduino IDE รุ่นตอนนี้

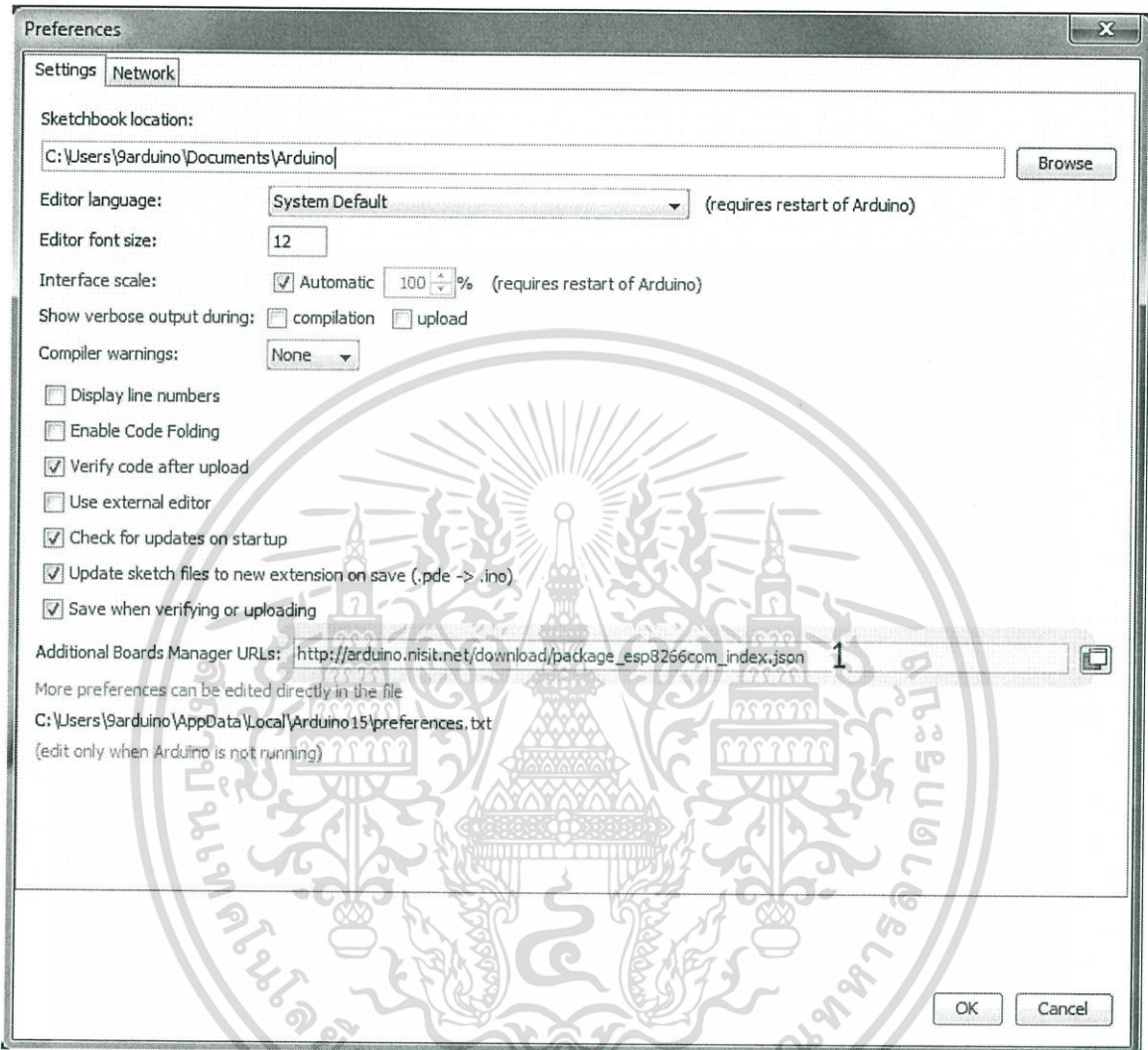
1. เริ่มจาก เลือกใช้ Arduino IDE เวอร์ชันใหม่ๆ 1.6.x ขึ้นไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

