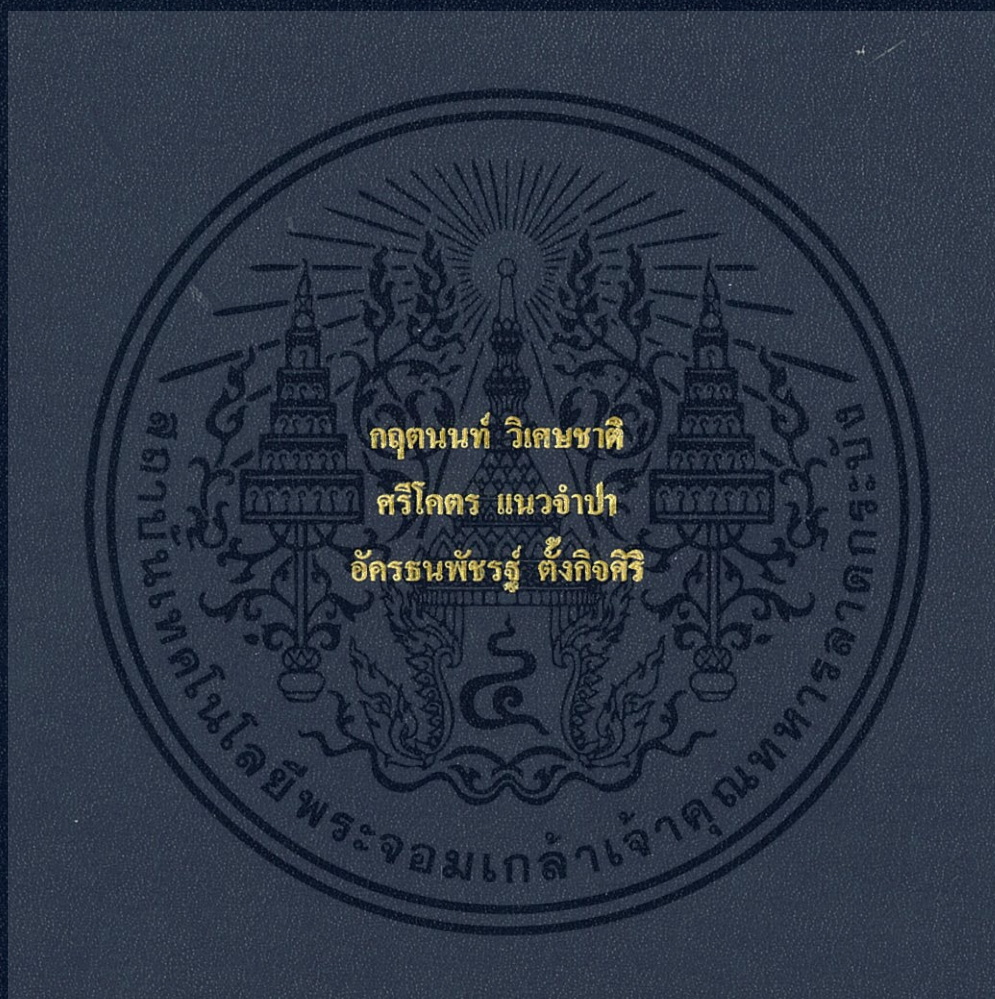


หมวกนิรภัยอัจฉริยะเพื่อการขับขี่จักรยานยนต์ให้ปลอดภัย

TECHNOLOGICAL DEVELOPMENT OF SMART HELMET FOR
MOTORCYCLIST SAFETY



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

หมวกนิรภัยอัจฉริยะเพื่อการขับขี่จักรยานยนต์ให้ปลอดภัย

TECHNOLOGICAL DEVELOPMENT OF SMART HELMET FOR
MOTORCYCLIST SAFETY



T149408



เลขหมู่.....
เลขทะเบียน 149408
วันเดือนปี ๓.7 อ.พ. 2561

b. 12885538
i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2559

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง หมวกนิรภัยอัจฉริยะเพื่อการขับขี่จักรยานยนต์ให้ปลอดภัย

TECHNOLOGICAL DEVELOPMENT OF SMART HELMET FOR MOTORCYCLIST
SAFETY

ผู้จัดทำ

1. นายกฤตนนท์ วิเศษชาติ รหัสนักศึกษา 56010020
2. นายศิริโคตร แนวจำปา รหัสนักศึกษา 56011190
3. นายอัทรชนพัชรฐ์ ตั้งกิจศิริ รหัสนักศึกษา 56011439



อาจารย์ที่ปรึกษา

(ดร.ธัญชัช ศรีภาค)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมวดนิรภัยอัจฉริยะเพื่อการขับขี่จักรยานยนต์ให้ปลอดภัย

นายกฤตพนนท์	วิเศษชาติ	รหัสนักศึกษา	56010020
นายศรี โครตร	แนวจำปา	รหัสนักศึกษา	56011190
อักรชนพัชรฐ์	ตั้งกิจศิริ	รหัสนักศึกษา	56011439
ดร.ธนัญชัย	ตรีภาค	อาจารย์ที่ปรึกษา	ปีการศึกษา 2559

บทคัดย่อ

โครงการนี้ถูกพัฒนาขึ้นเพื่อเป็นทางเลือกหนึ่งในการแก้ปัญหาความสูญเสียที่เกิดจากอุบัติเหตุบนท้องถนนอันเนื่องมาจากการไม่สวมใส่หมวกนิรภัยของผู้ขับขี่รถจักรยานยนต์ โครงการจะเน้นไปที่การพัฒนาระบบต้นแบบ ทดลองและเก็บผลการทดลอง โดยการพัฒนาจะเป็นการสร้างระบบที่ประกอบไปด้วยส่วนการป้องกันและช่วยเหลือ โดยส่วนการป้องกันจะบังคับให้ผู้ขับขี่จำเป็นต้องสวมหมวกนิรภัย ซึ่งจะมีระบบซอฟต์แวร์และฮาร์ดแวร์บนหมวกนิรภัยที่จะตรวจสอบว่าผู้ขับขี่สวมหมวกนิรภัยอยู่หรือไม่ ขั้นตอนต่อไปจะส่งข้อมูลสถานะการสวมหมวกดังกล่าวและข้อมูลที่เกี่ยวข้องอื่น ๆ อาทิเช่น ข้อมูลการเคลื่อนไหวศีรษะ ซึ่งรวมถึงความเร็วเชิงมุม ความเร่ง และขนาดของมุมก้มหรือเงย เป็นต้น ไปที่ระบบซอฟต์แวร์และฮาร์ดแวร์ที่อยู่บนรถจักรยานยนต์ รถจักรยานยนต์จะใช้ข้อมูลดังกล่าวเพื่อควบคุมการติดเครื่องยนต์ ถ้าผู้ขับขี่ไม่ได้สวมหมวกนิรภัยอยู่ ตัวระบบบนรถจักรยานยนต์จะไม่ยอมให้มีการติดเครื่องยนต์เกิดขึ้น ในส่วนการช่วยเหลือนั้นจะมีการพัฒนาระบบเซิร์ฟเวอร์ที่คอยตรวจสอบสภาพของรถจักรยานยนต์และผู้ขับขี่ โดยที่ระบบบนรถจักรยานยนต์จะคอยรวบรวมข้อมูลที่ได้จากหมวกนิรภัยและนำมารวมกับข้อมูลที่อยู่บนรถจักรยานยนต์ ได้แก่ ข้อมูลพิกัดที่อยู่ทางภูมิศาสตร์ของรถจักรยานยนต์ ข้อมูลความเร็วและความเร่งของการเคลื่อนที่ของรถจักรยานยนต์ ซึ่งข้อมูลดังกล่าวได้มาจากโมดูล GPS ที่อยู่ในระบบภายในรถจักรยานยนต์ จากนั้นจะนำส่งข้อมูลเหล่านี้ไปที่ระบบเซิร์ฟเวอร์ ซึ่งระบบเซิร์ฟเวอร์จะทำการประมวลผลและประเมินสภาพของรถจักรยานยนต์และผู้ขับขี่ว่าอยู่ในสภาพปกติหรือไม่ หรืออยู่ในระหว่างเกิดอุบัติเหตุหรือมีความเสี่ยงที่จะเกิด ซึ่งหากเกิดอุบัติเหตุขึ้น ระบบจะสามารถแจ้งขอความช่วยเหลือได้อย่างทันที่

Technological Development Of Smart Helmet For Motorcyclist

Safety

Mr. Krittanon	Wisedchart	56010020
Mr. Srikote	Naewchampa	56011190
Mr. Akkaratanapat	Tangkitsiri	56011439
Mr. Thanunchai	Threepak	Advisor

Academic Year 2016

Abstract

In this project, we proposed a solution to the problem of the loss caused by road accidents which happen to riders who don't wear helmets. The project focused on developing a prototype, testing and record the results. The project was divided into two parts: prevention part and rescue part. In the prevention part, we constructed a system which forces riders to wear a helmet. This can be done using a software and hardware system on the helmet in which the system is responsible for checking whether the helmet is worn. Next, the wearing status and other related data (e.g., movement of the head which includes angular velocity, acceleration, and tilt angle) will be sent to a software and hardware system on the motorcycle. The motorcycle will then use those data to determine if the rider showed be allowed to start the ignition. That is, if a rider doesn't wear the helmet, then the system will not allow the engine to be ignited. In the rescue part, we have developed a software system on servers which is responsible for tracking status of riders and their motorcycles. After the system on a motorcycle has got data from the helmet, it will combine those data with data from its own, which are latitude and longitude of a current position of the motorcycle, the current velocity and acceleration of the motorcycle (these data come from GPS module within the system). Then the combined data will be sent right to the servers in which the servers will process and assess the status of the drivers and their motorcycles whether they are in normal state or they are in an accident or they are in any risky situation. If so, the system then can inform and ask for help in time.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีด้วยความช่วยเหลือจากหลายฝ่ายทั้งในทางตรงและทางอ้อม ซึ่งโครงการชิ้นนี้จะสำเร็จลงไม่ได้หากปราศจากความช่วยเหลือของบุคคลเหล่านี้

ผู้ออกทุนและสนับสนุนโครงการ “หมวกกันน็อคอัจฉริยะเพื่อการจับชีพจรที่ปลอดภัย” จากสมาคมวิจัยวิทยาการขนส่งแห่งเอเชีย ATRANS (Asian Transportation Research Society) และบริษัท ไอทีเอส คอนซัลแตนท์ จำกัด (Its Consultancy Co., Ltd.)

อาจารย์ที่ปรึกษา ดร.ธนัญชัย ตรีภาค ซึ่งเป็นผู้ให้คำแนะนำคำปรึกษา และให้ความช่วยเหลือตลอดระยะเวลาการทำโครงการซึ่งทำให้การทำงานเป็นไปได้อย่างราบรื่น

อาจารย์และบุคลากรต่าง ๆ ในสาขาวิชาวิศวกรรมศาสตร์สาขาวิศวกรรมคอมพิวเตอร์ที่ได้ให้คำแนะนำสั่งสอนความรู้ต่าง ๆ มาตลอด

รวมถึงห้องปฏิบัติการสมองกลฝังตัว Embedded System Laboratory ที่ได้เอื้อเฟื้อสถานที่ในการวิจัยและพัฒนาโครงการในท้ายที่สุดขอขอบคุณบิดามารดาและครอบครัวที่ได้เลี้ยงดูสั่งสอนพร้อมให้การศึกษาก่อและกำลังใจเสมอมา

นายกตฉนันทน์ วิเศษชาติ

นายศรี โศทร แนวจำปา

นายอักรชนพัชรรัฐ ตั้งกิจศิริ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ.....	1
1.1. ที่มาและปัญหา.....	1
1.2. วัตถุประสงค์	2
1.3. ขอบเขตโครงการ	2
1.4. ขั้นตอนการดำเนินงาน	2
1.5. ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง.....	3
2.1. งานวิจัยและบทความ	3
2.2. อัลกอริทึมและวิธีการแก้ไขปัญหา	5
2.3. เทคโนโลยีที่เกี่ยวข้อง.....	13
บทที่ 3 การวิเคราะห์และออกแบบระบบ	18
3.1. ภาพรวมการทำงานภายในระบบ	18
3.2. ส่วนประกอบภายในระบบ.....	19
3.3. อุปกรณ์และเทคโนโลยีที่นำมาใช้	21
3.4. การออกแบบการทำงานและสถาปัตยกรรมของระบบ	24

บทที่ 4 การทดลองและผลการทดลอง	43
4.1. ทดสอบการเชื่อมต่อระหว่างหมวกนิรภัยและรถจักรยานยนต์	43
4.2. ทดสอบความแม่นยำในการตรวจสอบสถานะการสวมใส่หมวกนิรภัย.....	52
4.3. ทดสอบความแม่นยำในการตรวจจับความผิดปกติขณะขับขี่	54
บทที่ 5 ผลการดำเนินงาน	60
5.1. หมวกนิรภัยอัจฉริยะที่พัฒนาแล้ว	60
5.2. Web Application	61
5.3. Mobile Application	62
บทที่ 6 สรุปและข้อเสนอแนะ	66
6.1. บทสรุป	66
6.2. ปัญหา อุปสรรค และแนวทางการแก้ไข.....	66
6.3. ข้อเสนอแนะ	67
ภาคผนวก ก.....	68
1. การพัฒนาอัลกอริทึมของฟังก์ชันผลรวมความชัน (Slope-Sum Function).....	68
2. การพัฒนาอัลกอริทึมในการหาจุดยอดของคลื่น (Peak Detection).....	69
3. การพัฒนาอัลกอริทึมในการหาค่า Threshold	70
4. การหาค่าอัตราการเต้นของหัวใจโดยใช้ค่า Threshold	71
5. การพัฒนาอัลกอริทึม Butterworth Filter	72
เอกสารอ้างอิง.....	73

สารบัญรูปภาพ

หน้าที่

รูป 2.1 ภาพอธิบายการทำงานของ ECALL.....	4
รูป 2.2 ตัวอย่างกราฟ PPG (PHOTOPLETHYSMOGRAM)	5
รูป 2.3 ตัวอย่างรูปคลื่นก่อนที่จะผ่าน SLOPE-SUM FUNCTION	6
รูป 2.4 รูปคลื่นหลังที่จะผ่าน SLOPE-SUM FUNCTION	7
รูป 2.5 แนวแกนของ ACCELEROMETER.....	9
รูป 2.6 แนวแกนของ GYROSCOPE.....	9
รูป 2.7 แนวแกนของ GYROSCOPE.....	10
รูป 2.8 การเปิด CONNECTION ระหว่างอุปกรณ์ CENTRAL และ PERIPHERAL	12
รูป 2.9 BLUETOOTH LOW ENERGY STACK.....	13
รูป 2.10 อุปกรณ์วัดชีพจร PULSE SENSOR	14
รูป 2.11 ตัวอย่างการใช้งานของ PULSE SENSOR.....	14
รูป 2.12 ตัวอย่างสัญญาณชีพจรที่อ่านจาก SENSOR.....	14
รูป 2.13 แกนอ้างอิง	15
รูป 2.15 BLUETOOTH SMART หรือ BLUETOOTH LOW ENERGY (BLE)	16
รูป 2.16 ภาพอธิบายการทำงานของ GNSS	17
รูป 2.17 ภาพ SMART PHONE ที่ใช้ ANDROID เป็นระบบปฏิบัติการ	17
รูป 3.1 ส่วน ประกอบหลักของระบบ	19
รูป 3.2 NRF51822 ชิพ ARM CORTEX-M0 WITH BLE MODULE	21
รูป 3.3 ตัวอย่างการใช้งานของ PULSESENSOR.....	22
รูป 3.4 บอร์ด IMU ที่ใช้ชิพ MPU9250	22
รูป 3.5 ภาพหน่วยประมวลผล LINKIT SMART 7688 DUO	23
รูป 3.6 ส่วนประกอบฮาร์ดแวร์ภายในหมวกนิรภัย.....	24
รูป 3.7 ส่วนประกอบฮาร์ดแวร์ภายในหมวกนิรภัยและการเชื่อมต่อระหว่างฮาร์ดแวร์แต่ละส่วน...25	
รูป 3.8 การติดต่อส่วนประกอบฮาร์ดแวร์ภายในหมวกนิรภัย.....	27
รูป 3.9 ERROR AND EVENT CHARACTERISTIC	28
รูป 3.10 ACCELERATION CHARACTERISTIC.....	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.11 ACCELERATION CHARACTERISTIC.....	28
รูป 3.12 ภาพประกอบการออกแบบในส่วนของการควบคุมรถจักรยานยนต์.....	29
รูป 3.13 ภาพประกอบการวางส่วนรับผิดชอบให้แก่ฮาร์ดแวร์.....	30
รูป 3.14 ภาพประกอบการทำงานในส่วนของตัวควบคุมรถจักรยานยนต์.....	32
รูป 3.15 ภาพประกอบการออกแบบในส่วน MOBILE APPLICATION.....	34
รูป 3.16 ภาพประกอบการทำงานบน MOBILE APPLICATION.....	35
รูป 3.17 ภาพประกอบการออกแบบฐานข้อมูลภายใน.....	36
รูป 3.18 ภาพรวมการทำงานขอระบบ.....	37
รูป 3.19 สถาปัตยกรรมของระบบ.....	38
รูป 3.20 ส่วนประกอบโครงสร้างของระบบ โดยละเอียด.....	40
รูปที่ 4.1 กราฟความสัมพันธ์ของการทดสอบ ระยะ 1 เมตรกับเวลา.....	45
รูปที่ 4.3 กราฟความสัมพันธ์ของการทดสอบระยะ 10 เมตรกับเวลา.....	49
รูปที่ 4.4 กราฟความสัมพันธ์ของการทดสอบระยะ 50 เมตรกับเวลา.....	51
รูปที่ 4.5 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบการไม่สวมหมวก 20 ครั้ง.....	52
รูปที่ 4.6 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบการสวมหมวก 20 ครั้ง.....	53
รูปที่ 4.7 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบการเคลื่อนไหวปกติ 20 ครั้ง.....	54
รูปที่ 4.8 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบการเคลื่อนไหวปกติ 20 ครั้ง.....	55
รูปที่ 4.9 กราฟความสัมพันธ์ระหว่างอัตราการเต้นของหัวใจระหว่าง SMART HELMET และ FITBIT ..	59
รูปที่ 5.1 อุปกรณ์หมวดนิรภัยที่พัฒนา.....	60
รูปที่ 5.2 หน้าจอ WEB APPLICATION ที่พัฒนา.....	61
รูปที่ 5.3 หน้าจอเข้าสู่ระบบ.....	62
รูปที่ 5.4 หน้าจอเชื่อมต่ออุปกรณ์.....	63
รูปที่ 5.5 หน้าแสดงบันทึกการเดินทางของผู้ใช้.....	64
รูปที่ 5.6 หน้าแสดงพฤติกรรมการขับขี่ของผู้ใช้.....	65

สารบัญตาราง

ตารางที่ 4.1 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 1 เมตร	44
ตารางที่ 4.2 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 1 เมตร	45
ตารางที่ 4.3 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 3 เมตร	46
ตารางที่ 4.4 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 3 เมตร	47
ตารางที่ 4.5 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 10 เมตร	48
ตารางที่ 4.6 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 10 เมตร	49
ตารางที่ 4.7 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 50 เมตร	50
ตารางที่ 4.8 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 50 เมตร	51
ตารางที่ 4.9 ผลการทดสอบการไม่สวมหมวก	52
ตารางที่ 4.10 ผลการทดสอบการสวมหมวก	53
ตารางที่ 4.11 ผลการทดสอบการเคลื่อนไหวปกติ	54
ตารางที่ 4.12 ผลการทดสอบมุมเอียงของศีรษะที่ผิดปกติ	55
ตารางที่ 4.13 ผลการทดสอบความเร่งที่ผิดปกติ	56
ตารางที่ 4.14 ผลการทดสอบความแม่นยำในการตรวจวัดอัตราการเต้นหัวใจ	58

บทที่ 1

บทนำ

1.1. ที่มาและปัญหา

จากข้อมูลทางสถิติอุบัติเหตุการจราจรทางบก โดยสำนักงานสถิติแห่งประเทศไทย ประจำปี พ.ศ. 2550 ถึง 2557 แสดงให้เห็นว่า อุบัติเหตุบนถนนที่เกิดกับรถจักรยานยนต์คิดเป็นร้อยละ 25 ของยอดการรับแจ้งอุบัติเหตุทั้งหมด โดยสูงเป็นอันดับหนึ่งหากจำแนกตามประเภทผู้ใช้ทางซึ่งอันดับสองคือรถยนต์นั่งส่วนตัวและจากรายงานสถานการณ์โลกด้านความปลอดภัยทางถนน โดยองค์การอนามัยโลก ประจำปี พ.ศ. 2558 ได้กล่าวว่า จำนวนการเสียชีวิตที่เกิดจากการขับขี่จักรยานยนต์นั้นคิดเป็นร้อยละ 23 ของจำนวนการเสียชีวิตจากการจราจรทางถนนทั้งหมด โดยสูงเป็นอันดับหนึ่งถัดจากกลุ่มคนเดินเท้าซึ่งสูงเป็นอันดับสอง ซึ่งจากปัญหาเรื่องอุบัติเหตุและการเสียชีวิตนี้เองที่ทำให้คณะผู้จัดทำโครงการตระหนักว่าการใช้รถใช้ถนนในประเทศไทยนั้นมีความอันตรายมากแค่ไหน

โดยในรายงานจากองค์การอนามัยโลกดังกล่าวยังได้บอกอีกว่า การสวมหมวกนิรภัยช่วยลดความเสี่ยงในการเสียชีวิตได้เกือบร้อยละ 40 และความเสี่ยงในการบาดเจ็บสาหัสได้ประมาณร้อยละ 70 ซึ่งตรงจุดนี้เองที่ทำให้คณะผู้จัดทำโครงการคิดว่า หากเราสามารถช่วยหาวิธีการหรืออุปกรณ์ที่ทำให้ผู้ขับขี่จักรยานยนต์หันมาสวมใส่หมวกนิรภัยกันมากขึ้น ก็จะเป็นการช่วยลดจำนวนผู้เสียชีวิตจากการจราจรให้น้อยลงไปได้อย่างมาก

ด้วยสถิติและข้อมูลข้างต้นจึงนำมาซึ่งการสร้างระบบหมวกนิรภัยอัจฉริยะที่จะช่วยในการตรวจสอบว่าผู้ขับขี่สวมหมวกนิรภัยอยู่หรือไม่ โดยหากพบว่าผู้ขับขี่ไม่ได้สวมหมวกนิรภัยอยู่ตัวรถจักรยานยนต์ก็จะส่งจะไม่ยอมให้มีการสตาร์ทเครื่องยนต์เพื่อให้ผู้ขับขี่สวมหมวกนิรภัยก่อนที่จะทำการขับขี่รถจักรยานยนต์ นอกจากนี้ในระบบยังมีการเก็บข้อมูลของการขับขี่ระหว่างการเดินทางอีกด้วย เช่น ความเร็วและตำแหน่งของรถจักรยานยนต์ ลักษณะการเคลื่อนที่ของหัวผู้สวมใส่หมวกนิรภัย เป็นต้น โดยข้อมูลดังกล่าวจะถูกนำไปวิเคราะห์เพื่อตรวจสอบว่าในขณะที่ผู้ขับขี่ได้อยู่ในระหว่างการเกิดอุบัติเหตุขึ้นหรือไม่เพื่อจะได้แจ้งเข้าสู่ศูนย์รับแจ้งอุบัติเหตุต่อไป

1.2. วัตถุประสงค์

- 1) สร้างหมวกนิรภัยตัวอย่างที่มีความสามารถในการตรวจสอบว่ากำลังถูกสวมโดยบุคคลอยู่หรือไม่และสามารถบอกลักษณะการเคลื่อนที่ของหัวผู้สวมใส่ได้
- 2) สร้างระบบวิเคราะห์ข้อมูลการขับขี่และประเมินเหตุการณ์การเกิดอุบัติเหตุจากการเดินทางด้วยรถจักรยานยนต์ขึ้นมา

1.3. ขอบเขตโครงการ

- 1) หมวกนิรภัยตัวอย่างที่สร้างขึ้นมาจะเกิดจากการตัดแปลงหมวกที่หาซื้อได้ตามท้องตลาดทั่วไป แต่จะมีลักษณะคือคลุมทั้งศีรษะของผู้สวมใส่
- 2) การตัดแปลงตัวรถจักรยานยนต์จะใช้วิธีการเพิ่มวงจรรายนอกเข้าไป ไม่มีการเข้าไปตัดแปลงตัวโปรแกรมภายในระบบควบคุมของรถจักรยานยนต์
- 3) การประเมินเหตุการณ์การเกิดอุบัติเหตุจะใช้ การวิเคราะห์ข้อมูลที่ได้จากการทดลองกับรถจักรยานยนต์และผู้ขับขี่ที่สวมใส่หมวกนิรภัย

1.4. ขั้นตอนการดำเนินงาน

- 1) กำหนดขอบเขตและความต้องการของระบบ
- 2) วิเคราะห์และออกแบบระบบ
- 3) จัดหาอุปกรณ์
- 4) พัฒนาและทดสอบระบบ

1.5. ประโยชน์ที่คาดว่าจะได้รับ

- 1) ได้รับความสนใจจากภาคส่วนที่มีผลในการผลักดันการพัฒนาโครงการต่อไป
- 2) เพิ่มอัตราการสวมใส่หมวกนิรภัยของผู้ขับขี่จักรยานยนต์เพิ่มมากขึ้น
- 3) ลดอัตราการเสียชีวิตที่เกิดกับผู้ขับขี่รถจักรยานยนต์ลดลง

บทที่ 2

ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1. งานวิจัยและบทความ

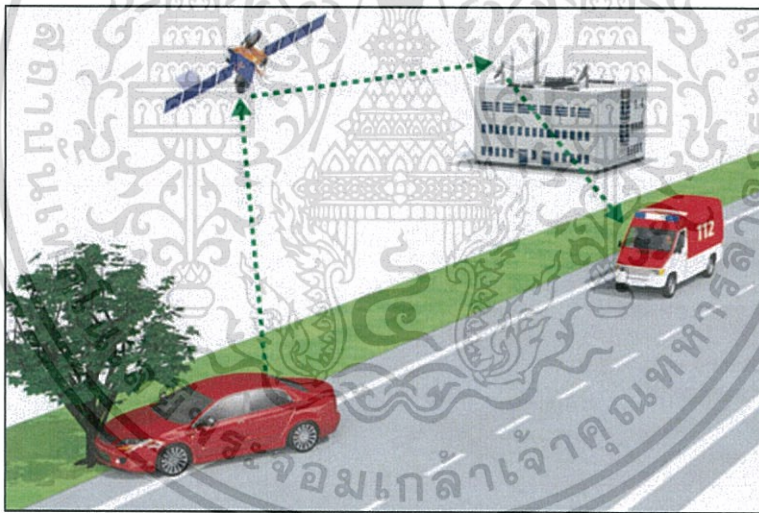
2.1.1. Smart helmet by international research journal of engineering and technology