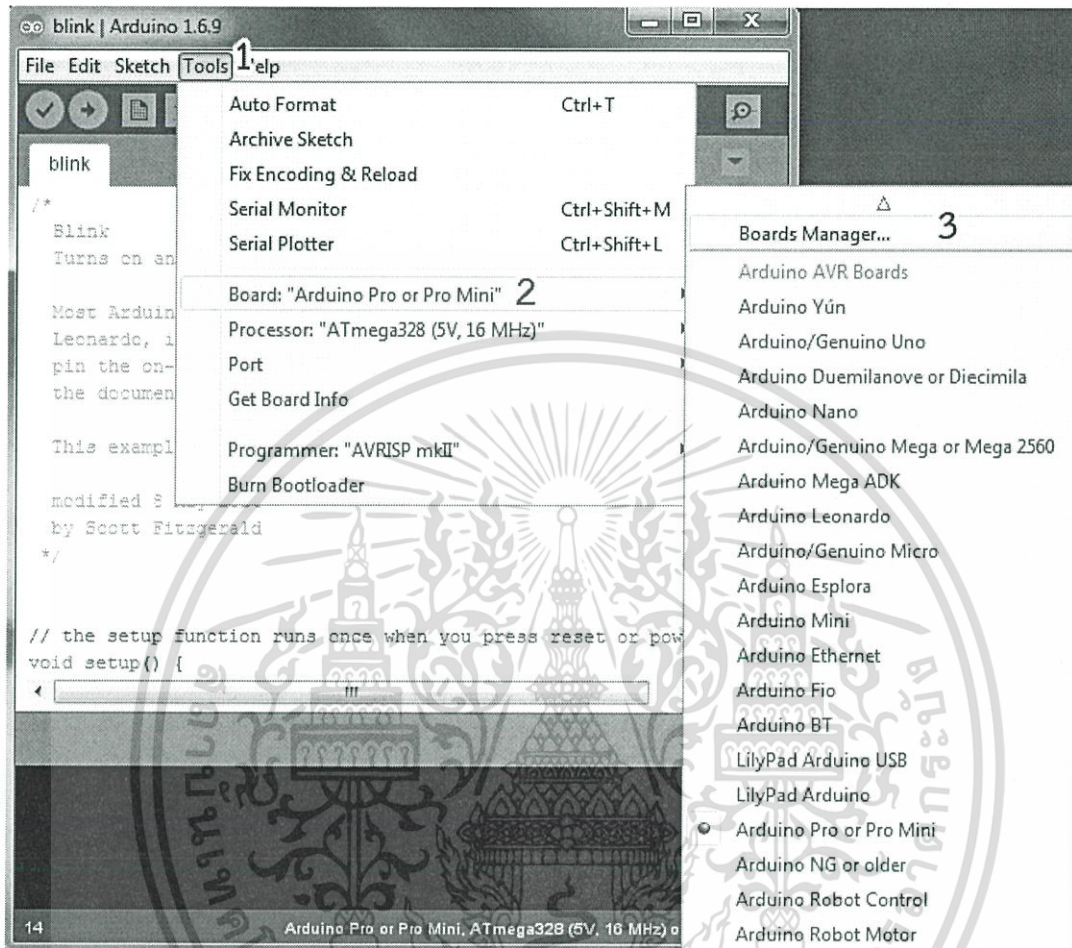
2. เปิดโปรแกรม Arduino IDE ขึ้นมา

- จากนั้นกดที่ปุ่ม File (ตามตำแหน่งที่ 1 จากรูปที่ 1)
- เลือก Preferences (ตามตำแหน่งที่ 2 จากรูปที่ 1)