โครงการนี้ต้องการสร้างวงจรที่จะใช้ภายในระบบ Smart Helmet โดยความสามารถของระบบดังกล่าวคือ ป้องกันไม่ให้ผู้ใช้สารถนในขณะที่ยังไม่ได้สวมหมวกนิรภัย ซึ่งการทำงานเบื้องต้นคือ ที่หมวกนิรภัยจะมี RF Transmitter และที่รถจะมี RF Receiver ติดอยู่กับตัวควบคุมการจ่ายไฟของเครื่องยนต์และในสถานะปกติจะไม่สามารถติดเครื่องยนต์ได้ แต่เมื่อผู้ขับขี่สวมหมวกตัว RF Transmitter ที่หมวกจะปล่อยสัญญาณ RF ออกมาจากนั้นเมื่อ RF Receiver ที่ตัวรถได้รับสัญญาณดังกล่าว ระบบจะปลดล๊อคระบบติดเครื่องยนต์ทำให้สามารถติดเครื่องยนต์ได้รถได้ตามปกติ

จากแนวคิดที่เกี่ยวข้องข้างต้น ทำให้เราสามารถนำมาประยุกต์ใช้ในเรื่องการติดต่อกันระหว่างหมวกนิรภัยและระบบติดเครื่องยนต์ได้ แต่อย่างไรก็ตามคลื่น RF ดังกล่าวก็ยังมีข้อเสียที่ทำให้ไม่เหมาะกับโครงการของเราอยู่คือ การส่งข้อมูลโดยใช้เพียงมาตรฐานของ RF อย่างเดียวนั้นทำให้เป็นการยากหากเราต้องการใช้ Protocol พิเศษบางอย่างในภายหลัง เช่น การทำ RF ให้กลายเป็น Mesh Network เป็นต้น ทำให้เราคิดว่าแนวคิดการใช้ RF ติดต่อกันระหว่างหมวกกันน็อกและระบบติดเครื่องยนต์นี้ยังไม่เพียงพอต่อความต้องการในอนาคต

2.1.2. eCall (Emergency Call)

เป็นระบบแจ้งอุบัติเหตุที่ถูกติดตั้งบนยานพาหนะในทวีปยุโรป โดยกรณีศึกษาอยู่ว่าเมื่อมีผู้ขับขี่กำลังอยู่ในเขตที่ปลอดภัยเช่นการเดินทางข้ามรัฐหรือจังหวัดและถ้าผู้ขับขี่เกิดอุบัติเหตุและผู้ขับขี่ไม่สามารถติดต่อขอความช่วยเหลือได้ด้วยตนเอง เมื่อเกิดอุบัติเหตุโอกาสที่ผู้ประสบอุบัติเหตุจะได้รับความช่วยเหลือนั้นจะมาจากผู้ขับขี่อื่นที่ได้ขับมาในเส้นทางเดียวกันและพบเห็นผู้ประสบอุบัติเหตุแล้วจึงจะสามารถติดต่อขอความช่วยเหลือได้ซึ่งในกรณีดังกล่าวนี้แม้ผู้ขับขี่จะไม่ได้รับการบาดเจ็บถึงแก่ชีวิตในทันทีแต่ก็อาจจะเสียชีวิตได้ในภายหลังหากได้รับการรักษาช้าเกินไป ด้วยเหตุนี้เอง eCall จึงมีบทบาทในเรื่องการตรวจจับอุบัติเหตุและวัดความรุนแรง เมื่อเกิดเหตุขึ้นตัวระบบจะส่งสัญญาณไปแก่ศูนย์รับแจ้งเหตุที่ใกล้ที่สุดและระบุพิกัดทางภูมิศาสตร์ให้ ซึ่งช่วยลดช่องว่างในช่วงเวลาขอความช่วยเหลือได้และลดจำนวนผู้เสียชีวิตจากการไม่ได้รับการช่วยเหลือไม่ทันเวลาได้อย่างมาก ซึ่งในปัจจุบันนี้ตัวระบบ eCall ถูก Build-In ลงใน GNSS ของรถยนต์ ซึ่งใช้งาน GPS และ GLONASS ในการทำงานร่วมกันเพื่อหาตำแหน่งที่แม่นยำขึ้น นอกจากนี้ยังถูกผลักดันให้เป็นมาตรฐานของทวีปยุโรปที่ว่ารถยนต์ทุกคันจะต้องมีระบบ eCall ติดอยู่



รูป 2.1 ภาพอธิบายการทำงานของ eCall

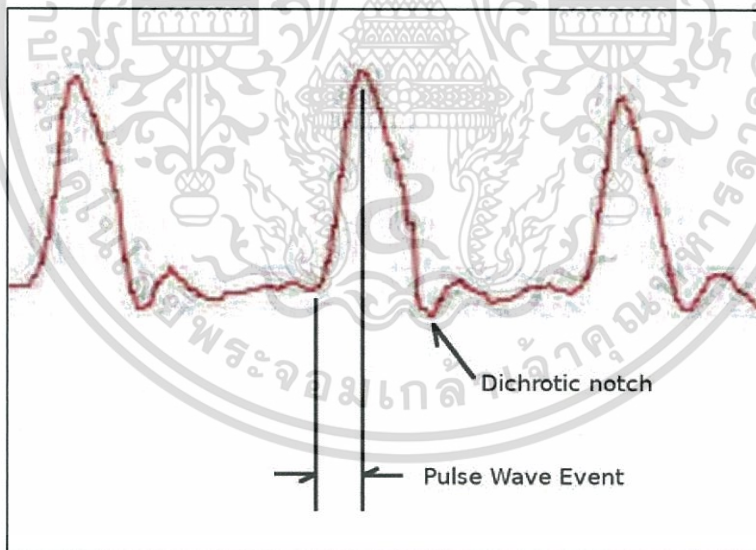
2.2. อัลกอริทึมและวิธีการแก้ไขปัญหา

2.2.1. การวัดอัตราการเต้นของหัวใจ

ในการตรวจสอบสถานะการสวมใส่หมวกนิรภัยของผู้ขับขี่ ผู้พัฒนาได้ตัดสินใจใช้อัตราการเต้นของหัวใจมาเป็นเกณฑ์การตัดสินใจ เนื่องจากการตรวจสอบสถานะด้วยวิธีอื่นอย่างเดียวยังไม่สามารถบอกได้ว่าศีรษะของผู้ขับขี่อยู่ในหมวกนิรภัยจริงหรือไม่ ตัวอย่างเช่น ผู้ขับขี่ไม่ได้เป็นผู้สวมใส่หมวกนิรภัย หากแต่ใช้วัตถุบางอย่างเพื่อให้สวิตช์ภายในหมวกถูกกด วิธีการดังกล่าวจะใช้ไม่ได้กับการวัดอัตราการเต้นของหัวใจ เนื่องจากตัววัตถุไม่มีชีพจร ภายในหัวข้อนี้จะกล่าวถึงเทคโนโลยี ทฤษฎี หลักการ และความรู้ที่ใช้ในการพัฒนาส่วนวัดอัตราการเต้นของหัวใจ

2.2.2. การวัดชีพจรโดยใช้ Photoplethysmograph

คือเครื่องมือวัดปริมาตรของอวัยวะโดยใช้แสง หลักการทำงานคร่าว ๆ คือการปล่อยให้แสงจากแหล่งกำเนิด ส่องลงไปที่ผิวของอวัยวะ จากนั้นวัดการเปลี่ยนแปลงของการดูดกลืนแสง ด้วยวิธีนี้เมื่อหัวใจทำการสูบฉีดเลือดมาหล่อเลี้ยงที่เนื้อเยื่อข้างใต้ผิวหนัง การดูดกลืนแสงภายใต้ผิวหนังจะเปลี่ยนไป โดยผลลัพธ์ที่ได้จาก Photoplethysmograph คือกราฟที่เรียกว่า Photoplethysmogram หรือ PPG



รูป 2.2 ตัวอย่างกราฟ PPG (Photoplethysmogram)

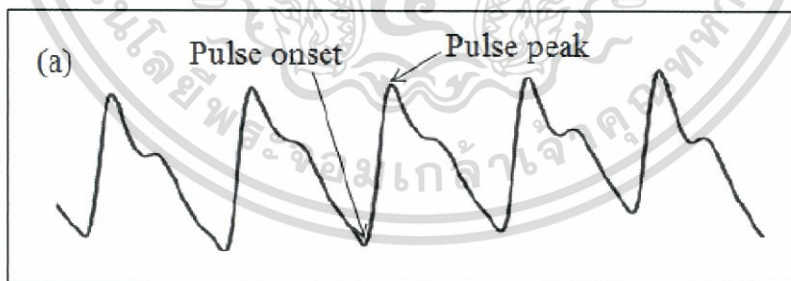
โดย PPG ที่ได้จากมนุษย์ที่มีชีวิตจะมีลักษณะดัง รูป 2.2 คือมีลักษณะเป็นคลื่น มีจุดของยอด และท้องของคลื่นมีรูปร่างที่สามารถคาดเดาได้วนซ้ำไปเรื่อย ๆ ด้วยรูปแบบดังกล่าว PPG สามารถนำมาใช้วัดอัตราการเต้นของหัวใจได้ ดังนั้นจึงสามารถนำมาตรวจสอบได้เช่นกันว่า สิ่งที่ Photoplethysmograph กำลังวัดอยู่ เป็นมนุษย์ที่มีชีวิตหรือไม่

แต่เนื่องจาก PPG ที่ได้โดยใช้ Sampling Rate สูง ๆ นั้นมักจะมีคลื่นรบกวนหรือ Noise อยู่เยอะ การกำจัดคลื่นรบกวนดังกล่าวสามารถทำได้โดยใช้ตัวกรองคลื่นความถี่ต่ำ หรือ Low-Pass Filter เข้ามาช่วย โดยในโครงการนี้ ผู้พัฒนาได้ใช้ Sampling Rate เป็น 500 Hz และใช้ Butterworth Filter เป็น Algorithm ของ Low-Pass Filter ดังกล่าว ซึ่งรายละเอียดการพัฒนา Butterworth Filter ในรูปของ Algorithm สามารถดูได้ที่ภาคผนวก ก.

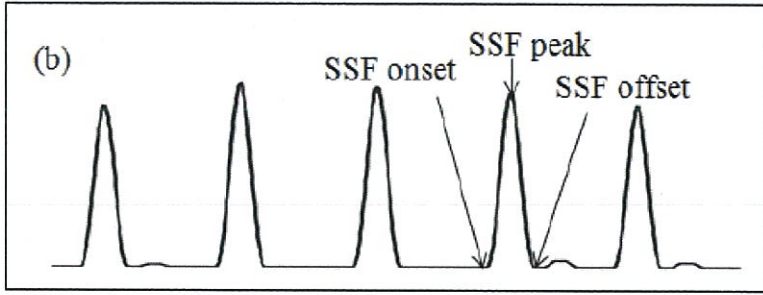
หลังจากได้ PPG ที่ถูกลดคลื่นรบกวนมาแล้ว จะมีขั้นตอนเพื่อหาอัตราการเต้นของหัวใจคือการปรับปรุงรูปร่างของคลื่นเพื่อให้ง่ายต่อการตรวจนับยอดคลื่น ต่อมาเป็นการตรวจนับยอดคลื่น สุดท้ายคือการคำนวณอัตราการเต้นของหัวใจในหน่วย BPM (Beats Per Minute) โดยใช้ Threshold

2.2.2.1. การปรับปรุงรูปร่างของคลื่นเพื่อนำมาใช้ในการตรวจนับยอดคลื่น

เป้าหมายคือ ปรับปรุงรูปร่างของคลื่นให้คอมพิวเตอร์สามารถแยกยอดคลื่นแต่ละยอดออกจากกันได้ชัดเจนขึ้น โดยวิธีการที่ใช้คือนำ PPG ที่ได้ มาเข้าใน Slope-Sum Function หรือ SSF เพื่อเพิ่มความชันของคลื่นในช่วงขาขึ้น กล่าวคือ ช่วงที่ความชันของ PPG เป็นบวกนั่นเอง และกดช่วงที่เหลือของ PPG ไม่ให้แสดงออกมาในกราฟผลลัพธ์ ตัวอย่างจะเห็นได้จาก รูป 2.3 ซึ่งเป็นรูปคลื่นก่อนที่จะผ่าน SSF และ รูป 2.4 ซึ่งเป็นรูปคลื่นหลังจากผ่าน SSF แล้ว



รูป 2.3 ตัวอย่างรูปคลื่นก่อนที่จะผ่าน Slope-Sum Function



รูป 2.4 รูปคลื่นหลังที่จะผ่าน Slope-Sum Function

โดยรูปสมการของ Slope-Sum Function (SSF) สามารถแสดงได้ดังนี้

$$SSF_i = \sum_{k=i-w}^i \Delta x_k \text{ where } \Delta x_k = \begin{cases} \Delta s_k & \text{if } \Delta s_k > 0 \\ 0 & \text{if } \Delta s_k \leq 0 \end{cases} \quad (2.1)$$

โดยที่ w คือความยาวของ Window ที่ใช้เก็บค่าความชันของกราฟ และ SSF คือ สัญญาณ PPG ที่ผ่านการกรองแล้ว

2.2.2.2. การตรวจจับยอดคลื่น

ด้วย Hill-Climbing Algorithm ตัวโปรแกรมจะทำการตรวจสอบสัญญาณ PPG ที่ผ่าน SSF มาว่า ใช้ Local Maximum หรือไม่ โดย Local Maximum สามารถนิยามได้ดังนี้

$$s_i \text{ is local maximum, if } s_{i-1} < s_i \text{ and } s_{i+1} < s_i \quad (2.2)$$

จากนั้นจะบันทึกค่า Local Maximum ดังกล่าวไว้ และเมื่อมีสัญญาณ PPG ตัวใหม่ที่เป็น Local Maximum เข้ามา ค่าดังกล่าวจะถูกนำมาเปรียบเทียบกับ Local Maximum ค่าเดิมที่ถูกบันทึกไว้ โดยถ้าค่าใหม่มากกว่าหรือเท่ากับค่าเดิม หรือมีค่าน้อยกว่าในขอบเขตที่กำหนดไว้ จะถือว่าเป็นยอดคลื่น หรือหากโปรแกรมไม่สามารถตรวจหายอดคลื่นได้ภายใน 3 วินาที ค่า Local Maximum ใหม่จะถือเป็นยอดคลื่น โดยทันที รายละเอียดการพัฒนา Algorithm ดังกล่าวสามารถดูได้ที่ภาคผนวก ก.

2.2.2.3. การคำนวณ Heart Rate ในหน่วย BPM (Beats Per Minute) โดยใช้ค่า Threshold

หลักการคือ ตรวจสอบค่าสัญญาณ PPG ที่ผ่าน SSF มา ว่ามีค่ามากกว่าค่ากลางค่าหนึ่งที่กำหนดไว้หรือไม่ โดยค่ากลางดังกล่าวจะถูกเรียกว่า Threshold ซึ่งหากมากกว่าค่าสัญญาณ PPG จะถูกบันทึกไว้ และเมื่อเกิดสัญญาณ PPG ที่มีค่ามากกว่า Threshold อีกครั้งเราจึงจะสามารถคำนวณค่าอัตราการเต้นของหัวใจจากสมการที่ 2.3

$$bpm = t_{minute} / \Delta t_i \quad (2.3)$$

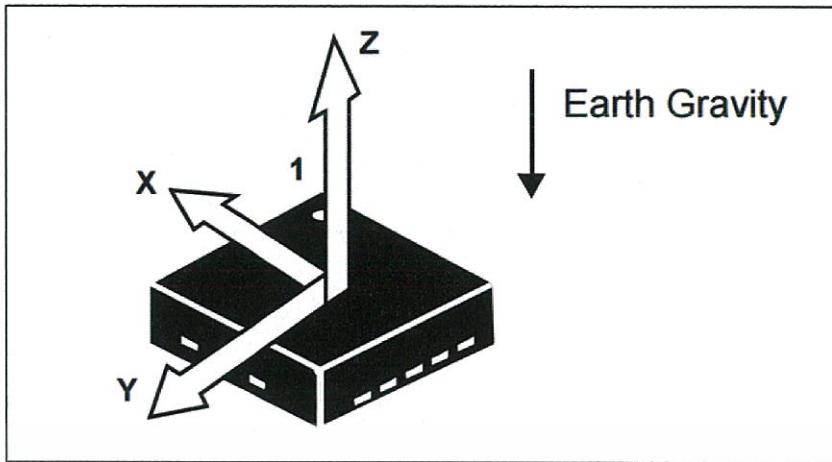
โดยที่ t คือเวลาที่ผ่านไปนับจากสัญญาณ PPG ตัวแรกที่เกินค่า Threshold จนถึงสัญญาณ PPG ตัวถัดมาที่เกินค่า Threshold ส่วน t_{minute} คือค่าเวลาใน 1 นาที ซึ่งมีหน่วยเดียวกับ t_i ค่า Threshold ที่ใช้ในโครงการนี้จะมีค่าเท่ากับ 70 เปอร์เซ็นต์ของค่าเฉลี่ยเลขคณิตของยอดคลื่นห้าตัวล่าสุด รายละเอียดการพัฒนา Algorithm ดังกล่าวสามารถดูได้ที่ภาคผนวก ก. หลังจากได้ค่าอัตราการเต้นของหัวใจมาแล้ว ค่าดังกล่าวสามารถนำมาพิจารณาได้ว่าอยู่ในช่วงของอัตราการเต้นของหัวใจในมนุษย์ปกติหรือไม่ เพื่อที่จะตัดสินใจได้ว่า ผู้ขับขี่สวมหมวกอยู่จริงไหม

2.2.3. การพิจารณาลักษณะการเคลื่อนไหวของศีรษะ

ในการตรวจสอบลักษณะการเคลื่อนไหวของศีรษะของผู้ขับขี่ ผู้พัฒนาได้แบ่งประเภทการเคลื่อนไหวออกเป็นสามแบบ ได้แก่ ความเร็วในการหันศีรษะ ความแรงในการเคลื่อนไหวศีรษะ และมุมของการก้มหรือเงยของศีรษะ

2.2.3.1. การวัดความแรงในการเคลื่อนไหวศีรษะ

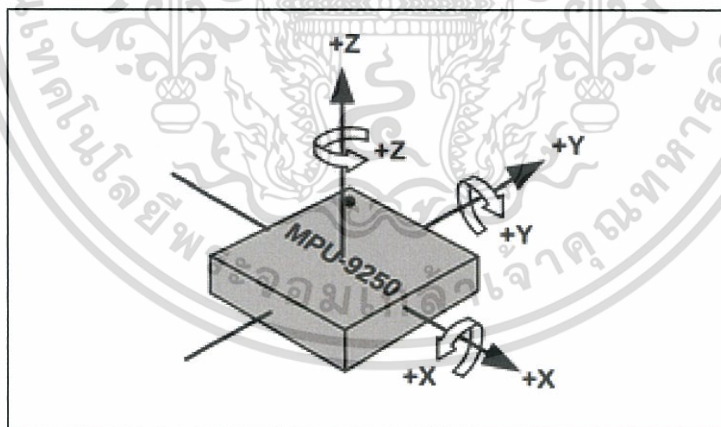
การวัดความแรงนั้นสามารถทำได้โดยใช้อุปกรณ์การวัดที่เรียกว่า Accelerometer โดยอุปกรณ์ดังกล่าวจะวัดค่าความแรงออกมาในหน่วยของค่าความเร่งเนื่องจากแรงโน้มถ่วงซึ่งสามารถแทนได้ด้วยอักษร g และจะแยกออกเป็นสามแกน ได้แก่ X, Y และ Z โดยที่เมื่อวาง Accelerometer ตามแนวระดับ แรงโน้มถ่วงของโลกจะอยู่ในแนวแกน Z พอดี ดังนั้นค่าความเร่งที่อ่านได้ที่แนวแกน Z จะอยู่ที่ประมาณ $1g$ รูป 2.5 แสดงให้เห็นแนวแกนทั้งสามเทียบกับแนวของแรงโน้มถ่วง



รูป 2.5 แนวแกนของ Accelerometer

2.2.3.2. การวัดความเร็วในการหันศีรษะ

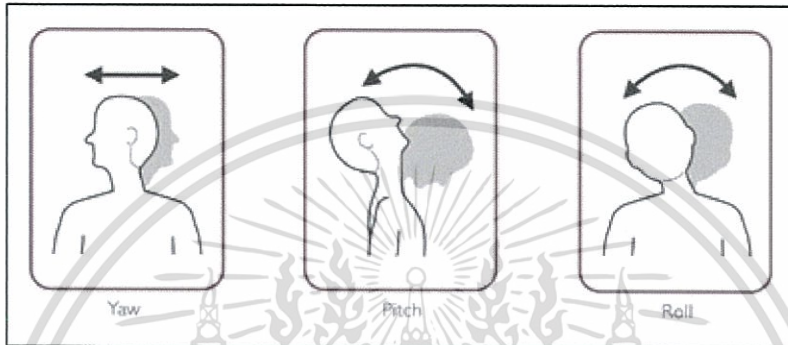
การวัดความเร็วในการหันศีรษะนั้นจะต่างกับการวัดความเร็วปกติที่วัดกับยานพาหนะตรงที่ การวัดความเร็วของยานพาหนะส่วนมากนั้นจะเป็นการวัดความเร็วเชิงเส้น มีหน่วยเป็นระยะทางต่อหน่วยเวลา แต่การวัดความเร็วของการหันศีรษะนั้นจะเป็นการวัดความเร็วเชิงมุม มีหน่วยเป็นมุมที่กวาดไปต่อหน่วยเวลา (Degree/Sec หรือ Rad/Sec) สามารถทำได้โดยใช้อุปกรณ์การวัดที่เรียกว่า Gyroscope โดยแยกออกเป็นสามแกนได้แก่ X, Y และ Z แสดงดังรูป 2.6



รูป 2.6 แนวแกนของ Gyroscope

2.2.3.3. การวัดมุมของการก้มหรือเงยของศีรษะ

ในการวัดมุมของการก้มหรือเงยของศีรษะจะใช้วิธีการวัดความเปลี่ยนแปลงของความเร่งไปตามแนวแกน X, Y และ Z โดยใช้ความรู้ที่ว่า หากแนวแกนใดก็ตามอยู่แนวเดียวกับแรงโน้มถ่วงของโลก จะมีค่าความเร่งอย่างน้อย $1g$ ดังนั้นเมื่อ Accelerometer เกิดการหมุน ค่าความเร่งที่ตกกับแกนต่าง ๆ จะค่อย ๆ เปลี่ยนไป โดยสามารถแบ่งแนวการหมุนได้เป็นสามแนว คือ Roll, Pitch และ Yaw แสดงดัง รูป 2.7



รูป 2.7 แนวแกนของ Gyroscope

แต่เนื่องจากการหมุนในแนวการหมุน Yaw ไม่ได้สร้างการเปลี่ยนแปลงของความเร่งต่อแกน X, Y หรือ Z ทำให้ Accelerometer ไม่สามารถวัดมุมในแนวแกนนี้ได้ โดยสมการในการคำนวณหามุมของแกน Pitch และ Roll นั้นคือ

$$Pitch = \tan^{-1} \left(\frac{G_Y}{\sqrt{G_X^2 + G_Z^2}} \right) \quad (2.4)$$

$$Roll = \tan^{-1} \left(\frac{-G_X}{G_Z} \right) \quad (2.5)$$

โดยที่ค่า G คือค่าความเร่งตามแนวแกน x มีหน่วยเป็นจำนวนเท่าของ g

2.2.4. การติดต่อสื่อสารระหว่างหมวกนิรภัยและรถจักรยานยนต์

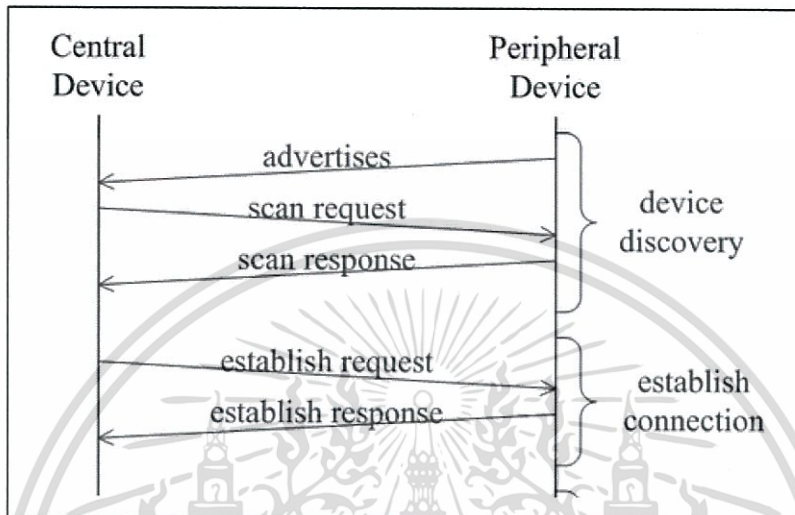
เพื่อลดระดับการใช้พลังงานและเพิ่มความยืดหยุ่นของทางเลือกอุปกรณ์ที่รองรับทางผู้พัฒนาได้ตัดสินใจใช้ Bluetooth Smart หรือ Bluetooth Low Energy หรือ BLE เป็นเทคโนโลยีในการติดต่อสื่อสารระหว่างหมวกนิรภัยและรถจักรยานยนต์ ในกรณีที่รถจักรยานยนต์ไม่ได้ถูกดัดแปลงให้ติดต่อกับหมวก ผู้ใช้จะสามารถเปลี่ยนไปใช้ Smart Phone ที่มีแทนได้ทันทีเนื่องจาก Smart Phone ส่วนใหญ่รองรับเทคโนโลยี Bluetooth Smart อยู่แล้ว ความสามารถของ Bluetooth Smart ที่ถูกนำมาใช้ในโครงการนี้ได้แก่ การยืนยันตัวตน การแลกเปลี่ยนข้อมูลโดยใช้ GATT และการยืนยันสิทธิ์ของปฏิบัติการ

2.2.4.1. การยืนยันตัวตน

ในการยืนยันตัวตนของ Bluetooth Smart หรือ Bluetooth Low Energy จะมีขั้นตอนที่ใช้ในการยืนยันตัวตนดังนี้

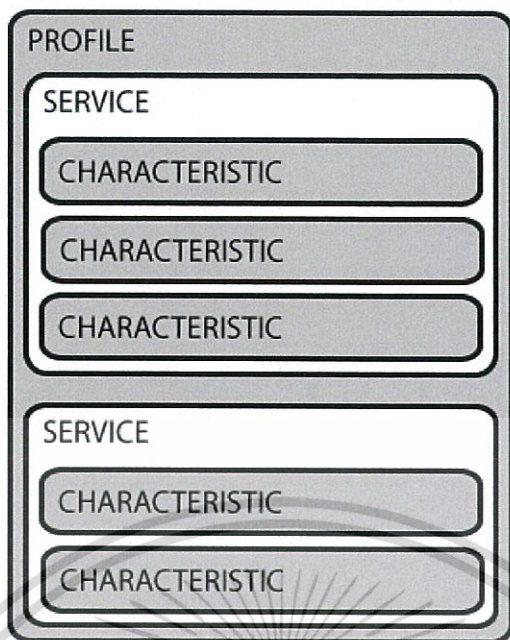
- 1) หมวกนิรภัยจะทำหน้าที่เป็นอุปกรณ์ Peripheral ที่จะส่ง Advertisement ออกไปเป็นระยะ ๆ เพื่อให้อุปกรณ์ Central สามารถรับรู้การมีอยู่ของหมวกนิรภัยได้
- 2) รถจักรยานยนต์จะทำหน้าที่เป็นอุปกรณ์ Central ที่คอยมองหา Advertisement ของอุปกรณ์ Peripheral อื่น ๆ
- 3) เมื่อรถจักรยานยนต์เจอ Advertisement ก็จะตรวจสอบว่าใช่ Advertisement ของหมวกนิรภัยหรือไม่ ซึ่งภายใน Advertisement ของหมวกนิรภัยจะมี Code ที่เป็นเหมือน ID แทรกอยู่ เมื่อรถจักรยานยนต์ตรวจเจอ Code ดังกล่าวใน Advertisement ก็จะส่งคำขอร้องเชื่อมต่อไปที่หมวกนิรภัย
- 4) หลังจากหมวกนิรภัยได้รับคำขอร้องเชื่อมตอก็จะทำการตรวจสอบคำขอร้องว่ามาจากรถจักรยานยนต์ใช่หรือไม่ ถ้าใช่ก็จะยอมให้สร้างการเชื่อมต่อ

โดยกระบวนการดังกล่าวแสดงได้ตาม รูป 2.8 โดย Scan Request และ Scan Response เป็นส่วนเสริมที่ไม่จำเป็นต้องมี จะใช้ในกรณีที่อุปกรณ์ Central ต้องการข้อมูลเพิ่มเติมจากอุปกรณ์ Peripheral ก่อนที่จะทำการส่งคำร้องขอเชื่อมต่อ ซึ่ง GATT ของ Bluetooth Smart จะมีวิธีการแลกเปลี่ยนข้อมูลดังนี้



รูป 2.8 การเปิด Connection ระหว่างอุปกรณ์ Central และ Peripheral

- 1) หมากรอกจะทำหน้าที่เป็น GATT Server ซึ่งจะเก็บข้อมูลไว้ในโครงสร้างข้อมูลแบบ Services และ Characteristics โดย Services จะทำหน้าที่เป็นเหมือนจุดให้บริการข้อมูล โดยที่ตัวข้อมูลจริงๆ จะถูกเก็บไว้ใน Characteristics ตัวอย่างเช่น ในหมากรอกจะมี Service ที่เรียกว่า Head Motion Service โดยที่ข้างในจะมี 3 Characteristics ได้แก่ Acceleration Characteristic, Angular Velocity Characteristic และ Head Angle Characteristic เป็นต้น
- 2) รถจักรยานยนต์จะทำหน้าที่เป็น GATT Client คือเมื่อไรที่ต้องการข้อมูลบางอย่างก็จะส่งคำร้องขอข้อมูลเข้าไปที่ GATT Server ซึ่งในที่นี้คือหมากรอก โดย GATT Client จะต้องระบุ Service และ Characteristic ที่ต้องการโดย logical ของ view ของ service และ Characteristic ภายใต้อินเตอร์เฟซ Bluetooth Smart Profile จะแสดงในรูป 2.9



รูป 2.9 Bluetooth Low Energy Stack

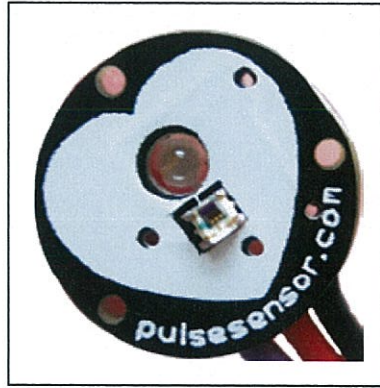
2.2.4.2. การยืนยันสิทธิ์ของปฏิบัติการ

การยืนยันสิทธิ์จะเกิดขึ้นที่ GATT Server โดยเมื่อมีคำร้องขอข้อมูลหรือคำร้องขอเขียนข้อมูลไปที่ Characteristic ใดๆ ก็ตาม GATT Server จะมีสิทธิ์ในการยอมรับหรือปฏิเสธคำขอเหล่านั้นได้ โดยขึ้นอยู่กับสิทธิ์ของตัว GATT Client นั้นเอง

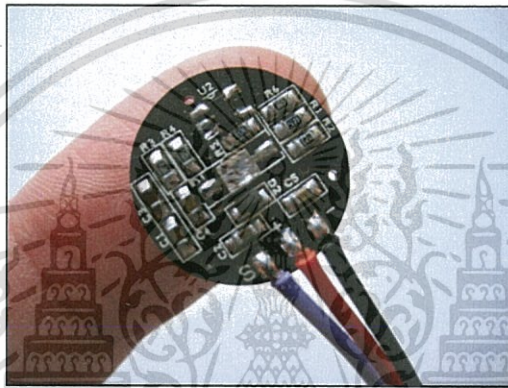
2.3. เทคโนโลยีที่เกี่ยวข้อง

2.3.1. เครื่องวัดอัตราการเต้นของหัวใจชีพจรโดยใช้แสง (Optical Heart Rate Measurement Sensor)

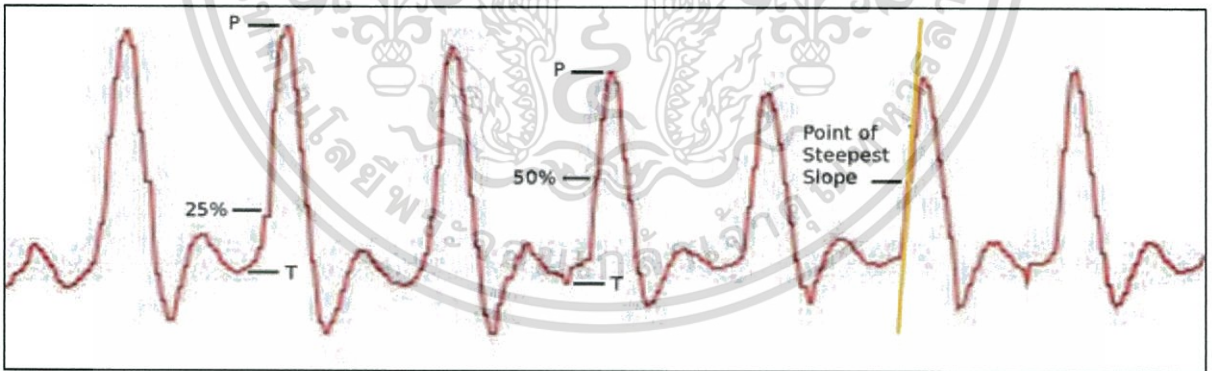
เป็น Heart Rate Measurement Sensor ที่เป็นแบบ Photoplethysmograph ซึ่งใช้หลักการวัดความหนาแน่นที่เปลี่ยนแปลงไปของแสงที่สะท้อนกลับมาจากอวัยวะที่เราทำการวัด (ในที่นี้คือนิ้ว ติ่งหู หรือขมับ) โดยที่ตัว Sensor จะมีส่วนที่ใช้สำหรับฉายไฟเข้าไปในอวัยวะและส่วนที่รับค่าความหนาแน่นแสงที่สะท้อนกลับมา รูป 2.10 แสดงภาพตัวอย่างของ Pulse Sensor อุปกรณ์วัดชีพจรที่ทีมผู้พัฒนาใช้ รูป 2.11 แสดงตัวอย่างวิธีการใช้งานของ Pulse Sensor และรูป 2.12 แสดงรูปคลื่นตัวอย่างของสัญญาณชีพจรที่อ่านได้จาก Sensor



รูป 2.10 อุปกรณ์วัดชีพจร Pulse Sensor



รูป 2.11 ตัวอย่างการใช้งานของ Pulse Sensor



รูป 2.12 ตัวอย่างสัญญาณชีพจรที่อ่านจาก Sensor

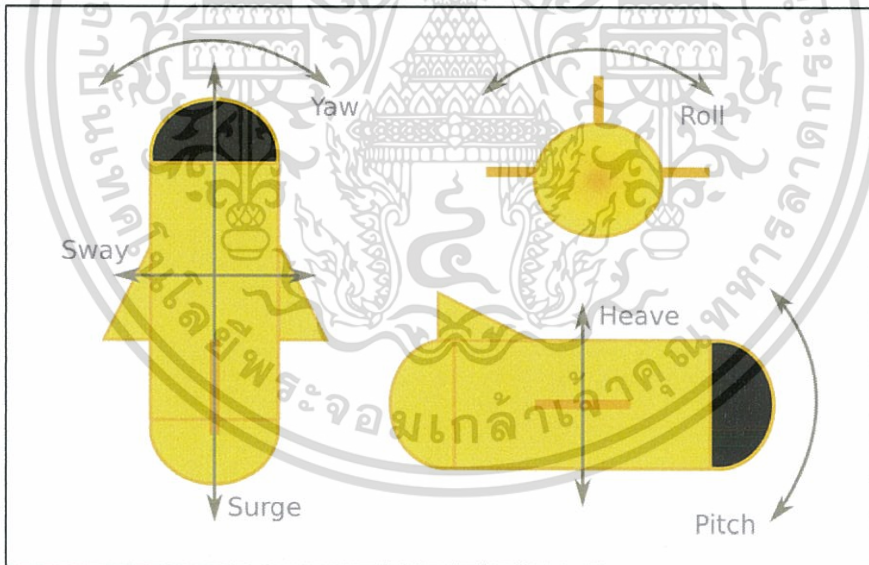
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2. หน่วยวัดความเฉื่อย (Inertial Measurement Unit)

ในการตรวจจับความผิดปกติ หรือ Anomaly ระหว่างการขับเคลื่อนของอากาศยานนั้น เราได้เลือกใช้วิธีตัดสินใจจากการเคลื่อนไหวที่ผิดปกติของศีรษะผู้ขับขี่ โดยอุปกรณ์ที่เฝ้าดูการเคลื่อนไหวของศีรษะก็คือหน่วยวัดความเฉื่อย หรือ Initial Measurement Unit ทั้งนี้ตัว Initial Measurement Unit เองถูกติดตั้งในอากาศยานไว้คนขับส่วนใหญ่เพื่อใช้ในการเฝ้าดูการเคลื่อนไหวของอากาศยานนั้น ๆ โดยภายใน Initial Measurement Unit จะประกอบด้วย Sensor อย่างน้อยสองอย่าง ได้แก่

- 1) Accelerometer ใช้สำหรับการวัดความเร่งที่เกิดขึ้นในแนวระนาบ X-Y และแนวดิ่ง Z โดยจะวัดเทียบกับค่าความเร่งเนื่องจากแรงโน้มถ่วงของโลกที่ระดับน้ำทะเล หรือค่า G เช่น 1g 2g 3.5g หรือ 1.02g เป็นต้น
- 2) Gyroscope ใช้สำหรับวัดความเร็วเชิงมุมที่เกิดขึ้นในแนวระนาบ X-Y และแนวดิ่ง Z โดยจะวัดเป็นหน่วย Degree/Second หรือ Radian/Second

ทั้งนี้หนึ่งในสิ่งผิดปกติที่เกิดขึ้นระหว่างการขับเคลื่อนก็คือ การหันศีรษะในองศาที่ผิดปกติ โดยการหาองศาของศีรษะนั้น เราจะใช้แกนอ้างอิงแบบเดียวกันกับที่อากาศยานทั่ว ๆ ไปใช้ ได้แก่ Roll Pitch และ Yaw แสดงในรูป 2.13



รูป 2.13 แกนอ้างอิง

การคำนวณหาองศาในแนวแกน Pitch และ Roll นั้น ทำได้โดยอาศัย Accelerometer ซึ่งสามารถวัดความเร่งเนื่องจากแรงโน้มถ่วงของโลกตามแนวแกน X, Y และ Z ได้ โดยใช้สูตรการคำนวณจากสมการที่ 2.4 และ 2.5 ในการคำนวณ

2.3.3. Bluetooth Smart หรือ Bluetooth Low Energy

เนื่องจากบนหมวกนิรภัยนั้น การใช้สายจ่ายพลังงานให้ตลอดเวลาเป็นการสร้างความลำบากให้กับผู้ขับขี่ เราจึงเลือกที่จะหาวิธีลดการบริโภคพลังงานของหมวกลงให้มากที่สุดเท่าที่เป็นไปได้ ผลที่ได้คือการคำนวณบางส่วนจะถูกกระทำที่ Server และบนตัวรถที่มีพลังงานใช้งานตลอดเวลา และ Wireless Communication Protocol ที่ใช้จะต้องประหยัดพลังงานมากที่สุดเท่าที่เป็นไปได้ในขณะเดียวกันก็ต้องเป็น Protocol ที่มีการใช้งานในหลายอุปกรณ์และง่ายต่อการพัฒนา ด้วยเหตุผลที่กล่าวมา ทำให้ Bluetooth Smart หรือ Bluetooth Low Energy กลายเป็นตัวเลือกที่ดีเลือกใช้งาน



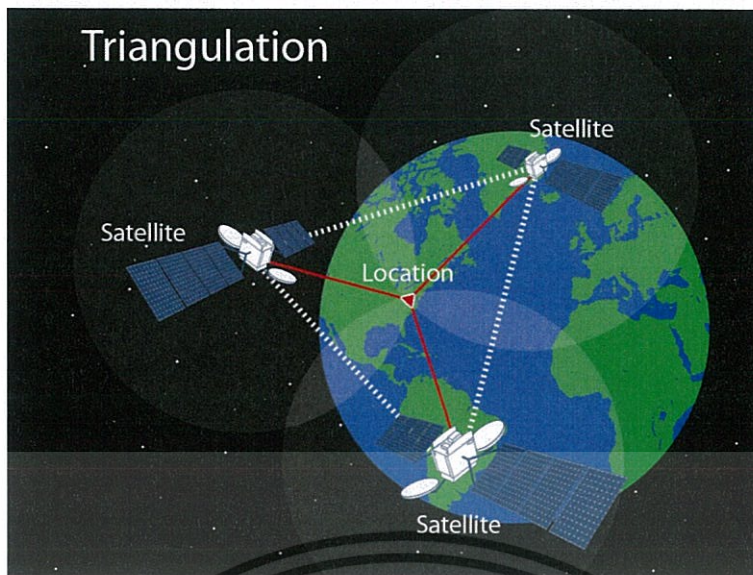
รูป 2.15 Bluetooth Smart หรือ Bluetooth Low Energy (BLE)

2.3.4. GNSS หรือ Global Navigation Satellite System

GNSS คือ ระบบที่ระบุตำแหน่งบนพื้นโลกจากเวลาและพิกัดที่ได้จากการเข้ารหัสสัญญาณดาวเทียมที่โคจรอยู่บนอวกาศ ทั้ง GPS และ GLONASS ต่างก็เป็นตัวอย่างของ GNSS และระบบในปัจจุบันสามารถคำนวณความเร็วและทิศทางซึ่งนำมาใช้ร่วมกับโปรแกรมแผนที่ เพื่อใช้ในการนำทางได้ โดย GNSS แบ่งเป็น 2 อย่างหลัก ๆ คือ

- 1) GPS (Global Positioning System) คือ ระบบนำทางผ่านดาวเทียมทำงานดำเนินการโดยประเทศสหรัฐอเมริกา
- 2) GLONASS (Global Navigation Satellite System) คือ ระบบนำทางผ่านดาวเทียมทำงานดำเนินการโดยประเทศรัสเซีย

ซึ่งแม้ทั้งสองระบบจะสามารถใช้งานได้ทั่วโลก แต่ในสภาพแวดล้อมภายในอาคารนั้นระบบทั้งสองไม่สามารถให้ข้อมูลที่ถูกต้องแล้วแม่นยำได้และการสะท้อนของสัญญาณกับต้นไม้หรืออาคารก็มีโอกาสทำให้การทำงานของระบบถูกหยุดชะงักได้เช่นกัน



รูป 2.16 ภาพอธิบายการทำงานของ GNSS

2.3.5. Android Operating System

เป็นระบบปฏิบัติการที่ถูกพัฒนาโดย Google มีพื้นฐานมาจาก Linux Kernel ตัวระบบปฏิบัติการนี้ออกแบบมาเพื่อรองรับ Touch Screen โดยมักถูกใช้บน Smart Phone และ Tablet นอกจากนี้ทั้งตัว Smart Phone และ Tablet ต่างก็มี Sensor หลากหลายตัว อย่างเช่น Magnetometer, Accelerometer, Gyroscope, GPS, Mobile Communication, Bluetooth และ Camera ทั้งยังเป็นอุปกรณ์ที่ผู้ใช้มีผู้ใช้งานจำนวนมาก โดยในรูป 2.7 เป็นตัวอย่างอุปกรณ์ที่ใช้ระบบปฏิบัติการ Android



รูป 2.17 ภาพ Smart Phone ที่ใช้ Android เป็นระบบปฏิบัติการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การวิเคราะห์และออกแบบระบบ

3.1. ภาพรวมการทำงานภายในระบบ

การไหลของข้อมูลภายในระบบจะเริ่มจากส่วนของหมวกนิรภัย เริ่มแรกผู้ขับขี่จะสวมหมวกนิรภัย และทำให้เกิด Event การใส่หมวกเกิดขึ้นภายในหมวกนิรภัย จากนั้นเหตุการณ์ดังกล่าวจะถูกส่งไปที่ตัวรถจักรยานยนต์และทำให้วงจรการติดเครื่องยนต์ครบวงจร ผู้ขับขี่จะสามารถติดเครื่องรถได้ปกติ ในขณะเดียวกัน หากไม่มีเหตุการณ์การสวมหมวกเกิดขึ้น วงจรการติดเครื่องยนต์บนตัวรถจักรยานยนต์ก็จะยังคงไม่ครบวงจรและผู้ขับขี่ก็จะไม่สามารถติดเครื่องยนต์ได้ ซึ่งหลังจากที่ส่วนควบคุมที่อยู่บนรถจักรยานยนต์ได้รับ เหตุการณ์มาจากหมวกนิรภัยแล้ว ตัวรถจักรยานยนต์ก็จะทำการส่งข้อมูลดังกล่าวต่อไปให้กับ Server ซึ่งจะทำการเก็บข้อมูลเหตุการณ์ประจำวันแต่ละคันไว้และนำมาแสดงผลในกรณีที่ผู้ใช้ต้องการดู

หลังจากที่ผู้ขับขี่สามารถติดเครื่องยนต์และได้ทำการขับขี่ตามปกติแล้ว ในระหว่างนี้หมวกนิรภัยก็จะทำการส่งค่าที่อ่านได้จาก Sensor บนหมวกไปที่รถจักรยานยนต์เป็นระยะ ๆ เพื่อเป็นการบอกสถานะการเคลื่อนไหวศีรษะของผู้ขับขี่ และนอกจากค่าที่อ่านได้จาก Sensor แล้ว บนหมวกนิรภัยจะมีการคำนวณ Event ความผิดปกติต่าง ๆ ที่เกิดขึ้นในระหว่างการขับขี่อีกด้วย Event ดังกล่าวได้แก่

- 1) การหันหัวในองศาที่ผิดปกติไปจากองศาที่ควรจะมีได้ในระหว่างขับขี่รถจักรยานยนต์ เช่น การก้มในลักษณะที่ต่ำเกินไป
- 2) ความเร่งที่เกิดขึ้นมีค่าเกินกว่าค่ากลางที่ได้กำหนดไว้ ซึ่งเหตุการณ์ดังกล่าวอาจจะเกิดได้จากการเบรกรกระทันหันหรืออุบัติเหตุต่าง ๆ ก็ได้
- 3) ค่าความเร็วเชิงมุมมีค่าเกินกว่าค่ากลางที่ได้กำหนดไว้ ซึ่งเหตุการณ์นี้อาจจะเกิดจากการที่ผู้ขับขี่ทำการหันศีรษะเร็วมาก ๆ หรืออุบัติเหตุต่าง ๆ ก็ได้
- 4) หมวกนิรภัยไม่สามารถตรวจจับได้ว่า มีผู้ขับขี่กำลังสวมตัวหมวกอยู่ เหตุการณ์นี้อาจจะเกิดจากการงใจถอดหมวกของผู้ขับขี่ หรืออาจเกิดจากอุบัติเหตุที่ทำให้หมวกนิรภัยหลุดจำศีรษะของผู้ขับขี่

ทั้งนี้ไม่ว่าจะเป็นข้อมูลที่ได้อ่านจาก Sensor หรือได้จากการคำนวณเหตุการณ์ ข้อมูลทั้งหมดก็ จะถูกส่งไปยังรถจักรยานยนต์แล้วจากนั้นจักรยานยนต์ก็จะส่งต่อไปให้กับ Server เพื่อจัดเก็บไว้สำหรับการวิเคราะห์ข้อมูลเหล่านี้ในอนาคต หรือแสดงผลในกรณีที่ผู้ใช้ต้องการดู

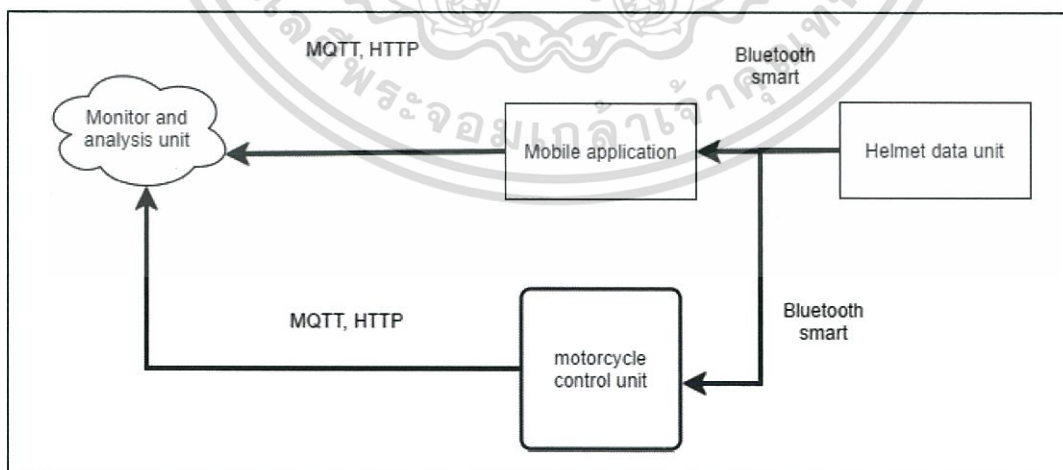
นอกเหนือจากข้อมูลที่ได้มาจากหมวกนิรภัยแล้วนั้น ตัวรถจักรยานยนต์เองก็จะมีการสร้างข้อมูล ส่วนของตัวเองด้วย ข้อมูลที่เพิ่มเข้ามาในส่วนนี้ได้แก่

- 1) ความเร็วปัจจุบันของรถจักรยานยนต์
- 2) ตำแหน่งทางภูมิศาสตร์ปัจจุบันของรถจักรยานยนต์
- 3) ทิศทางเชิงภูมิศาสตร์ที่รถจักรยานยนต์กำลังมุ่งหน้าไป
- 4) เวลาปัจจุบัน

ข้อมูลเหล่านี้เองก็จะถูกส่งขึ้นไปบน Server ด้วยเหตุผลเช่นเดียวกันกับข้อมูลชุดก่อน ๆ หน้า เช่นกัน โดยที่ฝั่งของ Server ข้อมูลต่าง ๆ จะถูกเก็บไว้เพื่อนำไปวิเคราะห์เพื่อผู้ใช้ที่ต้องการทราบ สถานะปัจจุบันของผู้ขับขี่เช่นว่าในตอนนี้อยู่ที่ตำแหน่งใดกำลังขับขี่รถจักรยานยนต์ด้วยความเร็วเท่าไรหรือเกิด เหตุการณ์ผิดปกติอะไรขึ้นกับผู้ขับขี่หรือไม่ เป็นต้น นอกจากนี้ทางฝั่ง Server เอง ยังมีการคำนวณหา ค่า ความเสี่ยงของเส้นทางที่ผู้ขับขี่กำลังใช้อยู่ด้วย ตัวอย่างเช่น บนถนนบางสาย มีสถิติการเกิดอุบัติเหตุ มากกว่าถนนสายอื่น ๆ หรืออาจจะเป็นตัวผู้ขับขี่เองที่มีพฤติกรรมขับขี่ที่เสี่ยงอันตรายและนำมา แสดงผล

3.2. ส่วนประกอบภายในระบบ

ระบบถูกแบ่งออกเป็นระบบย่อยใหญ่ ๆ สามระบบ ได้แก่หน่วยรับข้อมูลหมวกนิรภัย หรือ Helmet Data Unit หรือ หน่วยควบคุมรถจักรยานยนต์ Motorcycle Control Unit หน่วยแสดงผล Mobile Application และ ส่วนวิเคราะห์ข้อมูล Monitor And Analysis Unit ซึ่งทั้ง 4 ส่วนจะทำการติดต่อ และส่งข้อมูลไปแต่ละส่วนดังแผนภาพดังนี้ตามรูป 3.1



รูป 3.1 ส่วน ประกอบหลักของระบบ

3.2.1. หน่วยรับข้อมูลหมวกนิรภัย หรือ Helmet Data Unit

ทำหน้าที่ในการเก็บข้อมูลจาก Sensor ต่าง ๆ บนหมวกนิรภัยและคำนวณหาเหตุการณ์ความผิดปกติต่าง ๆ ที่เกิดขึ้นในระหว่างการขับขี่จักรยานยนต์ จากนั้นจึงส่งข้อมูลเหล่านี้ไปที่ Motorcycle Control Unit โดยผ่าน Bluetooth Smart หรือ Bluetooth Low Energy เป็น Network Protocol และผ่านที่ Mobile Application โดยใช้ Bluetooth Smart หรือ Bluetooth Low Energy เป็น Network Protocol เช่นเดียวกัน

3.2.2. หน่วยควบคุมรถจักรยานยนต์ หรือ Motorcycle Control Unit

ทำหน้าที่รับข้อมูลจาก Helmet Data Unit และตรวจสอบว่า มีการใส่หมวกเกิดขึ้นหรือไม่ ถ้ามีให้ปิดวงจรการติดเครื่องยนต์ ถ้าไม่มีให้เปิดไว้อย่างเดิม จากนั้นจึงสร้าง Packet ข้อมูลใหม่โดยเพิ่มส่วนของ