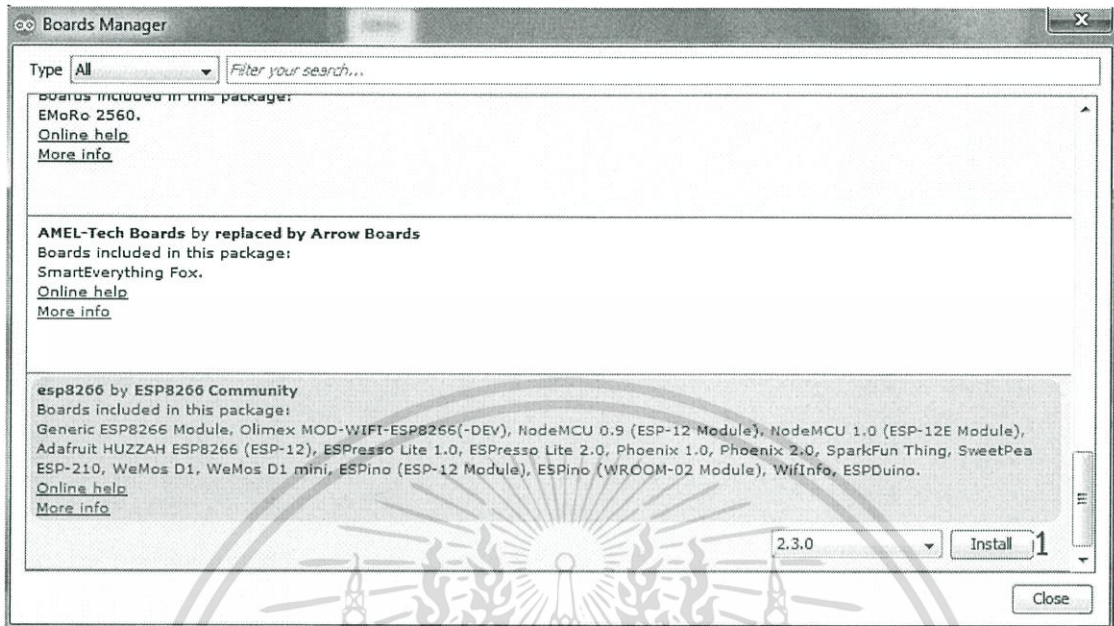
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จากนั้นไปที่ช่อง Additional Boards Manager URLs: (ตามตำแหน่งที่ 1 จากรูปที่ 2) ให้ Copy คำว่า http://arduino.nisit.net/download/package_esp8266com_index.json ใส่ลงในช่อง Additional Boards Manager แล้วกด OK