- 1) ความเร็วปัจจุบันของรถจักรยานยนต์
- 2) ตำแหน่งทางภูมิศาสตร์ปัจจุบันของรถจักรยานยนต์
- 3) ทิศทางเชิงภูมิศาสตร์ที่รถจักรยานยนต์กำลังมุ่งหน้าไป และ
- 4) เวลาปัจจุบัน

เข้าไปแล้วจึงส่ง Packet ข้อมูลใหม่ไปที่ Monitor And Analysis Unit โดยผ่าน MQTT หรือ HTTP เป็น Network Protocol

3.2.3. หน่วยแสดงผลและวิเคราะห์ข้อมูล หรือ Monitor And Analysis Unit

ทำหน้าที่รับข้อมูลจาก MCU มาเก็บและแสดงผลให้กับผู้ใช้เพื่อแสดงสถานะการขับขี่ปัจจุบันของผู้ขับขี่ นอกจากนี้ยังทำการคำนวณและวิเคราะห์ข้อมูลที่ได้รับมาเพื่อหาค่าความเสี่ยงของเส้นทางที่ผู้ขับขี่กำลังใช้งาน

3.3. อุปกรณ์และเทคโนโลยีที่นำมาใช้

3.3.1. หน่วยรับข้อมูลหมวกนิรภัย หรือ Helmet Data Unit

จากการศึกษาข้อมูลและเทคโนโลยีแล้วได้ทำเลือกใช้ ของอุปกรณ์ในส่วนที่นำมาใช้งานในหน่วยรับข้อมูลหมวกนิรภัย หรือ Helmet Data Unit ได้เลือกใช้อุปกรณ์ดังนี้

3.3.1.1. หน่วยประมวลผลหลัก NRF51822 Arm Cortex-M0

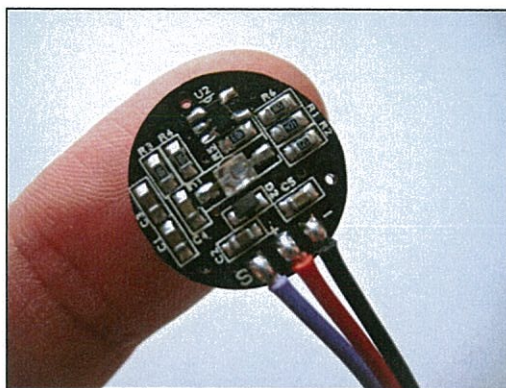
ด้วยเหตุผลทางด้านพลังงานตามที่ได้กล่าวไปแล้ว นั้น ทำให้เราได้เลือกใช้สถาปัตยกรรมของหน่วยประมวลผลเป็น Arm Cortex-M0 ซึ่งเป็นสถาปัตยกรรมของ Arm ที่ประหยัดพลังงานที่สุดในขณะนี้ หลังจากที่ได้ทำการค้นหาผู้ผลิตชิพโมเดลนี้ เราได้ตกลงที่จะใช้ NRF51822 ซึ่งใช้สถาปัตยกรรมของ Arm ตัวที่กล่าวไป นอกจากนี้ในตัวชิพเองยังได้บรรจุ Bluetooth Low Energy Module ไว้ภายในอีกด้วย โดยชิพตัวดังกล่าวนี้ถูกผลิตโดยบริษัท Nordic Semiconductor ซึ่งถือเป็นหนึ่งในบริษัทที่มีชื่อเสียงทางด้าน Wireless Communication มากแห่งหนึ่ง



รูป 3.2 Nrf51822 ชิพ Arm Cortex-M0 With Ble Module

3.3.1.2. เครื่องวัดอัตราการเต้นของหัวใจโดยใช้แสง (Optical Heart Rate Measurement)

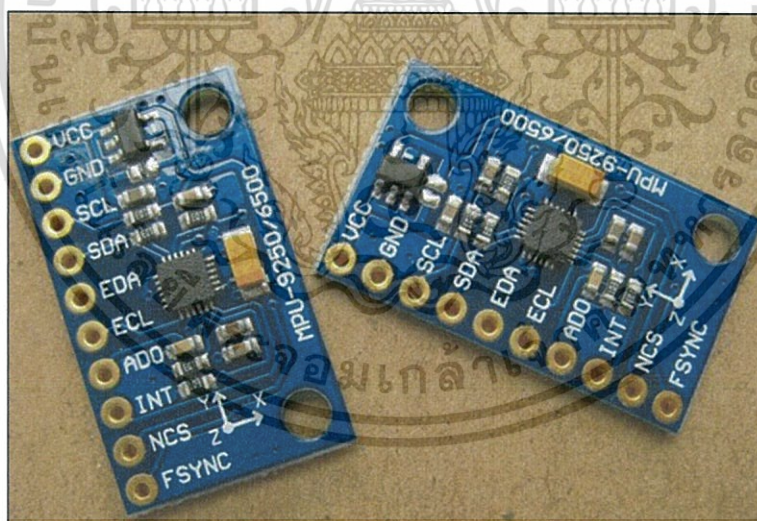
ในการตรวจสอบว่าหมวกกำลังถูกสวมอยู่หรือไม่ เราได้ใช้วิธีวัดชีพจรของวัตถุที่อยู่ในหมวก ด้วยวิธีนี้จะทำให้เราสามารถยืนยันได้อย่างน้อยว่า สิ่งที่อยู่ในหมวกเป็นสิ่งมีชีวิต โดยเราได้ใช้ Heart Rate Measurement Sensor ที่เป็นแบบ Photoplethysmograph ซึ่งใช้หลักการวัดความหนาแน่นที่เปลี่ยนแปลงของแสงที่สะท้อนกลับมาจากอวัยวะที่เราทำการวัด (ในที่นี้คือนิ้ว ติ่งหู หรือ ขมับ) โดยที่ตัว Sensor จะมีส่วนที่ใช้สำหรับฉายไฟเข้าไปในอวัยวะและส่วนที่รับค่าความหนาแน่นแสงที่สะท้อนกลับมา รูป 3.3 แสดงภาพตัวอย่างของ Pulse Sensor อุปกรณ์วัดชีพจรที่ทีมผู้พัฒนา



รูป 3.3 ตัวอย่างการใช้งานของ PulseSensor

3.3.1.3. เซ็นเซอร์วัดความเฉื่อย (Inertial Measurement Unit)

ทั้งนี้หนึ่งในสิ่งผิดปกติที่เกิดขึ้นระหว่างการขับขี่ก็คือ การหันศีรษะในองศาที่ผิดปกติ โดยการหาองศาของศีรษะนั้นได้โดยอาศัย Accelerometer ซึ่งสามารถวัดความเร่งเนื่องจากแรงโน้มถ่วงของโลกตามแนวแกน X, Y และ Z ได้ โดยในโครงการนี้ ตัว IMU ที่เราได้เลือกใช้คือ MPU9250 ซึ่งถือเป็นชิพประมวลผลทางด้าน Motion Processing ที่มีชื่อเสียงตัวหนึ่ง โดยตัว Kernel ของ Android เองก็รองรับชิพรุ่นนี้เช่นกัน



รูป 3.4 บอร์ด IMU ที่ใช้ชิพ MPU9250

3.3.1.4. หน่วยรับข้อมูลหมวกนิรภัย หรือ Helmet Data Unit

จากความต้องการของระบบที่ได้วิเคราะห์มา สามารถแบ่งเป็นส่วนที่อยู่บนหมวกนิรภัย ได้แก่ ส่วนของการพิจารณาการเคลื่อนไหวของศีรษะ ส่วนของการตรวจสอบว่าหมวกนิรภัยถูกสวมอยู่หรือไม่ และส่วนที่ใช้ในการติดต่อสื่อสารระหว่างหมวกนิรภัยและรถจักรยานยนต์

3.3.2. หน่วยควบคุมรถจักรยานยนต์ หรือ Motorcycle Control Unit

ในส่วนของอุปกรณ์ที่นำมาใช้ในการในหน่วยควบคุมรถจักรยานยนต์ Motorcycle Control Unit ได้เลือกใช้อุปกรณ์ดังนี้

3.3.2.1. หน่วยประมวลผลหลัก (Linkit Smart 7688 DUO)

เป็นส่วนที่ทำหน้าที่นำข้อมูลสถานะต่าง ๆ ที่ได้จากหมวกกันน็อกและรับค่าตำแหน่งในปัจจุบันจาก GNSS และส่งขึ้นไปยัง Sever นอกจากนี้จะทำหน้าที่เป็นตัวกลางการติดต่อระหว่างหมวกนิรภัยและตัว Sever แล้ว ยังทำหน้าที่จัดการการสตาร์ทเครื่องยนต์ของยานพาหนะด้วยหากผู้ใช้ไม่สวมหมวกกันน็อกก่อนจะไม่สามารถสตาร์ทรถได้ โดยในระบบนี้เลือกใช้ Linkit Smart 7688 DUO ซึ่งเป็นอุปกรณ์ที่มีส่วนประกอบ 2 ส่วนคือ

- 1) MT7688AN Micro Processor มี Core เป็น MIPS24KEc ที่พัฒนาโดย MIPS Technologies และ OpenWrt เป็น Operating System คอยจัดการงานต่าง ๆ ให้ ซึ่งถูกนำไปใช้ในระบบ Embedded ตัวอย่าง เช่น Routers, Residential Gateways, And Video Game Consoles
- 2) Atmega32u4 Micro Controller มี Core เป็น Atmel AVR ที่ถูกพัฒนาโดย Atmel ซึ่งเป็น Chipset ซึ่งถูกนำไปใช้เป็น Micro Controller หลายรุ่น



รูป 3.5 ภาพหน่วยประมวลผล Linkit Smart 7688 DUO

3.3.2.2. ไมโครโปรเซสเซอร์ในการติดต่อ (NRF51822 Arm Cortex-M0)

เราได้เลือกใช้สถาปัตยกรรมของหน่วยประมวลผลเป็น Arm Cortex-M0 ซึ่งเป็นสถาปัตยกรรมของ Arm และเป็นชิปเดียวกันกับที่ใช้ในหน่วยรับข้อมูลหมวกนิรภัย (Helmet Data Unit)

3.4. การออกแบบการทำงานและสถาปัตยกรรมของระบบ

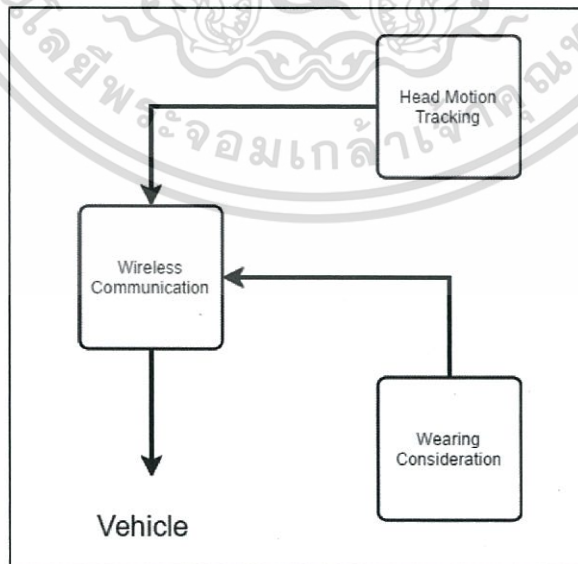
3.4.1. หน่วยรับข้อมูลหมวกนิรภัย (Helmet Data Unit)

จากความต้องการของระบบที่ได้วิเคราะห์มาแล้ว สามารถหน่วยรับข้อมูลหมวกนิรภัย ได้แก่ ส่วนของการพิจารณาการเคลื่อนไหวของศีรษะ ส่วนของการตรวจสอบว่าหมวกนิรภัยถูกสวมอยู่หรือไม่ และส่วนที่ใช้ในการติดต่อสื่อสารระหว่างหมวกนิรภัยและรถจักรยานยนต์

3.4.1.1. การวางส่วนประกอบฮาร์ดแวร์ภายในหมวกนิรภัย

โดยแต่ละส่วนของส่วนประกอบภายในหน่วยรับข้อมูลหมวกนิรภัย หรือ Helmet Data Unit จะมีหน้าที่ดังนี้

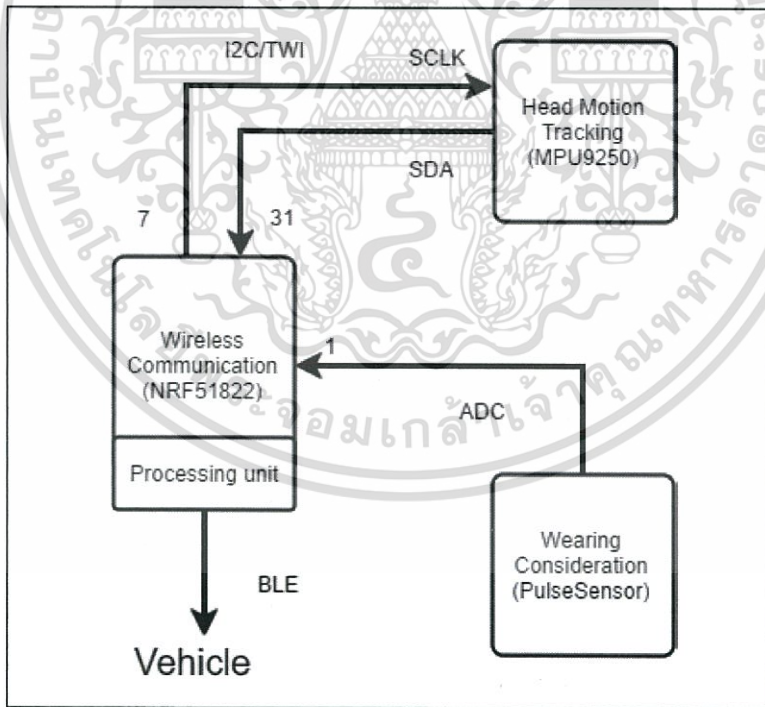
- 1) Head Motion Tracking: ตรวจสอบสภาพการเคลื่อนไหวของศีรษะ ได้แก่ การกระชากที่เกิดขึ้นที่หมวกนิรภัย การหมุนของศีรษะ และขนาดของมุมก้มหรือเงย
- 2) Wearing Consideration: ตรวจสอบว่าในขณะที่หมวกได้ถูกสวมใส่อย่างถูกต้องหรือไม่
- 3) Wireless Communication: รับผิชอบการติดต่อสื่อสารระหว่างหมวกนิรภัยและรถจักรยานยนต์



รูป 3.6 ส่วนประกอบฮาร์ดแวร์ภายในหมวกนิรภัย

ซึ่งในการพัฒนาจริงผู้พัฒนาได้วางฮาร์ดแวร์แต่ละส่วนและทำการเชื่อมต่อแต่ละส่วนเข้าด้วยกันดังนี้

- 1) Head Motion Tracking ใช้ Accelerometer และ Gyroscope โดยจะใช้เซนเซอร์ที่เรียกว่า Inertial Measurement Unit หรือ IMU โดย IMU จะเป็นหน่วยการวัดความเฉื่อยที่ประกอบได้ Accelerometer และ Gyroscope ซึ่ง IMU ที่ใช้ในโครงการนี้คือ MPU9250 ภายในประกอบด้วย Accelerometer, Gyroscope และ Magnetometer อินเตอร์เฟสที่ใช้ติดต่อกับส่วนประมวลผลคือ I2C (Inter-Integrated Circuit)
- 2) Wearing Consideration ใช้ Photoplethysmograph หรือเครื่องวัดปริมาตรของอวัยวะโดยใช้แสงซึ่ง Photoplethysmograph จะทำหน้าที่วัดอัตราการเต้นหัวใจของผู้ขับขี่ ตัวที่ใช้ในโครงการนี้มีชื่อว่า Pulse Sensor ทำงานร่วมกับลิ้มิตสวิทช์เพื่อเพิ่มความแม่นยำในการตรวจสอบ
- 3) Wireless Communication ใช้ NRF51822 ซึ่งเป็นทั้งโมดูล Bluetooth Smart และหน่วยประมวลผลสถาปัตยกรรม ARM ภายใน IC ตัวเดียวกัน



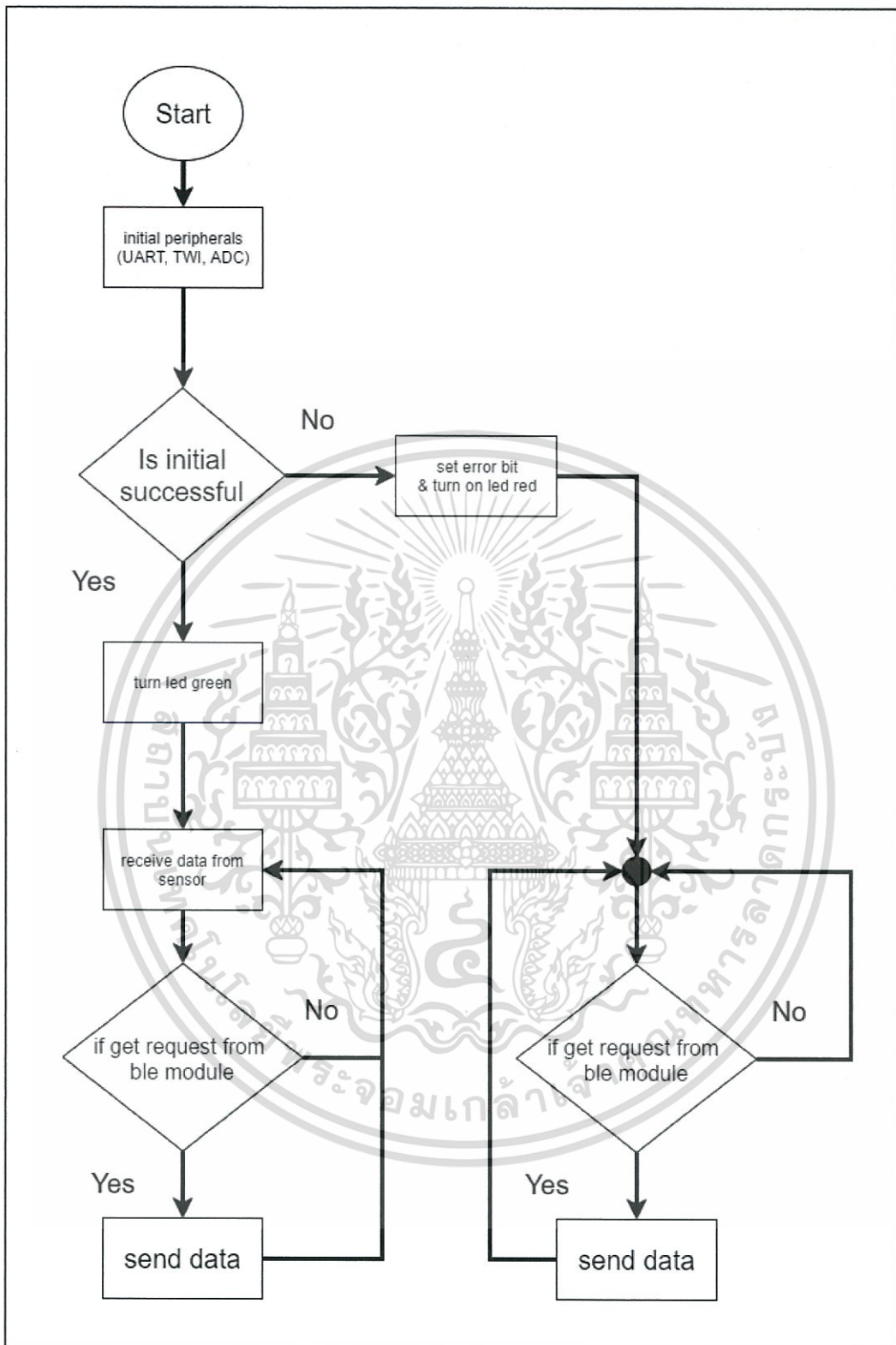
รูป 3.7 ส่วนประกอบฮาร์ดแวร์ภายในหมวกนิรภัยและการเชื่อมต่อระหว่างฮาร์ดแวร์แต่ละส่วน

3.4.1.2. การออกแบบลำดับการทำงาน

หลังจากเปิดการทำงานของฮาร์ดแวร์ต่าง ๆ ภายในหมวกนิรภัยจะเกิดการดำเนินงานภายในฮาร์ดแวร์ต่าง ๆ ดังนี้

- 1) NRF51822 ตรวจสอบสภาพการทำงานของฮาร์ดแวร์ที่อยู่ในหมวกนิรภัยทั้งหมด หากมีฮาร์ดแวร์ตัวไหนรวมถึงตัวมันเองอยู่ในสภาพที่ไม่พร้อมทำงาน NRF51822 จะแสดงออกมาทาง LED และเก็บรายละเอียดไว้ใน Characteristic และ Service ที่กำหนดไว้
- 2) NRF51822 เก็บสถานะของการเคลื่อนไหวศีรษะจาก MPU9250 เมื่อผู้ใช้สวมหมวก NRF51822 จะได้ข้อมูลสถานะการสวมหมวกดังกล่าวมาจาก Pulse Sensor และลิ้มิตสวิตช์
- 3) ข้อมูลต่าง ๆ ตั้งแต่ข้อ 1-3 จะถูกส่งให้กับรถจักรยานยนต์ผ่านทางโมดูล Bluetooth Smart โดยรถจักรยานยนต์จะเป็นส่วนที่ส่งคำร้องขอข้อมูลต่าง ๆ





รูป 3.8 การติดต่อส่วนประกอบฮาร์ดแวร์ภายในหมวกนิรภัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.1.3. การออกแบบ Services และ Characteristics ภายใน Bluetooth Smart Profile

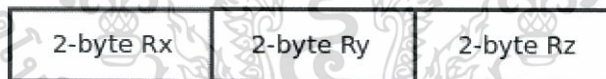
เนื่องจากหน้าที่ของหมวกนิรภัยมีอยู่สามอย่างแบ่งตามส่วนประกอบซึ่งโดยรวมแล้วคือการเก็บสถานะของผู้ขับขี่และส่งให้กับรถจักรยานยนต์โดย Helmet Information Services จะให้ข้อมูลต่าง ๆ ที่จำเป็นแก่รถจักรยานยนต์ โดยเป็น Service หลักและเป็นเพียง Service เดียวใน Profile ของหมวกนิรภัย เนื่องจากหน้าที่ของหมวกนิรภัยไม่มีอย่างอื่นนอกจากการนำส่งข้อมูลแก่รถจักรยานยนต์ โดยจะสามารถออกแบบ Services และ Characteristics ได้ดังนี้

- 1) Error and Event Characteristic เก็บข้อมูลที่บ่งบอกสถานะของหมวกนิรภัย เช่น การทำงานที่ผิดพลาดของฮาร์ดแวร์ และบอกถึงเหตุการณ์ที่เกิดขึ้น เช่น เกิดการสวมหมวกหรือการเคลื่อนไหวศีรษะในลักษณะไม่ปกติ โดยข้อมูลจะเก็บอยู่ในรูปแบบแสดงดังรูป 3.9



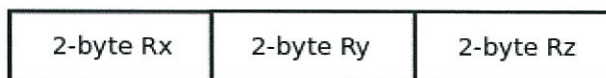
รูป 3.9 Error and Event Characteristic

- 2) Acceleration Characteristic เก็บข้อมูลดิบของค่าความเร่งที่เกิดขึ้นกับหมวกนิรภัย โดยค่าสูงสุด (0x7FFF) จะหมายถึงค่าความเร่ง 16G และค่าต่ำสุด (0x8000) จะหมายถึงค่าความเร่ง -16G โดยจะเก็บอยู่ในรูปแบบดังรูป 3.10



รูป 3.10 Acceleration Characteristic

- 3) Angular Velocity Characteristic เก็บข้อมูลดิบของค่าความเร็วเชิงมุมที่เกิดขึ้นกับหมวกนิรภัย โดยค่าสูงสุด (0x7FFF) จะหมายถึงความเร็วเชิงมุม 2000 degree/sec และค่าต่ำสุด (0x8000) จะหมายถึงค่าความเร็วเชิงมุม -2000 degree/sec โดยจะเก็บอยู่ในรูปแบบดังรูป 3.11



รูป 3.11 Acceleration Characteristic

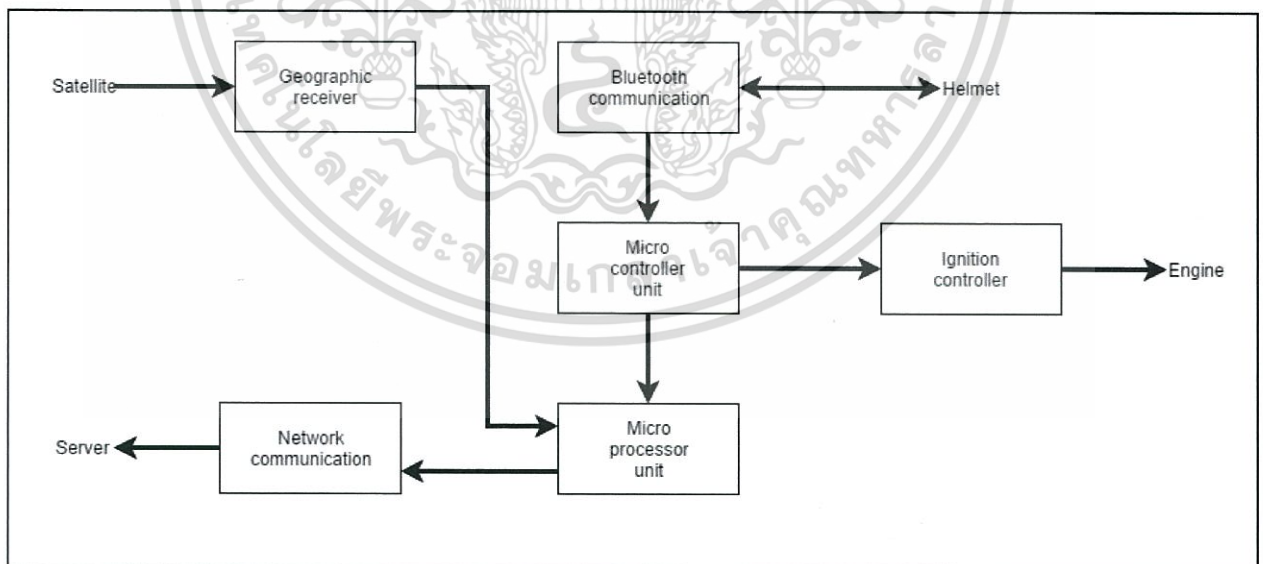
3.4.2. หน่วยควบคุมรถจักรยานยนต์ Motorcycle Control Unit

จากความต้องการของระบบเราจะต้องทำให้ระบบส่วนนี้สามารถ ติดต่อรับข้อมูลกับส่วนหมวกนิรภัย ส่วนควบคุมรถจักรยานยนต์ ส่วนรับข้อมูลพิกัดทางภูมิศาสตร์และ ส่วนที่ใช้ในการเชื่อมต่อ Server

3.4.2.1. การวางส่วนประกอบฮาร์ดแวร์ภายในรถจักรยานยนต์

โดยในแต่ละส่วนที่ประกอบภายในจะมีดังนี้

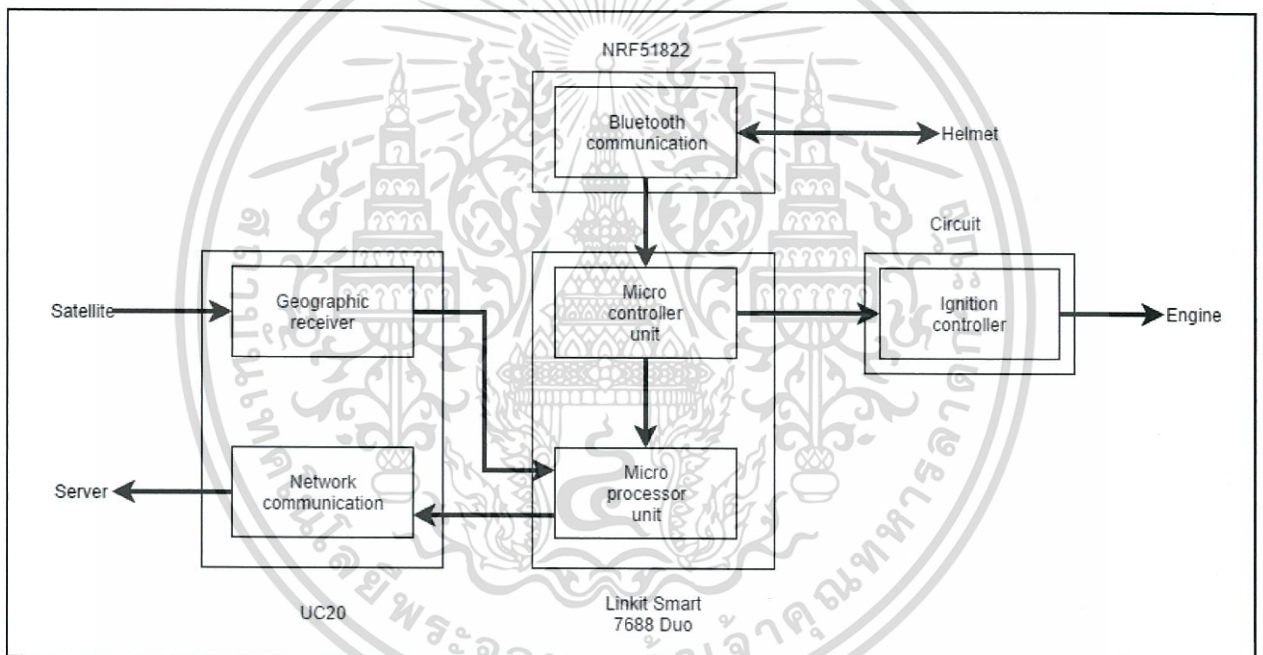
- 1) Bluetooth Communication: รับผิดชอบส่วนติดต่อขอข้อมูลจากหมวกนิรภัย
- 2) Microcontroller Unit: รับผิดชอบนำข้อมูลจาก หมวกนิรภัย เพื่อไปควบคุมส่วนการอนุญาตสตาร์ทเครื่องยนต์และส่งไปให้แก่ส่วน Microprocessor
- 3) Ignition Controller: รับผิดชอบส่วนการเปิดปิดวงจรสตาร์ทเครื่องยนต์
- 4) Microprocessor Unit: รับผิดชอบขอข้อมูลพิกัดทางภูมิศาสตร์ และข้อมูลจาก
- 5) Microcontroller Unit จากนั้นนำข้อมูลส่งไปยัง Server
- 6) Geographic Receiver: รับผิดชอบขอพิกัดปัจจุบันจากดาวเทียมที่โคจรอยู่รอบโลก
- 7) Network Communication: รับผิดชอบติดต่อส่งข้อมูลจาก Microprocessor Unit ไปยัง Server



รูป 3.12 ภาพประกอบการออกแบบในส่วนของการควบคุมรถจักรยานยนต์

ในส่วนการพัฒนาจริง ผู้พัฒนาได้วางฮาร์ดแวร์และวางโมดูลที่ใช้เชื่อมต่อไว้ดังนี้

- 1) Bluetooth Communication ใช้ NRF51822 ซึ่งเป็นทั้งโมดูล Bluetooth Smart และหน่วยประมวลผลสถาปัตยกรรม ARM ภายใน IC ตัวเดียวกัน
- 2) Microcontroller unit และ Microprocessor Unit ใช้บอร์ด Linkit Smart 7688 duo โดยภายในบอร์ดจะมี Atmega32u4 เป็น Microcontroller unit และ MT7688AN เป็น Microprocessor Unit
- 3) Ignition Controller ใช้วงจรควบคุมไฟ 12v ด้วยไฟ 5v ด้วย relay และ Component ต่าง ๆ
- 4) Geographic Receiver & Network Communication ใช้โมดูล UC20 ซึ่งจะมีหน้าที่เป็น GNSS Module และ 3G Module แยกกันทำงาน



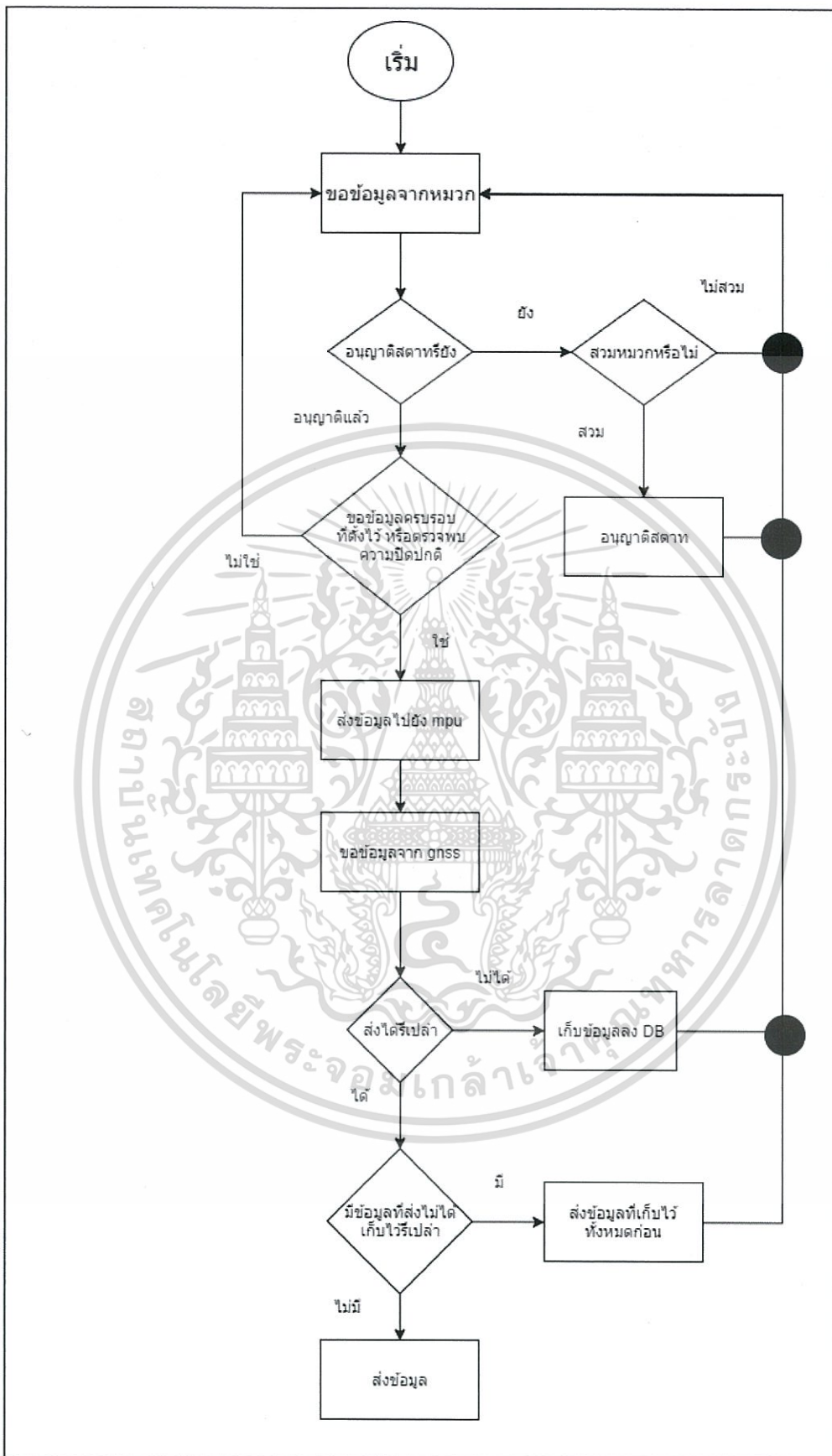
รูป 3.13 ภาพประกอบการวางส่วนรับผิดชอบให้แก่ฮาร์ดแวร์

3.4.2.2. การออกแบบการทำงาน

หลังจากที่เปิดการทำงาน Micro Processor จะจำเป็นต้องใช้เวลาในการ Boot ประมาณ 20 วินาที ส่วน Micro Controller สามารถทำงานได้เลย ซึ่งจะมีการทำงานจะเริ่มต้นจาก Micro Controller ติดต่อขอข้อมูลหวมกนิริภัยผ่าน Bluetooth Communication โดยในข้อมูลที่ขอมานั้นจะมีข้อมูลว่าหวมกนิริภัยถูกใส่อยู่หรือไม่ หากถูกใส่อยู่จะทำการปิดวงจรการสตาร์ทเครื่องยนต์ เมื่อ Micro Controller ทำงานแล้วจะขอข้อมูลจากหวมกนิริภัยเป็นระยะ ๆ ถ้าถึงเวลาที่กำหนด หรือ ได้รับข้อมูลว่าตรวจพบการเคลื่อนไหวของศีรษะผิดปกติจะทำการส่งข้อมูลไปยัง Micro Processor ทันทีเมื่อ Micro Processor ได้ทำการ Boot ขึ้นมาจะทำการเปิดช่องทางการสื่อสาร 3 ช่องทางประกอบด้วย

- 1) ช่องทางการสื่อสารกับ Micro Controller ผ่าน ttyS0
- 2) ช่องทางการสื่อสารกับช่องขอรับข้อมูลพิกัดทางภูมิศาสตร์ของ UC20 ผ่าน ttyUSB1
- 3) ช่องทางการสื่อสารกับช่องทางการสื่อสารกับ Internet ของ UC20 ผ่าน ttyUSB3

เมื่อทำการเปิดช่องทางทั้งหมดแล้วจะทำการ Activate GNSS และ Activate Network Interface ของ Module UC20 หลังจากนั้นจะทำการ Setup MQTT Client และทำการเชื่อมต่อกับ Broker ที่อยู่ฝั่ง Server เมื่อ Micro Processor เตรียมพร้อมแล้วจึงจะสามารถรับข้อมูลจาก Micro Controller และส่งขึ้น Server ได้ และเมื่อได้รับข้อมูลจาก Micro Controller จะทำการขอข้อมูล Geographic Location จาก UC20 และส่งขึ้นไปยัง Server หากไม่สามารถทำการส่งข้อมูลได้ไม่ว่าจะเพราะอยู่ในที่อับสัญญาณ 3G หรือเกิดอุบัติเหตุจนทำให้ส่งข้อมูลไม่ได้จะทำการเก็บข้อมูลที่ส่งไม่ได้ไว้ใน Local Database และเมื่อสามารถทำการส่งได้อีกครั้งจึงจะทำการนำข้อมูลทั้งหมดส่งไปยัง Server อีกครั้ง



รูป 3.14 ภาพประกอบการทำงานในส่วนของตัวควบคุมรถจักรยานยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

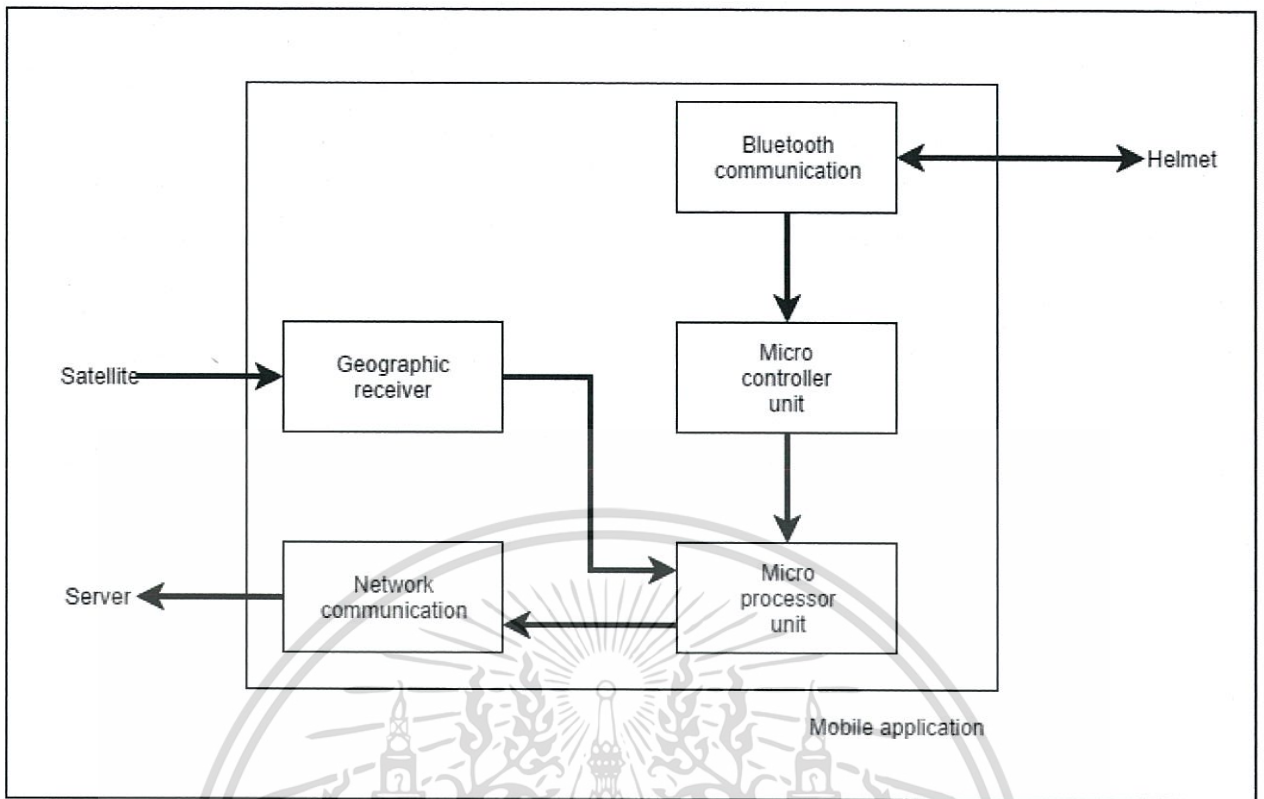
3.4.3. หน่วยส่งข้อมูลและแสดงผลข้อมูลให้แก่ผู้ใช้

จากความต้องการของระบบใน platform ที่ 2 ที่ต้องการให้หมวกนิรภัยเป็น Smart Wearable ดังนั้นส่วนที่อยู่ระหว่างหมวกนิรภัยและ server นั้น จะใช้ Smart Phone เป็นช่องทางการส่งข้อมูลแทนบอร์ดต่าง ๆ ที่กล่าวไปในหน่วยควบคุมรถจักรยานยนต์ ในส่วนนี้จะทำงานเหมือน หน่วยควบคุมรถจักรยานยนต์ เพียงแต่ไม่สามารถควบคุมการสตาร์ทเครื่องยนต์ได้ แต่จะสามารถแสดงผลข้อมูลความเสี่ยงของผู้ขับขี่ได้

3.4.3.1. การใช้งานฮาร์ดแวร์ภายใน Smart Phone

โดยในแต่ละส่วนที่ประกอบภายในจะมีดังนี้

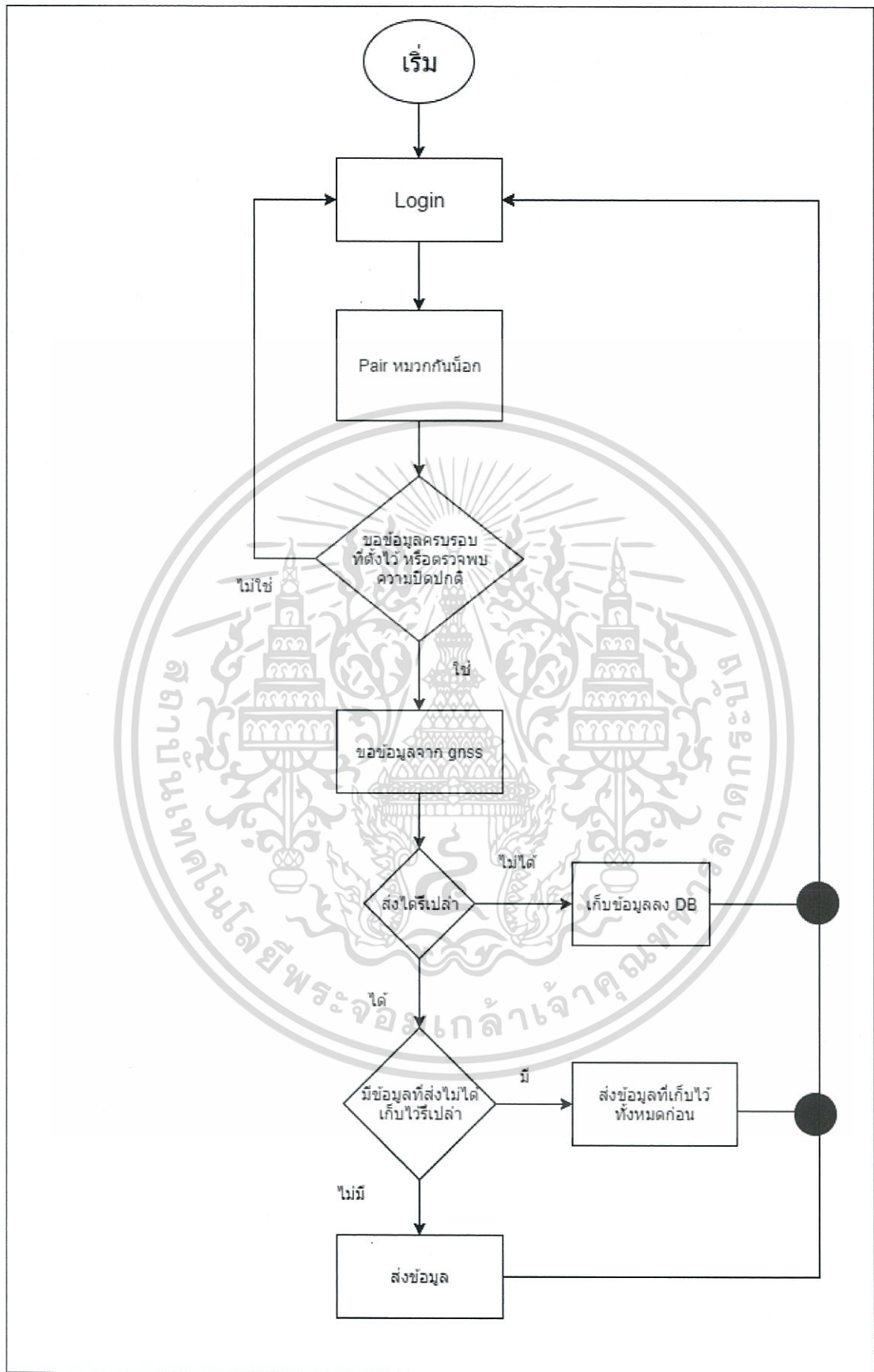
- 1) Bluetooth Communication: รับผิดชอบส่วนติดต่อของข้อมูลจากหมวกนิรภัย โดย Smart Phone ในสมัยนี้ มี Bluetooth module ติดไว้อยู่แล้ว
- 2) Geographic Receiver: รับผิดชอบขอพิกัดปัจจุบันจากดาวเทียมที่โคจรรอบโลก
- 3) Network Communication: รับผิดชอบติดต่อส่งข้อมูลจาก หมวกนิรภัย ไปยัง Server
- 4) Processor Unit: ในส่วนนี้จะทำหน้าที่เหมือน Micro Controller Unit ร่วมกับ Microprocessor Unit ดังนี้ ทำการขอข้อมูลจากหมวกนิรภัยเป็นระยะถ้าถึงเวลาที่กำหนดหรือตรวจพบการเคลื่อนไหวของศีรษะผิดปกติจะทำการส่งข้อมูลไปยัง Server และทำหน้าที่ Setup MQTT Client และเชื่อมต่อกับ Server เมื่อส่งข้อมูลไม่ได้ก็จะทำการเก็บข้อมูลที่ส่งไม่ได้ไว้ใน Local Database
- 5) Visualize: ทำการขอข้อมูลความเสี่ยงของผู้ใช้มาแสดงผลเป็นกราฟให้ผู้ใช้



รูป 3.15 ภาพประกอบการออกแบบในส่วน Mobile application

3.4.3.2. การออกแบบการทำงาน

หลังจากที่เปิดการทำงาน ผู้ใช้จะเป็นต้องเลือกหมวดที่ต้องการจะ Pair ข้อมูลด้วยตนเอง หลังจาก Application ถูกเปิดผู้ใช้ต้อง Login ด้วย Username Password ประจำตัว ระบบจะทำการ Setup MQTT Client, GNSS Module, Bluetooth, Database หลังจากนั้นจึงจะให้ผู้ใช้ทำการเลือกว่าจะ pair ไปยังหมวดขึ้นไหน ระบบจะทำการทำงานอัตโนมัติเหมือนหน่วยควบคุมรถจักรยานยนต์ แต่ส่วนในการแสดงผลข้อมูลผู้ใช้ต้องเป็นผู้เลือกเข้าไปดูด้วยตนเอง



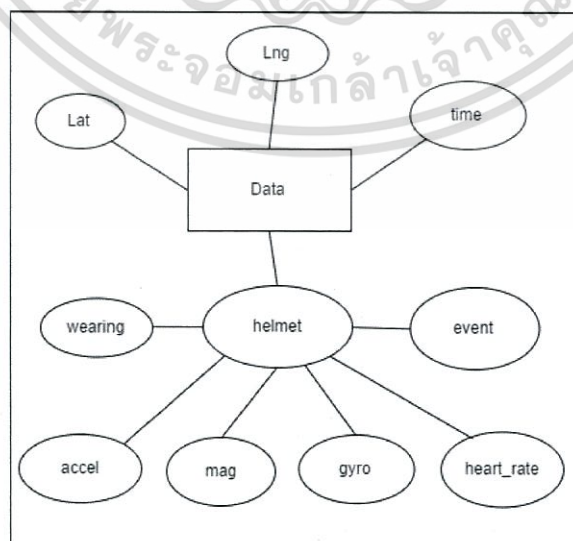
รูป 3.16 ภาพประกอบการทำงานบน Mobile Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3.3. การออกแบบฐานข้อมูลภายใน

สำหรับฐานข้อมูลภายในจะเป็นข้อมูลที่ได้จากหมวดข้อมูลที่ถูกแปลงเป็นข้อมูลให้อ่านได้เข้าใจได้และข้อมูลพิกัดทางภูมิศาสตร์ ซึ่งข้อมูลทั้งหมดจะถูกรวบรวมและส่งขึ้นไปหากครบรอบการขอข้อมูลหรือเกิดเหตุการณ์ที่ผิดปกติ นอกจากนี้ก่อนที่จะทำการส่งข้อมูลก็จะทำการแนบข้อมูลเวลาปัจจุบันในช่วงเวลานั้นไปด้วย เพื่อให้สามารถนำไปคำนวณได้ถูกต้องมากขึ้น โดยข้อมูลที่อยู่ในฐานข้อมูลภายในจะถูกเก็บไว้เมื่อข้อมูลที่จะถูกส่งไปนั้นไม่สามารถส่งขึ้นไปได้ และเมื่อจะทำการส่งข้อมูลจะเช็คว่ามีข้อมูลที่ไม่ได้ส่งอยู่ในฐานข้อมูลหรือไม่ถ้ามีข้อมูลอยู่ในฐานข้อมูลจะแสดงว่ามีข้อมูลก่อนหน้าที่ยังไม่ถูกส่งขึ้นไปก็จะทำการส่งข้อมูลที่อยู่ในฐานข้อมูลขึ้นไปทั้งหมดและส่งข้อมูลล่าสุดขึ้นไป ซึ่งเมื่อดำเนินการส่งข้อมูลที่ค้างไว้แต่ละครั้งจะทำการลบข้อมูลที่อยู่ภายในออกไปด้วย โดยฐานข้อมูลที่ถูกเก็บมีรายละเอียดดังนี้

- 1) Lat: ค่า latitude ของรถจักรยานยนต์ในขณะนั้น
- 2) Lng: ค่า longitude ของรถจักรยานยนต์ในขณะนั้น
- 3) Helmet: ข้อมูลที่ได้จาก helmet ในขณะนั้น
- 4) wearing: สถานะการสวมใส่หมวกกันน็อกในขณะนั้น
- 5) accel: ค่า accelerate ของศีรษะผู้ขับขี่ในขณะนั้น
- 6) mag: ค่า magnetic ของศีรษะผู้ขับขี่ในขณะนั้น
- 7) gyro: ค่า gyro ของศีรษะผู้ขับขี่ในขณะนั้น
- 8) heart_rate: ค่า heartrate ของผู้ขับขี่ในขณะนั้น
- 9) event: ค่าเหตุการณ์ที่สามารถตรวจจับได้จากการคำนวณภายในหมวกกันน็อก
- 10) time: ค่าเวลาในขณะนั้น



รูป 3.17 ภาพประกอบการออกแบบฐานข้อมูลภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4. การออกแบบ Monitor And Analysis Unit