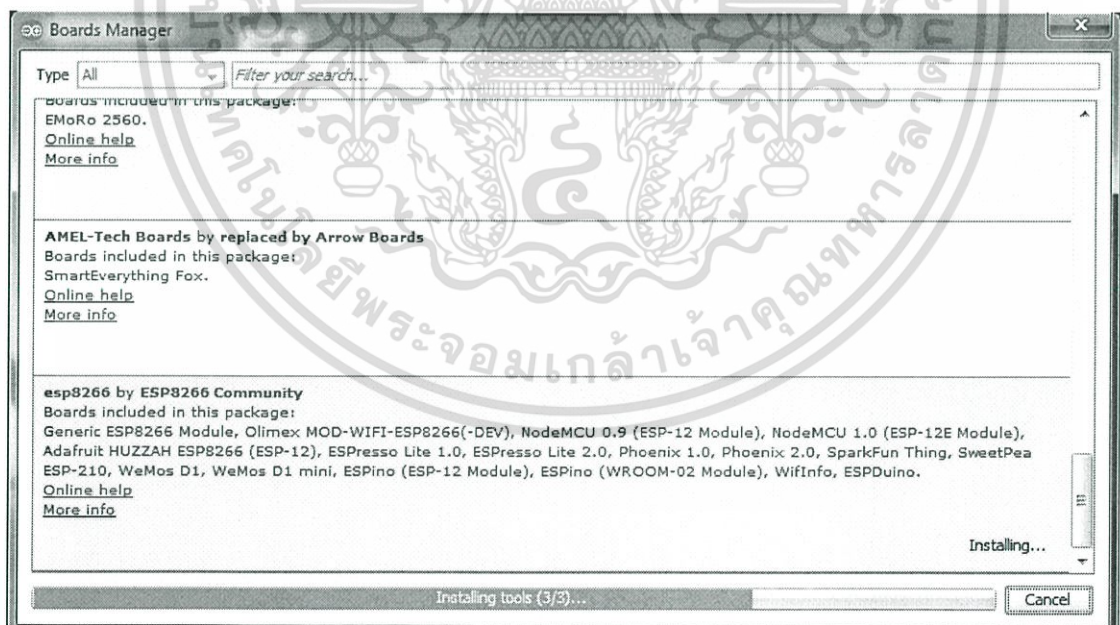


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. จากนั้นให้เลือก ที่ Tools (ตำแหน่งที่ 1) ไปที่ Board (ตำแหน่งที่ 2) ไปที่ Boards Manager (ตำแหน่งที่ 3)