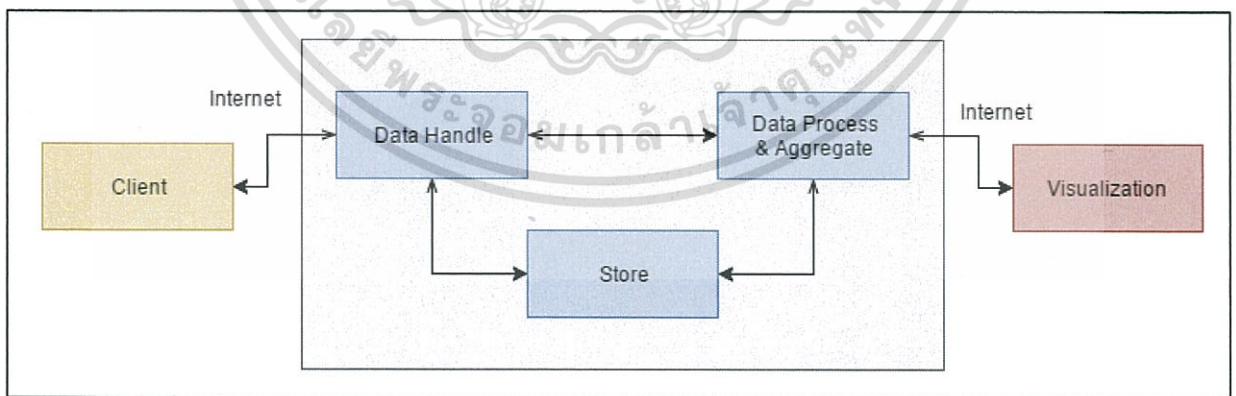
3.4.4.1. ภาพรวมขั้นตอนการทำงานของระบบ

ในการทำงานของระบบเมื่อระบบได้รับข้อมูลจาก Client ผู้ใช้งานหมวกนิรภัยจะทำการส่งข้อมูลผ่านเครือข่าย Internet จากนั้นจะเข้าสู่ Server และทำการวิเคราะห์ข้อมูลและส่งไปหาผู้ใช้งาน ซึ่งภายในระบบของ Server จะมีหน้าที่ในการทำงานของกระบวนการหลักอยู่ 3 ส่วนหลัก ๆ คือ

Data Handle คือการจัดเก็บข้อมูลโดยจะมีการรับข้อมูลจาก Client โคนผ่านทาง Mobile Application ผ่านรถจักรยานยนต์และผ่านทาง Web Application ซึ่งจะมีการรับส่งข้อมูลด้วยโปรโตคอล Mqtt และ Http เป็นหลักเมื่อหมวกนิรภัยตรวจเจอเหตุการณ์ที่สัมพันธ์ต่อการเกิดอุบัติเหตุหรือเมื่อต้องการส่งข้อมูลจะทำการส่งข้อมูลไปที่ Server และส่วนนี้จะทำหน้าที่ในการรับข้อมูล

Data Process And Aggregate เป็นการประมวลผล วิเคราะห์และ จัดเรียงข้อมูลที่ได้มาเพื่อให้อยู่ในรูปแบบที่ใช้งานได้และเหมาะสมในการจัดเก็บข้อมูลเช่น เมื่อระบบได้รับข้อมูลความเร็วมา ก็จะมีการวิเคราะห์เหตุการณ์ว่าเหตุการณ์นั้น ๆ มีการใช้ความเร็วมากขึ้นจะมีผลต่อความเสี่ยงในการเกิดอุบัติเหตุมากขึ้นเพียงใด หรือหากมีการขับขี่ที่มีการวิเคราะห์แล้วว่าผู้ขับขี่กำลังขับขี่อยู่ในถนนที่มีความเสี่ยงในการเกิดอุบัติเหตุ ซึ่งการวิเคราะห์นี้จำเป็นที่จะต้องมีการดึงข้อมูลการขับขี่เดิมที่เคยเก็บไว้มาทำการวิเคราะห์ร่วมด้วย จากนั้นเมื่อวิเคราะห์และจัดเรียงข้อมูลแล้วก็จะทำการจัดเก็บในฐานข้อมูล และทำการแสดงผลในรูปแบบของ Web Application และ Mobile Application ต่อไป

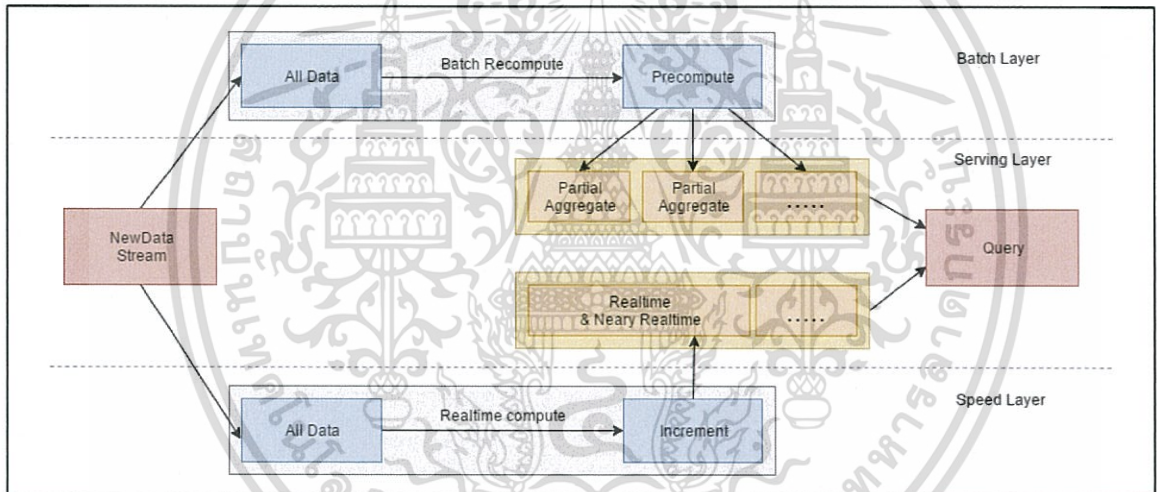
Store คือการจัดเก็บข้อมูล โดยจะทำหน้าที่หลักในการเป็นตัวจัดการการเก็บข้อมูลว่าข้อมูลแบบไหนควรมีการจัดเก็บอย่างไรเนื่องจาก ข้อมูลบางชนิดมีความถี่ในการเรียกใช้งานบ่อยบางชนิดใช้งานเพียงแค่ครั้งเดียว ก็จะมีรูปแบบการเก็บข้อมูลที่แตกต่างกันออกไป



รูป 3.18 ภาพรวมการทำงานของระบบ

3.4.4.2. รูปแบบสถาปัตยกรรมของระบบ

เนื่องจากการทำงานของระบบเป็นระบบที่มีความต้องการในการตอบสนองที่รวดเร็ว อีกทั้งยังมีโอกาสที่จะมีข้อมูลเข้ามาในระบบพร้อม ๆ กันในช่วงเวลาหนึ่งมาก ๆ พร้อมทั้งต้องทำการเก็บข้อมูลย้อนหลังเพื่อทำการวิเคราะห์ข้อมูลในการจับจีและข้อมูลความปลอดภัย ทำให้ระบบที่จะพัฒนาขึ้นมาขึ้นจำเป็นต้องมีความสามารถในการทำงานหลัก ๆ คือ สามารถทำงานที่รวดเร็วสามารถส่งข้อมูล ไปยังผู้ใช้งานได้ใกล้เคียงความเป็นจริงสามารถวิเคราะห์ข้อมูลร่วมกับข้อมูลย้อนหลังและหาความสัมพันธ์ได้รวมเร็วจากการวิเคราะห์ความต้องการข้างต้นจึงได้เลือก Lambda Architecture ซึ่งเป็นสถาปัตยกรรมที่ถูกนำมาใช้ในระบบที่มีการทำงานที่มีการประมวลผลแบบ Real Time มาเป็นส่วนหนึ่งของระบบซึ่งจุดเด่นของระบบนี้คือมีความเสถียรในการใช้งาน สามารถรองรับการใช้งานกับข้อมูลที่มีความหลากหลายและมีปริมาณสูงรวมทั้งความรวดเร็วในการอ่านเขียนข้อมูลซึ่ง ระบบจะมีส่วนประกอบไปด้วย 5 ส่วนหลักดังนี้



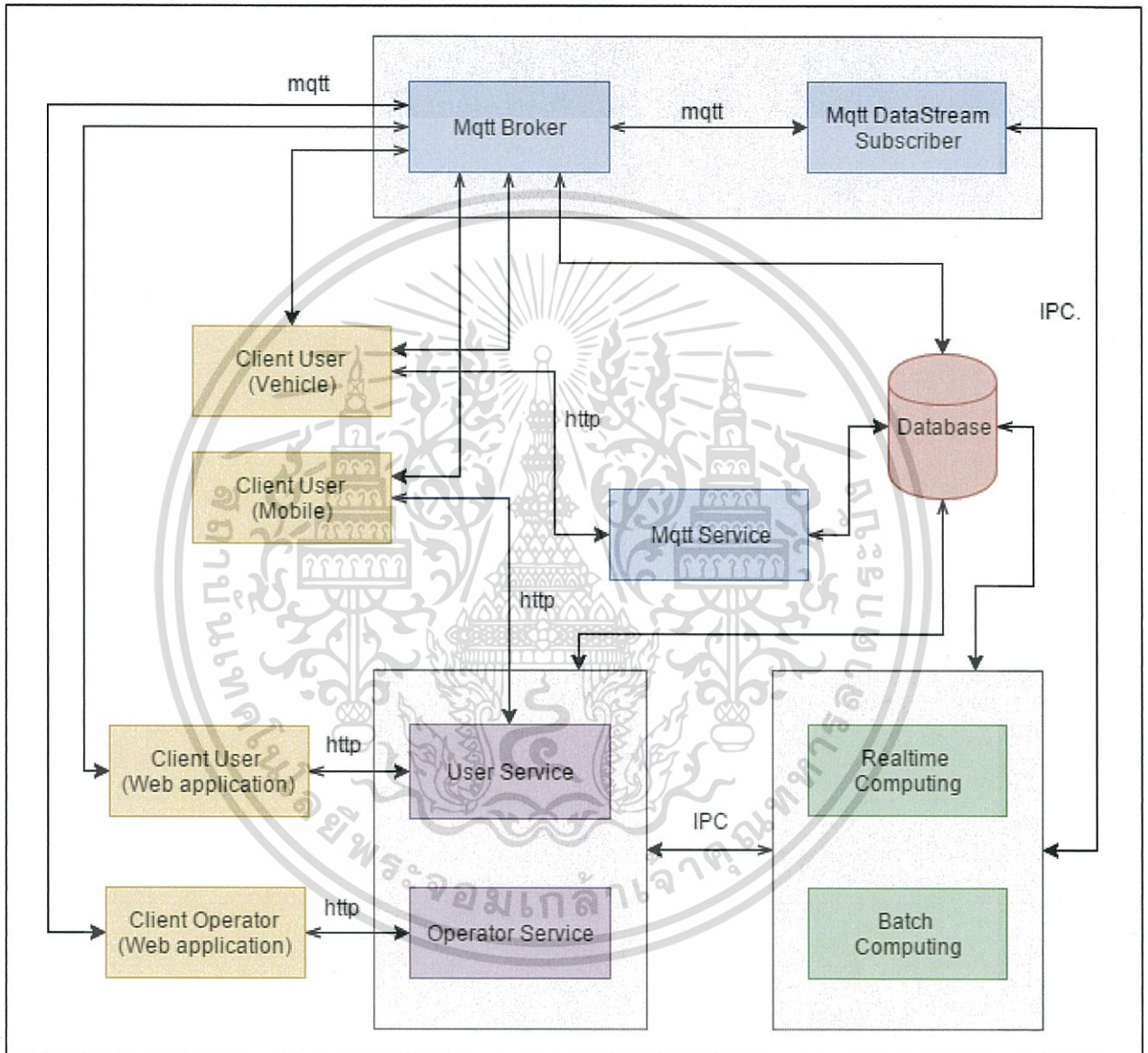
รูป 3.19 สถาปัตยกรรมของระบบ

- 1) New Data Stream คือข้อมูลชนิดต่าง ๆ ที่เข้ามาในระบบโดยข้อมูลที่ได้จะมีการจัดแบ่งไปในส่วนที่แตกต่างกัน เช่นถ้าข้อมูลนั้นต้องทำการ Process ร่วมกับข้อมูลก่อนหน้าที่เคยเก็บไว้แล้วและมีขนาดใหญ่และไม่ต้องการความเร็วในการวิเคราะห์และการตอบสนองก็จะถูกส่งให้ Batch Layer ส่วนข้อมูลที่มีความต้องการในการวิเคราะห์น้อย ๆ และต้องการข้อมูลที่มีความเร็วในการตอบสนองมาก ๆ จะส่งให้ Speed Layer ทำงาน โดยการทำงานของแต่ละส่วนก็จะมีกรข้อดีข้อเสียแตกต่างกัน

- 2) Batch Layer มีการทำงานเป็นลักษณะที่ไม่ต้องการความเร็วมมาก และจำเป็นที่ต้องมีการใช้การ Process ร่วมกับข้อมูลอื่น ๆ ซึ่งแบ่งหน้าที่ออกเป็น 2 ส่วนหลัก ๆ คือ
 - 2.1) ทำการจัดการข้อมูลที่เป็นข้อมูลดิบ ไม่จำเป็นต้องมีการประมวลผลมากและสามารถเพิ่มได้อย่างเดียวเท่านั้น
 - 2.2) ประมวลผลข้อมูลตามรายงานหรือข้อมูลรูปแบบต่าง ๆ ที่ต้องการการคำนวณซึ่งทำงานแบบ Batch Processing
- 3) Serving Layer คือส่วนที่ทำการรวบรวมข้อมูลและจัดเรียงข้อมูลที่ผ่านการ Process มาแล้วจากการทำงานของ Speed Layer และ Batch Layer เพื่อให้การอ่านหรือดึงข้อมูลทำได้อย่างรวดเร็ว
- 4) Speed Layer คือส่วนที่เข้ามาเพิ่มเติมสำหรับระบบที่มีการเปลี่ยนแปลงข้อมูลที่สูง และต้องการข้อมูลแบบ Real Time ซึ่งเข้ามาเพื่อสนับสนุน Serving Layer ให้สามารถทำงานได้อย่างอิสระมากขึ้น
- 5) Query ทำหน้าที่ Request ข้อมูลและรวมข้อมูลจาก Serve Layer และ Speed Layer เพื่อส่งข้อมูลกลับไปตามที่เราต้องการ

3.4.4.3. ส่วนประกอบของโครงสร้างระบบโดยละเอียด

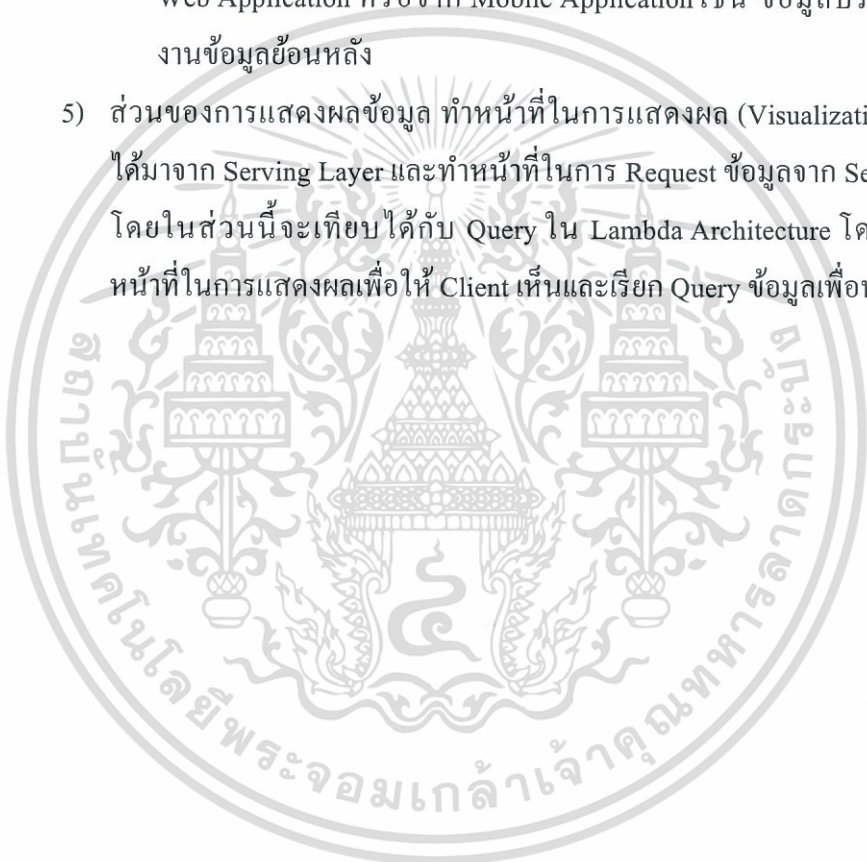
ภายในระบบของการประมวลผลฝั่ง Server จะถูกแบ่งออกเป็น 5 ส่วนหลัก ๆ แยกตามหน้าที่การทำงานและความเกี่ยวข้องกันตาม Lambda Architecture ซึ่งจะถูกแบ่งออกเป็น 5 ส่วนดังรูปที่ 3.20 ดังนี้



รูป 3.20 ส่วนประกอบโครงสร้างของระบบโดยละเอียด

- 1) ส่วนการรับข้อมูลจาก Client เป็นส่วนที่รับข้อมูล Client จาก Internet ที่ถูกส่งผ่านเครือข่าย 3G เข้าสู่ Server ในส่วนนี้จะเทียบได้กับ Data ในแผนภาพของ Lambda Architecture และทำการติดต่อกับส่วนหลัก 2 ส่วนในแผนภาพคือ
 - 1.1) Mqtt Broker เป็น Process ที่ทำหน้าที่ในการรับข้อมูลจาก Client ผ่านโปรโตคอล Mqtt และทำการจัดการช่องทาง (Topic) ของข้อมูลที่ได้รับจาก Client
 - 1.2) Mqtt Data Stream Subscriber เป็นส่วนที่ทำหน้าที่ในการรับข้อมูลจาก Mqtt Broker เพื่อทำการแยกช่องทาง (Topic) ของข้อมูลจากนั้นก็ทำการส่งข้อมูลที่ได้ไปในส่วนของ Batch Layer และ Speed Layer
- 2) ส่วนการวิเคราะห์ข้อมูลและจัดเรียงข้อมูล ส่วนนี้เป็นส่วนที่รับข้อมูลจาก Mqtt DataStream Subscriber เพื่อทำการวิเคราะห์ข้อมูลซึ่งเทียบได้กับ Batch Layer และ Speed Layer ในแผนภาพของ Lambda Architecture ซึ่งในส่วนนี้จะมีส่วนที่เกี่ยวข้อง 2 ส่วนคือ
 - 2.1) Real Time Computing ทำหน้าที่ในการเปลี่ยนแปลงหรือจัดเรียงข้อมูลใหม่ที่ไม่มีความสัมพันธ์กับข้อมูลเก่าที่ผ่านมาและต้องการความเร็วในการอ่านเขียนเช่น ความเร็ว สถานะ ความเร่ง ทิศทาง หรือ พิกัด
 - 2.2) Batch Computing ทำหน้าที่ในการวิเคราะห์และหาความสัมพันธ์ของข้อมูลใหม่ที่มีความสัมพันธ์กับข้อมูลเก่าเช่น ถนนเส้นที่กำลังขับขี้อยู่จะมีโอกาสเสี่ยงที่จะเกิดอุบัติเหตุได้มากเท่าไร หากเทียบกับความเร็วขณะนั้น ๆ หรือ ถนนที่กำลังขับขี้อยู่ที่นั้นเกิดอุบัติเหตุบ่อยหรือไม่และควรทำเช่นไรในการขับขี่ให้ปลอดภัยขึ้น
- 3) ส่วนของการจัดการการเก็บข้อมูลผู้ใช้ เป็นส่วนที่จะมีการทำงานหลักอยู่ที่ Mqtt Service ทำหน้าที่ในการจัดการ Username, Password และการทำ Authorization ของ Mqtt Broker

- 4) ส่วนการติดต่อกับ Client ส่วนนี้จะเป็นส่วนที่มีการติดต่อและส่งข้อมูลให้ Client ไปทำการแสดงผลผ่าน Web Application และ Mobile Application ซึ่งเทียบได้กับ Serving Layer ของ Lambda Architecture ซึ่งส่วนนี้จะแบ่งออกเป็น 2 ส่วนหลัก ๆ คือ
 - 4.1) User Service ทำหน้าที่ให้ข้อมูลต่าง ๆ กับ Client เมื่อมีการ Request จาก
 - 4.2) Web Application หรือจาก Mobile Application เช่น ข้อมูลประวัติการใช้งาน หรือข้อมูลย้อนหลัง
 - 4.3) Operator Service ทำหน้าที่ให้ข้อมูลต่าง ๆ กับ Client เมื่อมีการ Request จาก Web Application หรือจาก Mobile Application เช่น ข้อมูลประวัติการใช้งานข้อมูลย้อนหลัง
- 5) ส่วนของการแสดงผลข้อมูล ทำหน้าที่ในการแสดงผล (Visualization) ข้อมูลที่ได้มาจาก Serving Layer และทำหน้าที่ในการ Request ข้อมูลจาก Serving Layer โดยในส่วนนี้จะเทียบได้กับ Query ใน Lambda Architecture โดยส่วนนี้ทำหน้าที่ในการแสดงผลเพื่อให้ Client เห็นและเรียก Query ข้อมูลเพื่อนแสดงผล



บทที่ 4

การทดลองและผลการทดลอง

4.1. ทดสอบการเชื่อมต่อระหว่างหมวกนิรภัยและรถจักรยานยนต์

ทดสอบการเชื่อมต่อแบบไร้สายระหว่างหมวกนิรภัยและรถจักรยานยนต์ด้วยการเปิดระบบบนหมวกนิรภัย จากนั้นจับเวลาที่รถจักรยานยนต์ใช้ในการสร้างการเชื่อมต่อกับหมวกนิรภัยโดยตั้งค่าสูงสุดไว้ที่ 20 วินาที หากภายใน 20 วินาทีหมวกนิรภัยและรถจักรยานยนต์ไม่สามารถสร้างการเชื่อมต่อได้ จะถือว่าสร้างการเชื่อมต่อไม่สำเร็จ การทดสอบจะแบ่งออกเป็น 4 แบบ แตกต่างกันไปตามระยะห่างระหว่างหมวกนิรภัยและรถจักรยานยนต์ ซึ่งระยะดังกล่าวได้แก่ ระยะที่วางหมวกนิรภัยไว้บนรถจักรยานยนต์ ระยะที่หมวกนิรภัยอยู่ห่างจากรถจักรยานยนต์ 3 เมตร ระยะที่หมวกนิรภัยอยู่ห่างจากรถจักรยานยนต์ 10 เมตร และระยะที่หมวกนิรภัยอยู่ห่างจากรถจักรยานยนต์ 50 เมตร แต่ละแบบทำการทดสอบ 20 ครั้ง ซึ่งหลังจากหมวกนิรภัยเปิดการทำงาน ระบบจะรับการเชื่อมต่อจากรถจักรยานยนต์โดยทันที

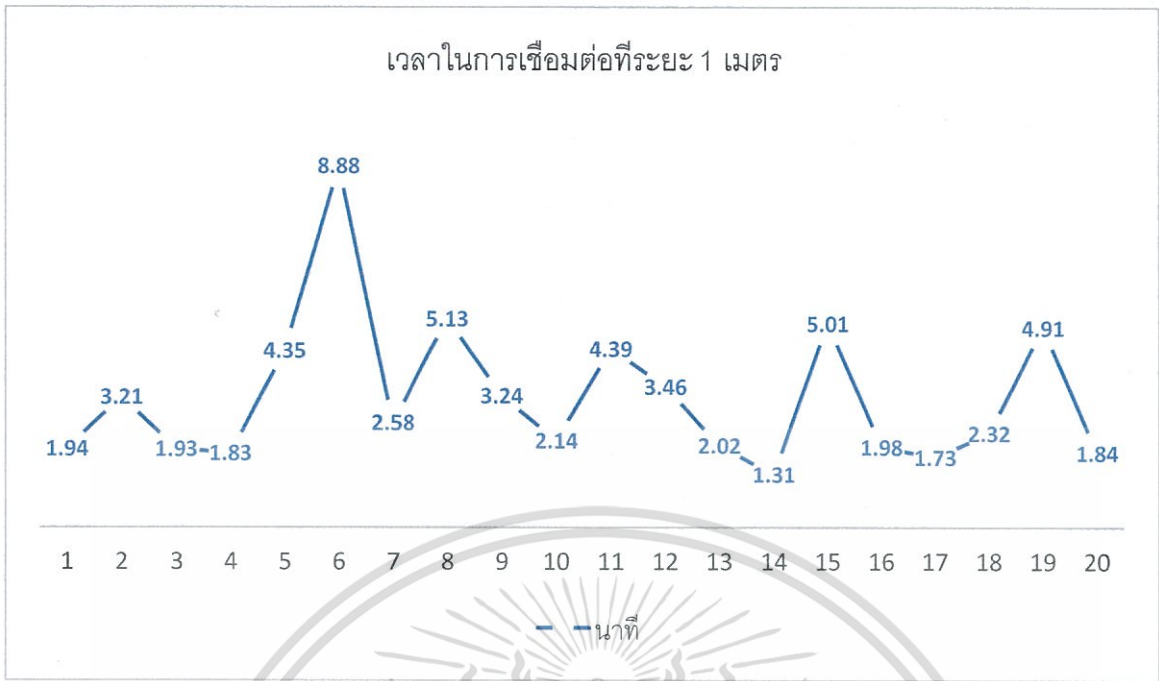
สมมุติฐาน: หากระยะห่างระหว่างหมวกนิรภัยและรถจักรยานยนต์ยังคงอยู่ในมาตรฐานระยะปฏิบัติการที่รองรับในมาตรฐาน Bluetooth Smart หรือ Bluetooth 4.0 คือถ้าน้อยกว่า 50 เมตร การเชื่อมต่อระหว่าง Helmet Sensors Unit และ Motorcycle Control Unit จะเกิดขึ้นได้เสมอ

ซึ่งได้ทำการทดสอบโดยการทำการทดสอบการเชื่อมต่อว่าสามารถเชื่อมต่อในระยะที่ต่างกันซึ่งได้ผลการทดสอบดังนี้

4.1.1. การทดสอบการเชื่อมต่อในระยะที่ห่างจากรถจักรยานยนต์ 1 เมตร จำนวน 20 ครั้ง

ตารางที่ 4.1 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 1 เมตร

ครั้งที่	เวลาที่ใช้ (วินาที)	เชื่อมต่อ
1	1.94	/
2	3.21	/
3	1.93	/
4	1.83	/
5	4.35	/
6	8.88	/
7	2.58	/
8	5.13	/
9	3.24	/
10	2.14	/
11	4.39	/
12	3.46	/
13	2.02	/
14	1.31	/
15	5.01	/
16	1.98	/
17	1.73	/
18	2.32	/
19	4.91	/
20	1.84	/



รูปที่ 4.1 กราฟความสัมพันธ์ของการทดสอบ ระยะ 1 เมตรกับเวลา

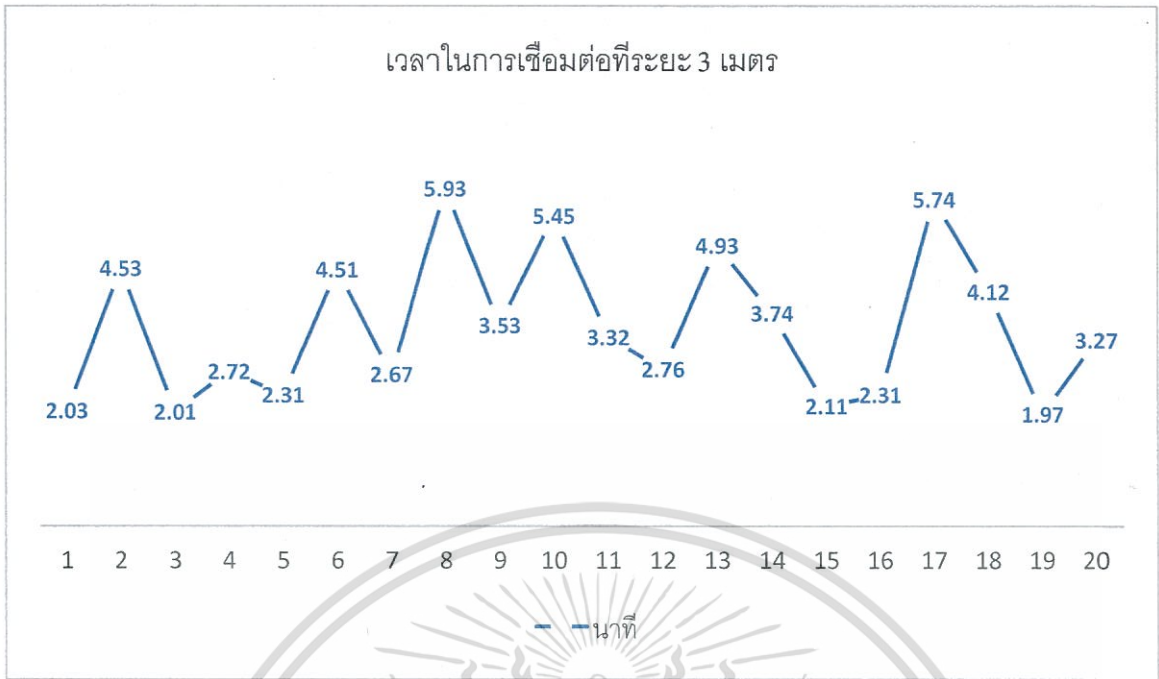
ตารางที่ 4.2 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 1 เมตร

ทั้งหมด	เชื่อมต่อ	ไม่เชื่อมต่อ	ถูกต้อง(%)
20	20	0	100

4.1.2. การทดสอบการเชื่อมต่อในระยะที่ห่างจากรถจักรยานยนต์ 3 เมตร

ตารางที่ 4.3 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 3 เมตร

ครั้งที่	เวลาที่ใช้ (วินาที)	เชื่อมต่อ
1	2.03	/
2	4.53	/
3	2.01	/
4	2.72	/
5	2.31	/
6	4.51	/
7	2.67	/
8	5.93	/
9	3.53	/
10	5.45	/
11	3.32	/
12	2.76	/
13	4.93	/
14	3.74	/
15	2.11	/
16	2.31	/
17	5.74	/
18	4.12	/
19	1.97	/
20	3.27	/



รูปที่ 4.2 กราฟความสัมพันธ์ของการทดสอบ ระยะ 3 เมตรกับเวลา

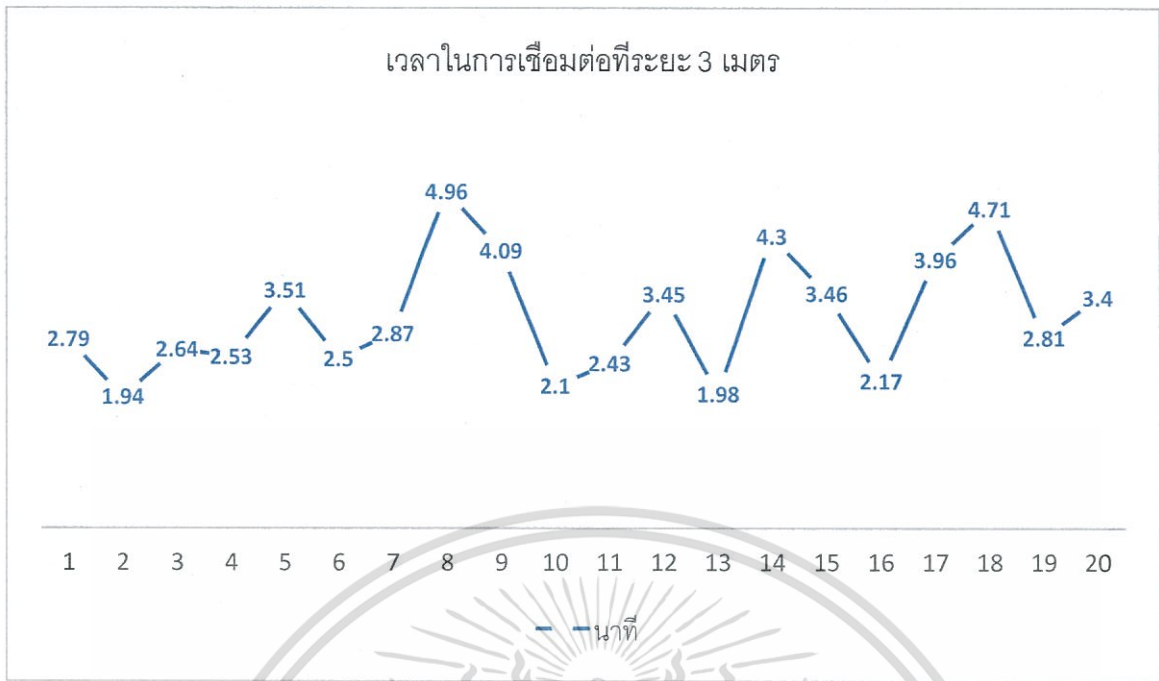
ตารางที่ 4.4 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 3 เมตร

ทั้งหมด	เชื่อมต่อ	ไม่เชื่อมต่อ	ถูกต้อง(%)
20	20	0	100

4.1.3. การทดสอบการเชื่อมต่อในระยะที่ห่างจากรถจักรยานยนต์ 10 เมตร

ตารางที่ 4.5 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 10 เมตร

ครั้งที่	เวลาที่ใช้ (วินาที)	เชื่อมต่อ
1	2.79	/
2	1.94	/
3	2.64	/
4	2.53	/
5	3.51	/
6	2.50	/
7	2.87	/
8	4.96	/
9	4.09	/
10	2.10	/
11	2.43	/
12	3.45	/
13	1.98	/
14	4.30	/
15	3.46	/
16	2.17	/
17	3.96	/
18	4.71	/
19	2.81	/
20	3.40	/



รูปที่ 4.3 กราฟความสัมพันธ์ของการทดสอบระยะ 10 เมตรกับเวลา

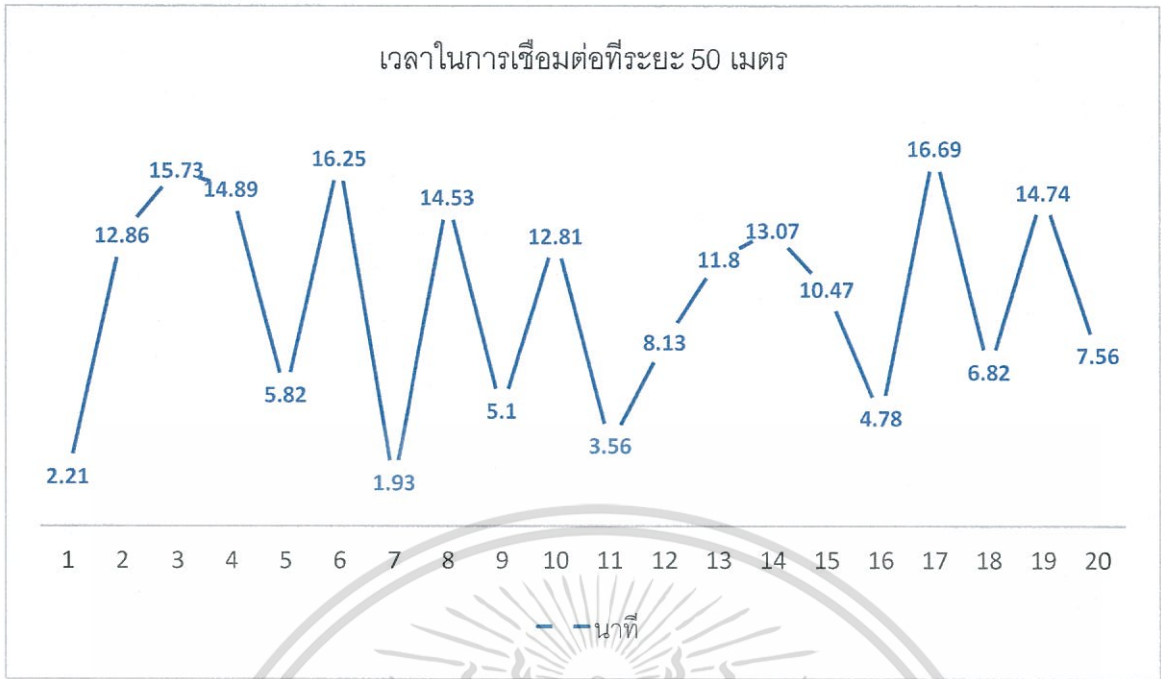
ตารางที่ 4.6 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 10 เมตร

ทั้งหมด	เชื่อมต่อ	ไม่เชื่อมต่อ	ถูกต้อง(%)
20	20	0	100

4.1.4. การทดสอบการเชื่อมต่อในระยะที่ห่างจากรถจักรยานยนต์ 50 เมตร

ตารางที่ 4.7 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 50 เมตร

ครั้งที่	เวลาที่ใช้ (วินาที)	เชื่อมต่อ
1	2.21	/
2	12.86	/
3	15.73	/
4	14.89	/
5	5.82	/
6	16.25	/
7	1.93	/
8	14.53	/
9	5.10	/
10	12.81	/
11	3.56	/
12	8.13	/
13	11.80	/
14	13.07	/
15	10.47	/
16	4.78	/
17	16.69	/
18	6.82	/
19	14.74	/
20	7.56	/



รูปที่ 4.4 กราฟความสัมพันธ์ของการทดสอบระยะ 50 เมตรกับเวลา

ตารางที่ 4.8 ผลการทดสอบการเชื่อมต่อของหมวกในระยะ 50 เมตร

ทั้งหมด	เชื่อมต่อ	ไม่เชื่อมต่อ	ถูกต้อง(%)
20	20	0	100

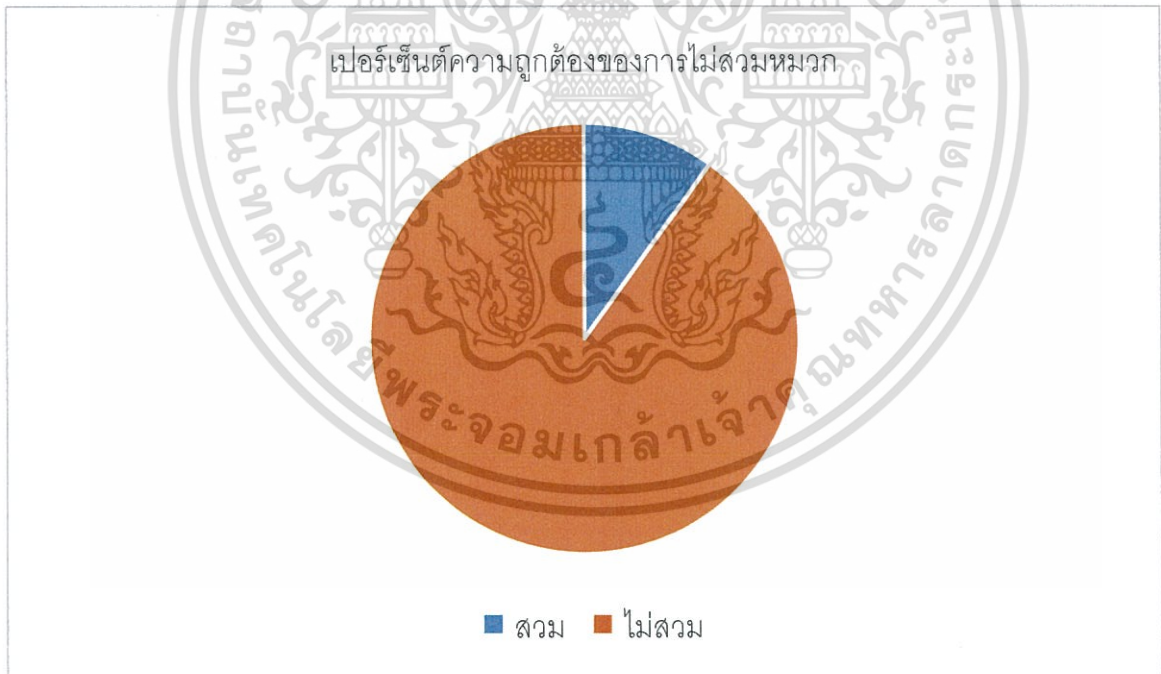
4.2. ทดสอบความแม่นยำในการตรวจสอบสถานะการสวมใส่หมวกนิรภัย

ทดสอบว่าระบบสามารถตรวจสอบว่าผู้ขับขี่กำลังสวมหมวกนิรภัยอยู่หรือไม่ ได้อย่างถูกต้องและแม่นยำแค่ไหน การทดสอบทำโดยการตรวจสอบที่จอมอนิเตอร์ซึ่งรับข้อมูลมาจาก Server ว่าตอบสนองกับการสวมหมวกนิรภัยอย่างไรบ้าง โดยหากสวมหมวกแล้วที่จอมอนิเตอร์แสดงสถานะว่าผู้ขับขี่สวมหมวก ถือว่าผลออกมาถูกต้อง แต่ถ้าผู้ขับขี่สวมหมวกแล้วที่จอมอนิเตอร์แสดงผลเป็นผู้ขับขี่ยังไม่สวมหมวก ถือว่าผลออกมาไม่ถูกต้อง การทดสอบแบ่งเป็น 2 แบบคือ การทดสอบไม่สวมหมวกนิรภัย และการทดสอบสวมหมวกนิรภัย ซึ่งแต่ละแบบทำการทดสอบจำนวน 20 ครั้ง

4.2.1. การทำการทดสอบการไม่สวมหมวกนิรภัย

ตารางที่ 4.9 ผลการทดสอบการไม่สวมหมวก

ทั้งหมด	สวม	ไม่สวม	ถูกต้อง(%)
20	2	18	90

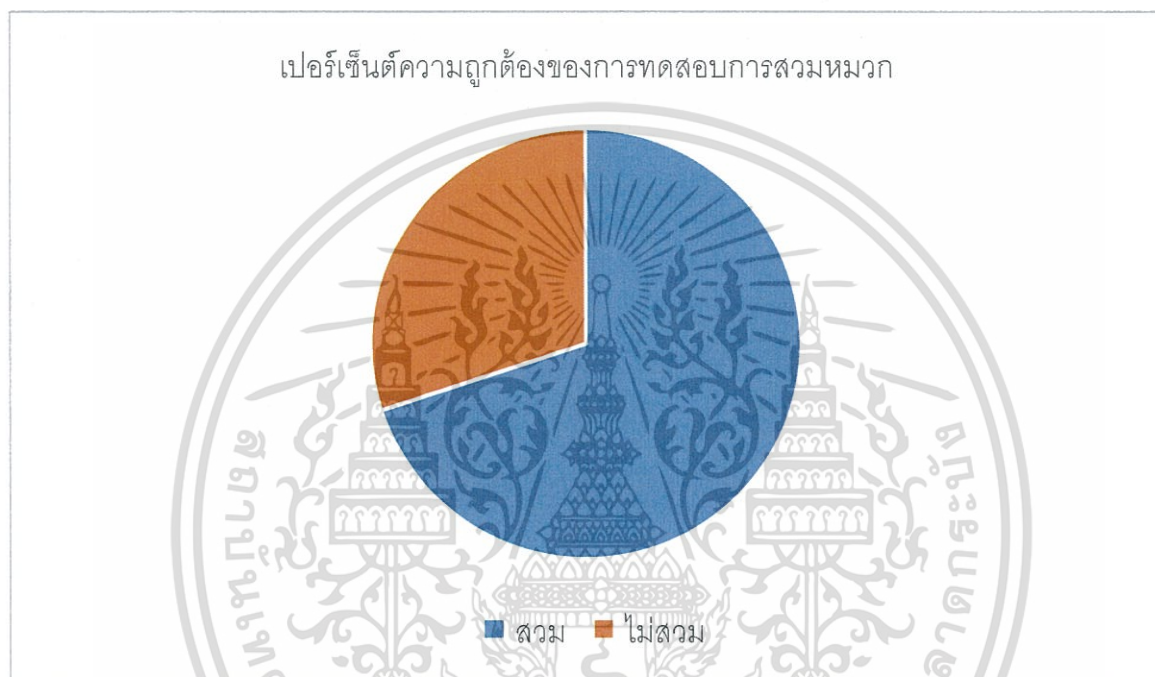


รูปที่ 4.5 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบการไม่สวมหมวก 20 ครั้ง

4.2.2. การทำการทดสอบการสวมหมวกนิรภัย

ตารางที่ 4.10 ผลการทดสอบการสวมหมวก

ทั้งหมด	สวม	ไม่สวม	ถูกต้อง(%)
20	14	6	70



รูปที่ 4.6 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบการสวมหมวก 20 ครั้ง

4.3. ทดสอบความแม่นยำในการตรวจจับความผิดปกติขณะขับขี่

ทดสอบว่าระบบสามารถตรวจจับความผิดปกติระหว่างขับขี่ที่เกิดขึ้นได้แม่นยำมากเท่าไร ซึ่งการทดสอบทำโดยการตรวจสอบที่จอมอนิเตอร์ซึ่งรับข้อมูลมาจาก Server ว่าตอบสนองกับการเคลื่อนไหวที่ผิดปกติของศีรษะอย่างไรบ้าง โดยการทดสอบจะแบ่งเป็น 3 แบบ ซึ่งแตกต่างกันตามลักษณะของความผิดปกติได้แก่ การเคลื่อนไหวปกติ มุมเอียงของศีรษะที่ผิดปกติ และความเร่งที่ผิดปกติ ซึ่งแต่ละแบบจะทำการทดสอบจำนวน 20 ครั้ง

4.3.1. การทดสอบการเคลื่อนไหวปกติ

เป็นการทดสอบทำโดยการตั้งหมวกนิรภัยให้วางอยู่บนพื้นระดับหนึ่ง ๆ ซึ่งถือเป็นการเคลื่อนไหวที่ปกติ

ตารางที่ 4.11 ผลการทดสอบการเคลื่อนไหวปกติ

ทั้งหมด	ผิดปกติ	ปกติ	ถูกต้อง(%)
20	1	19	95

เปอร์เซ็นต์ความถูกต้องของการทดสอบการเคลื่อนไหวปกติ



■ ผิดปกติ ■ ปกติ

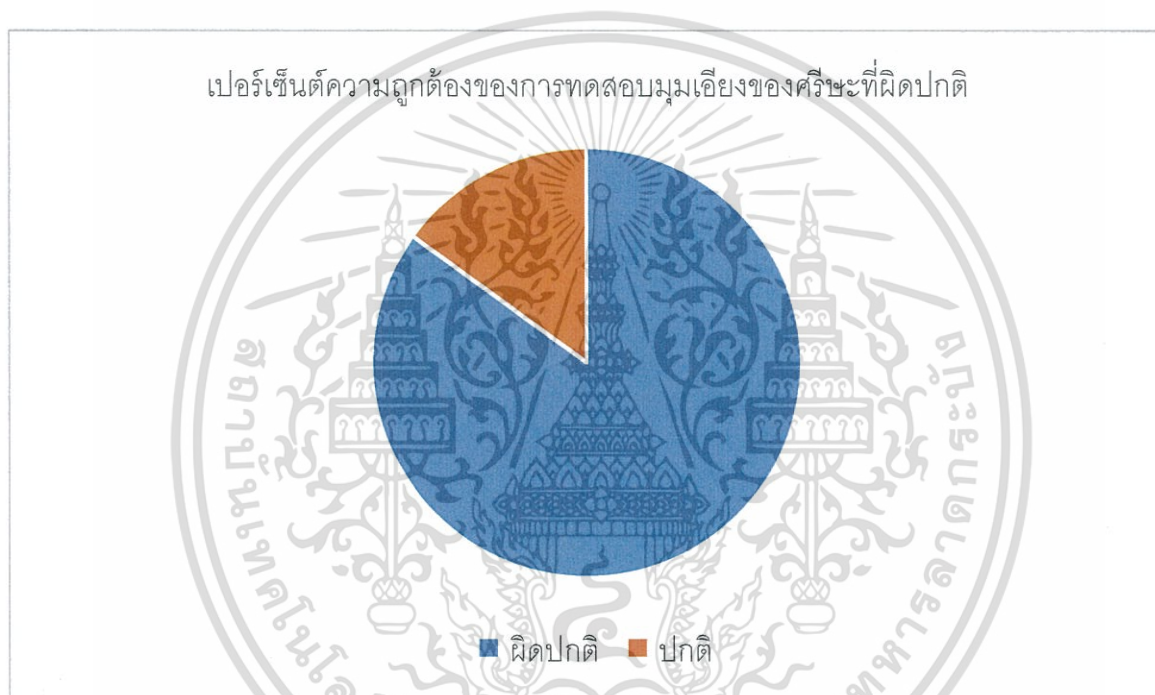
รูปที่ 4.7 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบการเคลื่อนไหวปกติ 20 ครั้ง

4.3.2. การทดสอบมุมเอียงของศีรษะที่ผิดปกติ

เป็นการทดสอบทำโดยการวางหมวกให้เอียงมากกว่า 45 องศาในแกน Pitch และ Roll ซึ่งถือเป็นมุมที่ผิดปกติในระหว่างการขับขี่

ตารางที่ 4.12 ผลการทดสอบมุมเอียงของศีรษะที่ผิดปกติ

ทั้งหมด	ผิดปกติ	ปกติ	ถูกต้อง(%)
20	17	3	85



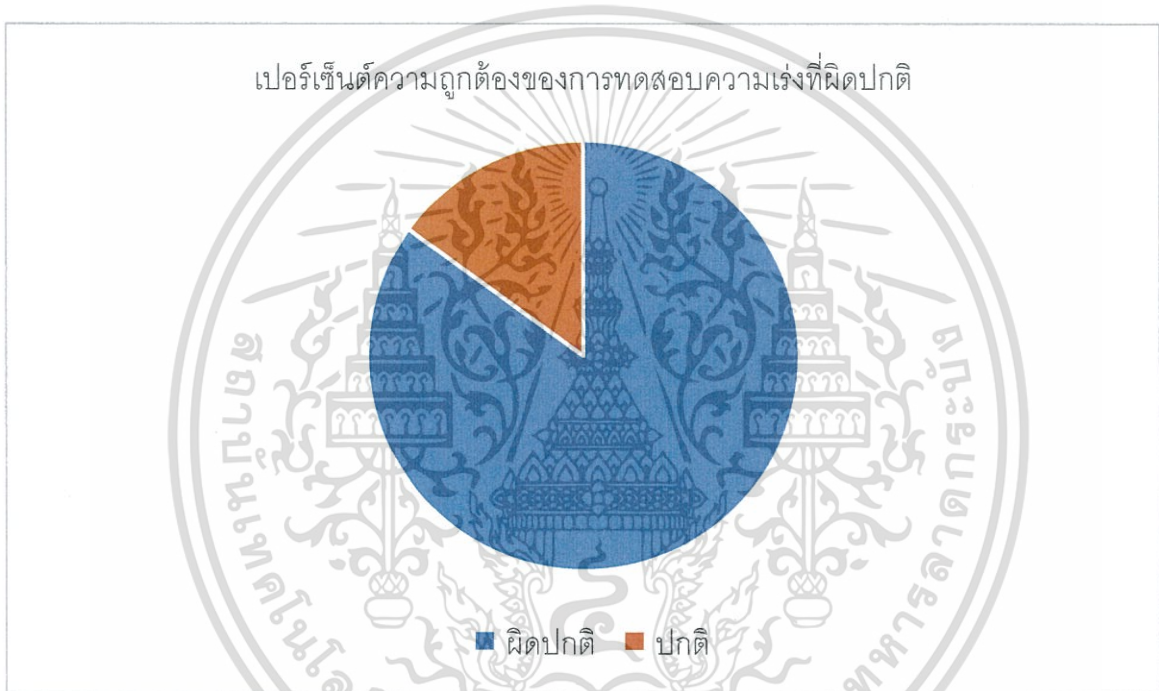
รูปที่ 4.8 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบการเคลื่อนไหวปกติ 20 ครั้ง

4.3.3. การทดสอบความเร่งที่ผิดปกติ

เป็นการทดสอบทำโดยการกระชากหมวกด้วยความเร่งที่สูงเพื่อสร้างความเร่งที่ผิดปกติให้กับหมวกนิรภัย