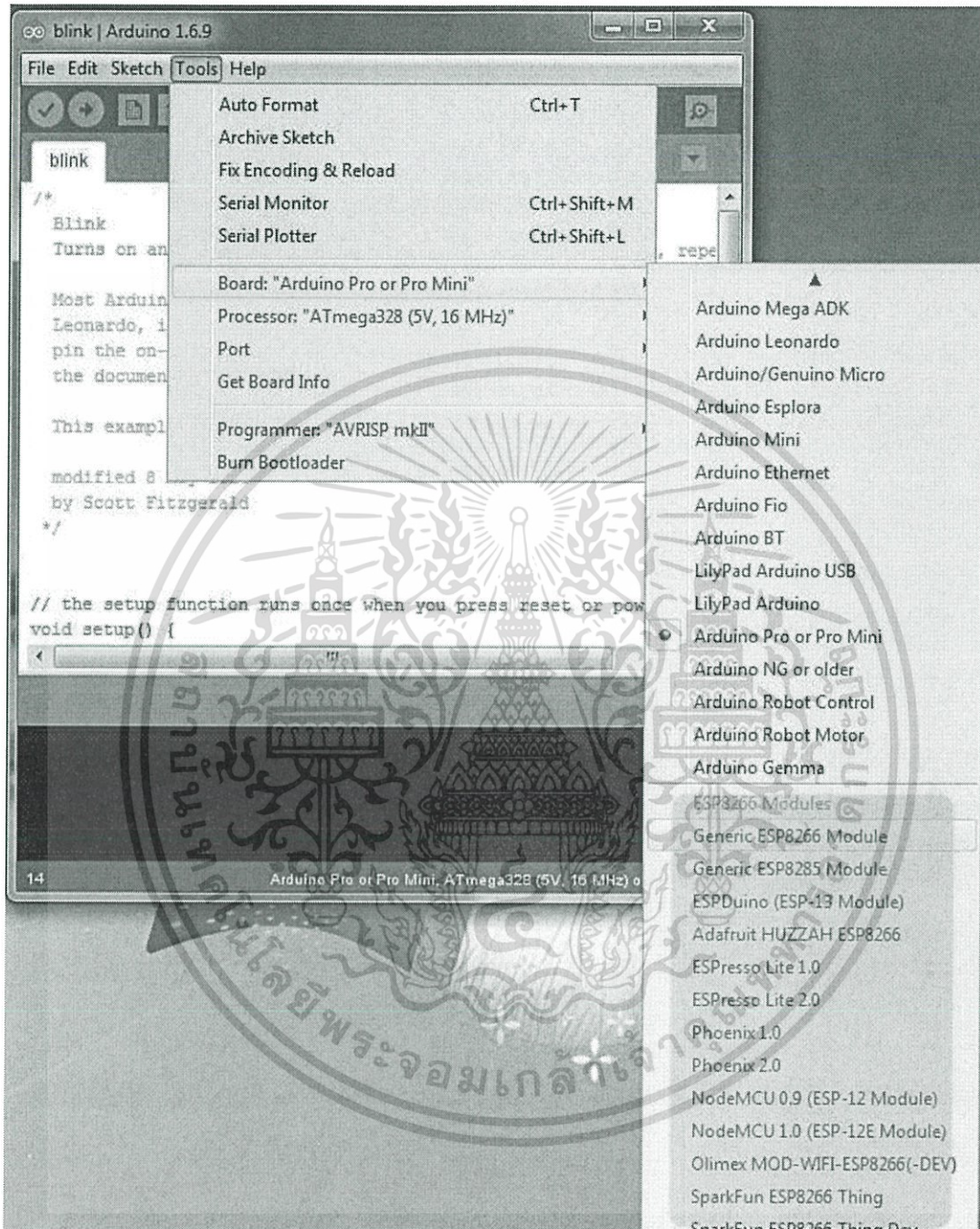


5. ค้นหา Board Esp8266 ตามภาพ จากนั้นกด Install



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. รอสักครู่ เพื่อทำการดาวน์โหลดและติดตั้ง



7. เมื่อติดตั้งเสร็จ ก็จะมี บอร์ด Esp8266 ให้เลือกตามที่ต้องการ จบขั้นตอนการติดตั้ง Nodemcu สำหรับใช้ กับ Arduino IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


การติดตั้ง Node.js


1. การจะนำ application ที่เขียนด้วย Node.js มาใช้งาน เราจะต้องติดตั้ง Node.js

Download ได้จาก: <https://nodejs.org/en/>

2. เมื่อติดตั้งเสร็จแล้ว ให้เราเปิด Command-line Interface (Command Prompt, Terminal) ขึ้นมา แล้วลองพิมพ์คำสั่ง `node -v`


Windows Installer
node-v0.10.29-x86.msi


Macintosh Installer
node-v0.10.29.pkg


Source Code
node-v0.10.29.tar.gz

Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.exe)	32-bit	64-bit
Mac OS X Installer (.pkg)	Universal	
Mac OS X Binaries (.tar.gz)	32-bit	64-bit
Linux Binaries (.tar.gz)	32-bit	64-bit
SunOS Binaries (.tar.gz)	32-bit	64-bit
Source Code	node-v0.10.29.tar.gz	

```
C:\Users\SUNDAY>node -v
v4.5.0
```

หากเลขเวอร์ชันของ Node.js แสดงขึ้นมา ก็แปลว่าเราได้ติดตั้ง Node.js เรียบร้อยแล้ว

รู้จักกับ npm

npm นั้นจะถูกติดตั้งมาพร้อมกับ Node.js เพื่อทำหน้าที่จัดการ package เสริมต่างๆ ไม่ว่าจะเป็นการติดตั้ง application หรือการติดตั้ง module ต่างๆ ที่เป็น dependency ของ application สำหรับวิธีใช้ npm นั้นก็ไม่มีอะไรยากเลย เพียงแค่เราเปิด Command-line Interface ขึ้นมา แล้วเข้าไปยัง path ที่ต้องการจะติดตั้ง(ที่อยู่ของ project เรา) จากนั้นให้พิมพ์คำสั่งนี้

```
npm install ชื่อแพ็คเกจ
```

เพียงเท่านี้ package ที่เราระบุก็จะถูกติดตั้งเรียบร้อยแล้ว เราจะใช้ package หลักๆคือ Express, Socket.io, JQuery

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้