ตารางที่ 4.13 ผลการทดสอบความเร่งที่ผิดปกติ

ทั้งหมด	ผิดปกติ	ปกติ	ถูกต้อง(%)
20	17	3	85



รูปที่ 4.8 กราฟเปอร์เซ็นต์ความถูกต้องของการทดสอบความเร่งที่ผิดปกติ 20 ครั้ง

4.3.4. ทดสอบความแม่นยำในการตรวจวัดอัตราการเต้นหัวใจ

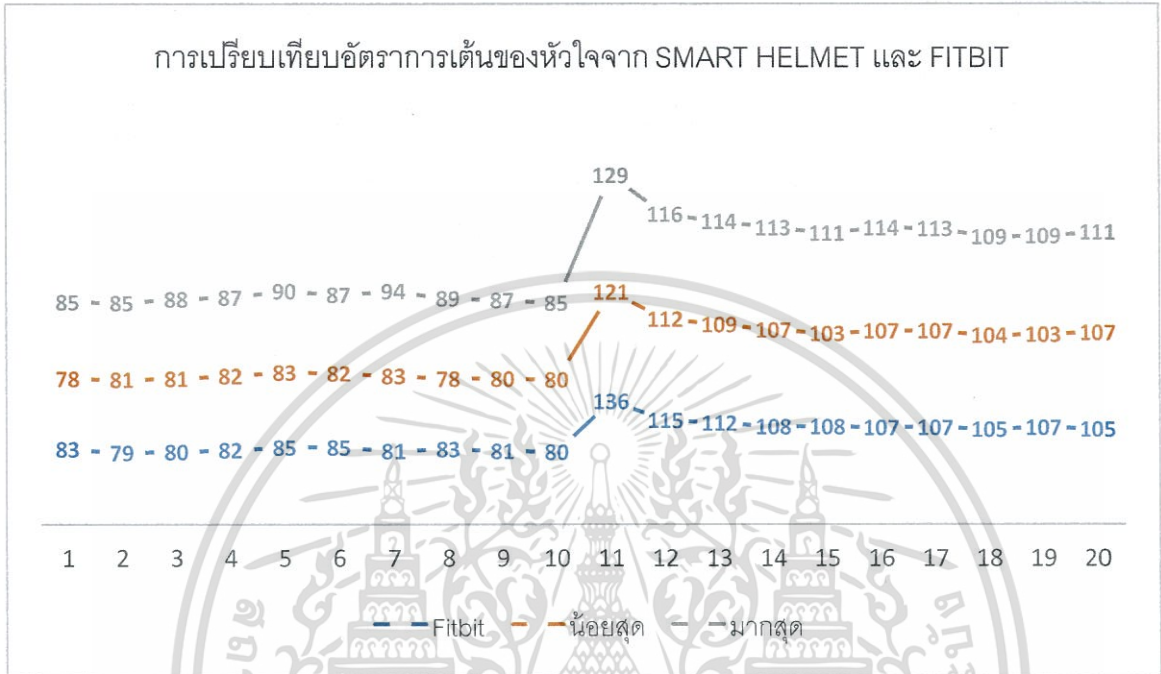
ทดสอบว่าระบบสามารถตรวจวัดอัตราการเต้นหัวใจของผู้ขับขี่ได้แม่นยำแค่ไหน การทดสอบทำโดยการให้ผู้ทดลองใส่ Fitbit ซึ่งเป็นอุปกรณ์ wearable ที่นิยมใช้เพื่อวัดผลการออกกำลังกาย เช่น นับก้าว วัดอัตราการเต้นหัวใจ เป็นต้น โดยในการทดสอบครั้งนี้จะใช้แค่ความสามารถในการวัดอัตราการเต้นหัวใจของ Fitbit ซึ่งการทดสอบได้ถูกทำทั้งหมด 20 ครั้ง 10 ครั้งแรกเป็นการวัดอัตราการเต้นหัวใจในสภาพที่ผู้ทดลองอยู่ในช่วงปกติ (พัก) และ 10 ครั้งสุดท้ายเป็นการวัดอัตราการเต้นหัวใจในสภาพหลังจากออกกำลังกายของผู้ทดลอง เนื่องจากค่าอัตราการเต้นหัวใจที่ได้จาก Fitbit นั้นมีการเปลี่ยนแปลงที่ช้ามาก การเปรียบเทียบจึงทำดังนี้ ขึ้นแรกอ่านค่าอัตราการเต้นหัวใจจาก Fitbit จากนั้นจับเวลา 15 วินาทีและดูว่าภายใน 15 วินาทีนี้ค่าอัตราการเต้นหัวใจที่ได้จากหวมกนิรภยมีค่าต่ำสุดและมากที่สุดเท่าไร



ตารางที่ 4.14 ผลการทดสอบความแม่นยำในการตรวจวัดอัตราการเต้นหัวใจ

ทั้งหมด	Fitbit	น้อยสุด	มากที่สุด
1	83	78	85
2	79	81	85
3	80	81	88
4	82	82	87
5	85	83	90
6	85	82	87
7	81	83	94
8	83	78	89
9	81	80	87
10	80	80	85
11	136	121	129
12	115	112	116
13	112	109	114
14	108	107	113
15	108	103	111
16	107	107	114
17	107	107	113
18	105	104	109
19	107	103	109
20	105	107	111

กราฟข้างล่างแสดงความสัมพันธ์ของการทดสอบทั้ง 20 ครั้งกับค่าอัตราการเต้นหัวใจที่วัดได้ในแต่ละครั้ง โดยประกอบไปด้วย สีน้ำเงิน สีเขียว และสีแดง ซึ่งหมายถึงค่าอัตราการเต้นหัวใจที่ได้จาก Fitbit, ค่าน้อยสุดจากหมวกนิรภัย และค่ามากที่สุดจากหมวกนิรภัย ตามลำดับ



รูปที่ 4.9 กราฟความสัมพันธ์ระหว่างอัตราการเต้นของหัวใจระหว่าง Smart Helmet และ Fitbit

บทที่ 5

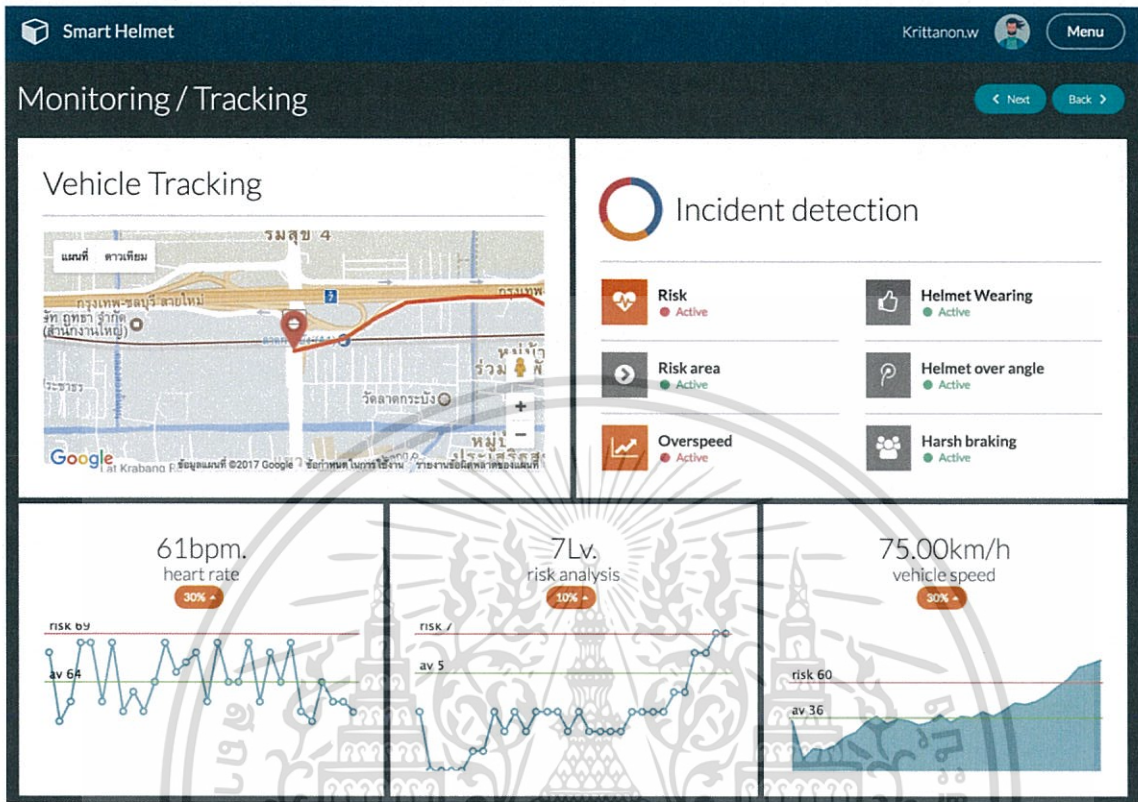
ผลการดำเนินงาน

5.1. หมวกนิรภัยอัจฉริยะที่พัฒนาแล้ว



รูปที่ 5.1 อุปกรณ์หมวกนิรภัยที่พัฒนา

5.2. Web Application



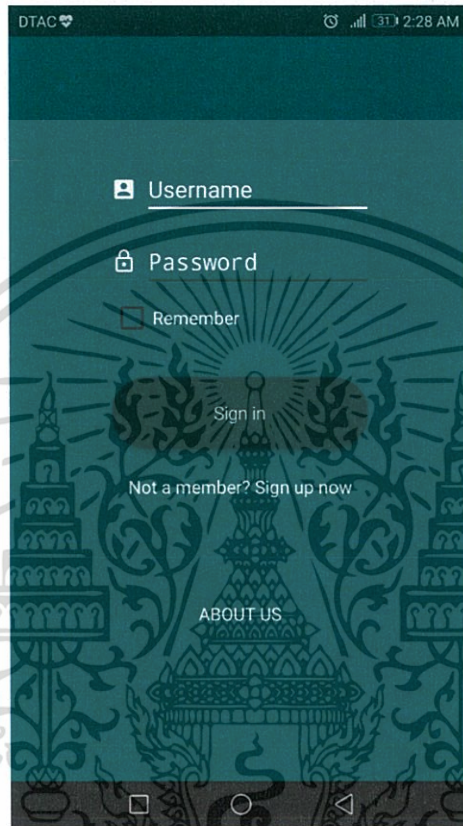
รูปที่ 5.2 หน้าจอ Web Application ที่พัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3. Mobile Application

5.3.1. การเข้าสู่ระบบ

ผู้ใช้งานจะต้องทำการกรอก Username และ Password ในการยืนยันตัวตนในการใช้งาน Application จากนั้นปุ่ม Sign In เพื่อทำการยืนยัน



รูปที่ 5.3 หน้าจอเข้าสู่ระบบ

5.3.2. การเชื่อมต่อระหว่างหมวกนิรภัยและ Application

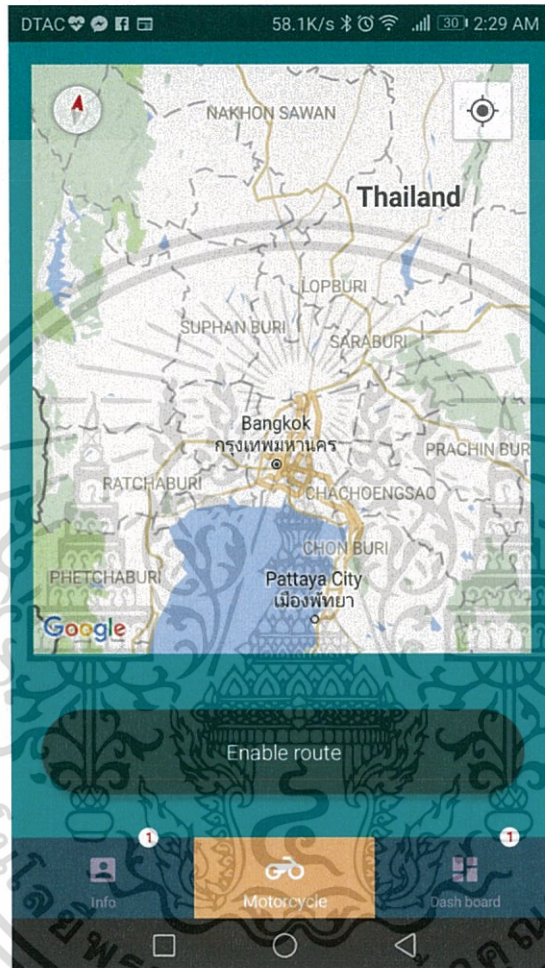
เมื่อเข้าสู่ระบบเรียบร้อยแล้วผู้ใช้งานจะต้องทำการเชื่อมต่อหมวกนิรภัยเข้ากับ Application โดยผ่าน Bluetooth ซึ่งเมื่อเข้าสู่หน้าจอนี้แล้วจะต้องทำการกดปุ่ม Scan จากนั้น Application จะทำการ List รายชื่ออุปกรณ์ที่สามารถเชื่อมต่อได้และหลังจากนั้นผู้ใช้งานจะต้องทำการเลือกอุปกรณ์ที่จะทำการเชื่อมต่อ



รูปที่ 5.4 หน้าจอเชื่อมต่ออุปกรณ์

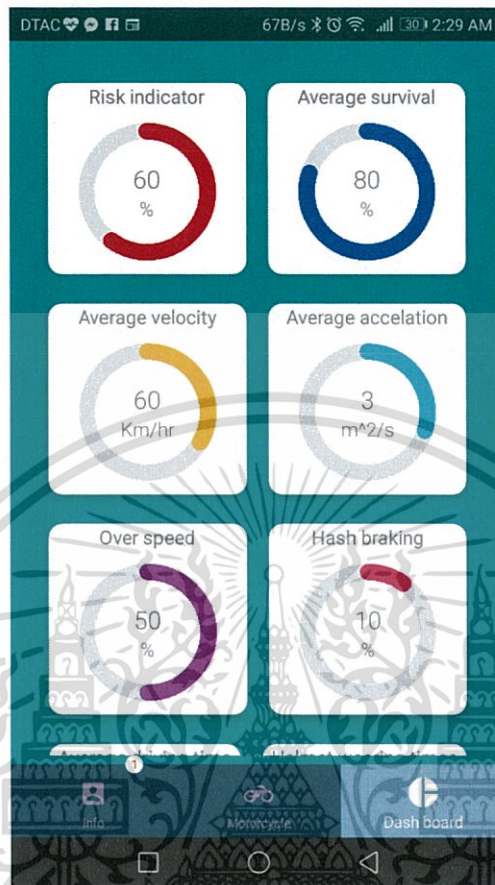
5.3.3. การแสดงผลประวัติการใช้งานการเดินทาง

เมื่อผู้ใช้งานทำการใช้งานอุปกรณ์เรียบร้อยแล้วระบบจะมีการเก็บข้อมูลการใช้งานต่าง ๆ และนำมาแสดงบน Application ของผู้ใช้งานเมื่อเข้าสู่หน้าจอแล้วจะต้องทำการกดปุ่ม Enable Route เพื่อทำการแสดงประวัติการใช้งานเส้นทาง



รูปที่ 5.5 หน้าแสดงบันทึกการเดินทางของผู้ใช้

5.3.4. ค่าสถิติพฤติกรรมของผู้ใช้งานหมวดนักเรียน



รูปที่ 5.6 หน้าแสดงพฤติกรรมการขับขี่ของผู้ใช้

บทที่ 6

สรุปและข้อเสนอแนะ

6.1. บทสรุป

หลังจากได้ทำการพัฒนาและทดสอบโครงการนี้ พบว่า ระบบทำงานได้ดีในการตรวจสอบว่าผู้ใช้ได้สวมใส่หมวกนิรภัยอยู่หรือไม่ ซึ่งเป็นการบังคับให้ผู้ขับขี่จำเป็นต้องสวมหมวกนิรภัยทุกครั้งก่อนการติดเครื่องยนต์ ในเรื่องการตรวจจับความผิดปกติที่เกิดขึ้นในระหว่างขับขี่ ยังทำได้ไม่แม่นยำเท่าไร ระบบสามารถแสดงข้อมูลสถานะทั้งหมดได้ทางจอมอนิเตอร์ซึ่งรับข้อมูลมาจาก Server

6.2. ปัญหา อุปสรรค และแนวทางการแก้ไข

- 1) Emulator ในการพัฒนา Mobile application ไม่สามารถรองรับการทำงานของ Google Map และ Bluetooth ได้

แนวทางการแก้ไข

จำเป็นต้องพัฒนาลงบนอุปกรณ์จริง จะช่วยแก้ปัญหาที่ไม่รองรับการทำงานได้แต่ทำให้การพัฒนาช้าลงไปเล็กน้อย

- 2) Library และ Method บางตัวถูก Deprecate หรือไม่รองรับการทำงานบนอุปกรณ์ที่ใช้พัฒนา

แนวทางการแก้ไข

บาง Library สามารถหาวิธีการต่าง ๆ มาแทนได้แต่บางตัวไม่สามารถใช้แทนกันได้ก็จำเป็นต้องลงมือบังคับการทำงานของ Method บางตัวให้สามารถทำงานได้ แต่การแสดงผลบางอย่างอาจไม่เกิดขึ้น แต่ต้องสามารถแน่ใจได้ว่าเมื่อลงในเครื่องที่ใหม่กว่าก็สามารถทำงานได้สมบูรณ์

- 3) ในการพัฒนาส่วนของรถจักรยานยนต์เมื่อทำการติดตั้งแล้วใช้เวลานานมากกว่าจะสามารถนำไปทดสอบได้

แนวทางการแก้ไข

ทำการทดสอบย่อย ๆ ที่ไม่เกี่ยวข้องกับภาพรวมก่อน เมื่อแน่ใจแล้วจึงทดสอบรวมจึงจะไม่เสียเวลาในการทดสอบมากนัก

6.3. ข้อเสนอแนะ

- 1) สามารถพัฒนาต่อได้ในกรณีที่หมวกนิรภัยกับรถจักรยานยนต์อยู่ห่างกันเกิน 50 เมตร ให้แจ้งเตือนผู้ใช้
- 2) พัฒนาต่อให้ระบบบนรถหนึ่งคันสามารถติดต่อกับระบบรถคันข้างเคียงได้ เพื่อการนำไปใช้สำหรับการประยุกต์ที่กว้างขึ้น
- 3) พัฒนาให้สามารถแจ้งข้อมูลจุดเสี่ยงด้วยเสียงผ่านหมวกนิรภัย พร้อมทั้งติดตั้งส่วนติดต่อกับโทรศัพท์เมื่อมีการติดต่อเข้ามาสามารถรู้ได้ในทันที



ภาคผนวก ก

1. การพัฒนาอัลกอริทึมของฟังก์ชันผลรวมความชัน (Slope-Sum Function)

1	static uint16_t slope_sum(struct queue *q, uint16_t x){
2	static uint32_t last_sum = 0;
3	uint16_t ssf, garbage;
4	int16_t lost, gain;
5	garbage = enqueue(q, x);
6	lost = q->tail->val - garbage;
7	lost = (lost < 0 ? 0 : lost);
8	gain = q->head->val - q->head->next->val;
9	gain = (gain < 0 ? 0 : gain);
10	ssf = last_sum - lost + gain;
11	last_sum = ssf;
12	return ssf;
13	}

- Struct Queue เป็นโครงสร้างข้อมูลที่เก็บข้อมูลในรูปแบบคิว โดยจะมีขนาดที่จำกัด นั่นคือ ทุกครั้งที่มีการเพิ่มข้อมูลเข้าไปที่หน้าสุดของคิว (Enqueue At Front) จะมีการดึงข้อมูลตัวท้ายสุดออกมาทุกครั้ง (Dequeue At Tail) เพื่อให้ขนาดของคิวเท่าเดิมตลอด
- ตัวแปร Last_Sum เป็น Persistent Variable
- ฟังก์ชันนี้ทำงานได้ปกติ ภายใต้สมมุติฐานที่ว่า ในการสร้างคิวครั้งแรก คิวจะไม่มีช่องว่างเลย และข้อมูลที่อยู่ข้างในทั้งหมดมีค่าเป็น 0
- เพราะฉะนั้น ในการหาผลรวมของค่าที่เข้ามาใหม่จึงทำได้โดย นำผลรวมที่คำนวณได้จากการเรียกฟังก์ชันครั้งก่อน บวกกับค่าความชันใหม่ที่ได้จากข้อมูลตัวใหม่ ลบด้วยค่าความชันที่เสียไปจากการ Dequeuer

2. การพัฒนาอัลกอริทึมในการหาจุดยอดของคลื่น (Peak Detection)

1	static uint8_t is_peak(uint16_t x, uint16_t last_peak){
2	static uint16_t last_x = 0;
3	static uint32_t t_last_peak = 0;
4	static enum WAVE_STATE state = SLIDING;
5	uint8_t peak = 0;
6	uint16_t diff_x = x - last_x;
7	switch (state) {
8	case CLIMBING:
9	if (diff_x <= 0) {
10	state = SLIDING;
11	if (x > last_peak last_peak - x <= 20 _millis() - t_last_peak > 3000) {
12	peak = 1;
13	t_last_peak = get_millis();
14	}
15	}
16	break;
17	case SLIDING:
18	if (diff_x >= 0){
19	state = CLIMBING;
20	}
21	break;
22	default: break;
23	}

- ทำ Hill Climbing โดยใช้การเปลี่ยน State ไปมาระหว่าง CLIMBING ซึ่งหมายถึงกำลังอยู่ในช่วงขาขึ้น หรือก็คือความชันเป็นบวก และ SLIDING ซึ่งหมายถึงกำลังอยู่ในช่วงขาลง หรือก็คือความชันเป็นลบ
- ทุกครั้งที่เกิดการเปลี่ยน State จาก CLIMBING เป็น SLIDING จะถือว่ามีการเกิดจุดยอดขึ้นเนื่องจากมีโอกาสที่จุดยอดที่เกิดขึ้นจะเป็นเพียงแค่ความไม่ต่อเนื่องของข้อมูล หรือค่าสัญญาณรบกวนต่าง ๆ ดังนั้นจะยังไม่ถือว่าเป็นจุดยอดจริง

- ทุกครั้งที่ตรวจพบจุดยอด จะทำการบันทึกไว้ และหากตรวจพบจุดยอดอีกครั้งจะทำการตัดสินใจว่าเป็นจุดยอดจริงหรือไม่ โดยตัดสินใจจากเกณฑ์ที่ว่า
 - หากจุดยอดใหม่มีค่ามากกว่าจุดยอดเดิม หรือ
 - หากจุดยอดใหม่มีค่าน้อยกว่าจุดยอดเดิมไม่เกินค่าที่กำหนด ในที่นี้คือ 20 หรือ ไม่มีการพบจุดยอดมาเป็นเวลานานค่าเวลาที่กำหนดไว้ ในที่นี้คือ 3 วินาที
 - หากจุดยอดใหม่ที่พบตรงกับเกณฑ์ข้อใดข้อหนึ่ง จะถือว่าเป็นจุดยอดจริงของคลื่น

3. การพัฒนาอัลกอริธึมในการหาค่า Threshold

1	static uint16_t calculate_treshold(struct queue *q, uint16_t x){
2	static float treshold = 0;
3	if (is_peak(x, q->head->val)) {
4	treshold *= PEAK_SAMPLE_SIZE;
5	treshold += (int16_t)(x - enqueue(q, x));
6	treshold /= PEAK_SAMPLE_SIZE;
7	}
8	}

- ในฟังก์ชันนี้จะใช้ Struct Queue แบบเดียวกับฟังก์ชัน Slope_Sum โดยคิวจะมีหน้าที่เก็บค่าจุดยอดคลื่นที่ผ่าน ๆ มาไว้
- ทำการหาค่าเฉลี่ยของจุดยอดภายในคิว โดยการนำค่าเฉลี่ยที่ได้จากการเรียกฟังก์ชันครั้งล่าสุด คูณจำนวนขนาดของคิว บวกด้วยค่าจุดยอดคลื่นที่เข้ามาใหม่ จากนั้นลบได้ค่าจุดยอดที่ถูก Dequeue ออกไป
- หลังจากนั้นหาค่าร้อยละ 70 ของค่าเฉลี่ยที่ได้ โดยค่านี้จะเป็น Threshold นั้นเอง

4. การหาค่าอัตราการเต้นของหัวใจโดยใช้ค่า Threshold

1	static uint8_t calculate_bpm(uint16_t treshold, uint16_t x){
2	static enum WAVE_STATE state = SLIDING;
3	static uint32_t t_fst_peak = 0;
4	static uint8_t count = 0, bpm = 0;
5	switch (state) {
6	case CLIMBING:
7	if (x < treshold) {
8	state = SLIDING;
9	if (count == 0) {
10	t_fst_peak = get_millis();
11	} else {
12	bpm = (uint8_t)(count / get_millis() - t_fst_peak) /
13	(1000 * 60.0));
14	count = 0;
15	break;
16	++count;
17	} break;
19	case SLIDING:
20	if (x > treshold){
21	state = CLIMBING;
22	} break;
23	default: break;
24	}
25	}

- ทำ Hill Climbing เช่นเดียวกับฟังก์ชัน is_peak แต่จะตัดสินใจเปลี่ยน State จากค่าสัญญาณเทียบกับค่า Threshold
- หากอยู่ใน CLIMBING State แล้วเปลี่ยนมาเป็น SLIDING State ครบสองครั้ง จะทำการหาค่าอัตราการเต้นของหัวใจ

5. การพัฒนาอัลกอริธึม Butterworth Filter

1	static uint16_t butterworth_lpf(uint16_t x){
2	static float v[2] = {0};
3	v[0] = v[1];
4	v[1] = (3.046874709125380054e-2 * x) + (0.93906250581749239892 * v[0]);
5	return (uint16_t)(v[0] + v[1]);
6	}

- จาก Code ในที่นี้ใช้ Butterworth Filter ที่มีออร์เดอร์ 1 และค่า Alpha1 เป็น 0.01
- ตัวแปร V เป็น Persistent Variable (Static ในบรรทัดที่ 2) ซึ่งหมายความว่า V จะถูกกำหนดค่าครั้งแรก (Initialize ในบรรทัดที่ 2) แค่ครั้งเดียว คือการเรียกฟังก์ชัน Butterworth_LPF ครั้งแรก โดยการเรียกฟังก์ชันครั้งต่อ ๆ ไป ค่า V จะยังคงเท่าเดิม และจะไม่มีการกำหนดค่าครั้งแรกอีก (Initialize)



เอกสารอ้างอิง

- [1] World Health Organization 2015. “Global Status Report On Road Safety 2015”
- [2] International Research Journal of Engineering and Technology (IRJET) 2017. “Smart Helmet”
- [3] e-Call. [Online] Available: <https://ec.europa.eu/digital-single-market/en/ecall-time-saved-lives-saved>
- [4] Dae-Geun Jang, Sangjun Park, and Minsoo Hahn 2014. “A RealTime Pulse Peak Detection Algorithm for the Photoplethysmogram”
- [5] oxullo 2016. “Arduino library for the Maxim Integrated MAX30100 oximetry / heart rate sensor.” [Online] Available : <https://github.com/oxullo/Arduino-MAX30100>
- [6] How to Use a Three-Axis Accelerometer for Tilt Sensing. [Online] Available: https://www.dfrobot.com/wiki/index.php/How_to_Use_a_Three-Axis_Accelerometer_for_Tilt_Sensing
- [7] Bluetooth SIG 2014, “Bluetooth Core Specification Version 4.2”
- [8] Pulse Sensor Amped – Technical Article. [Online] Available: <https://pulsesensor.com/pages/pulse-sensor-amped-arduino-v1dot1>
- [9] Photoplethysmogram. [Online] Available: <https://en.wikipedia.org/wiki/Photoplethysmogram>
- [10] Satellite navigation. [Online] Available: https://en.wikipedia.org/wiki/Satellite_navigation
- [11] Android Operating System. [Online] Available: [https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system))
- [12] BLE400 – NRF51822 Module. [Online] Available: <http://www.waveshare.com/wiki/BLE400>
- [13] Linkit Smart 7688 Duo. [Online] Available: http://wiki.seeed.cc/LinkIt_Smart_7688_Duo/
- [14] Lambda architecture. [Online] Available: <http://lambda-architecture.net>
- [15] nRF51822 Tutorial. [Online] Available: <https://devzone.nordicsemi.com/tutorials/>