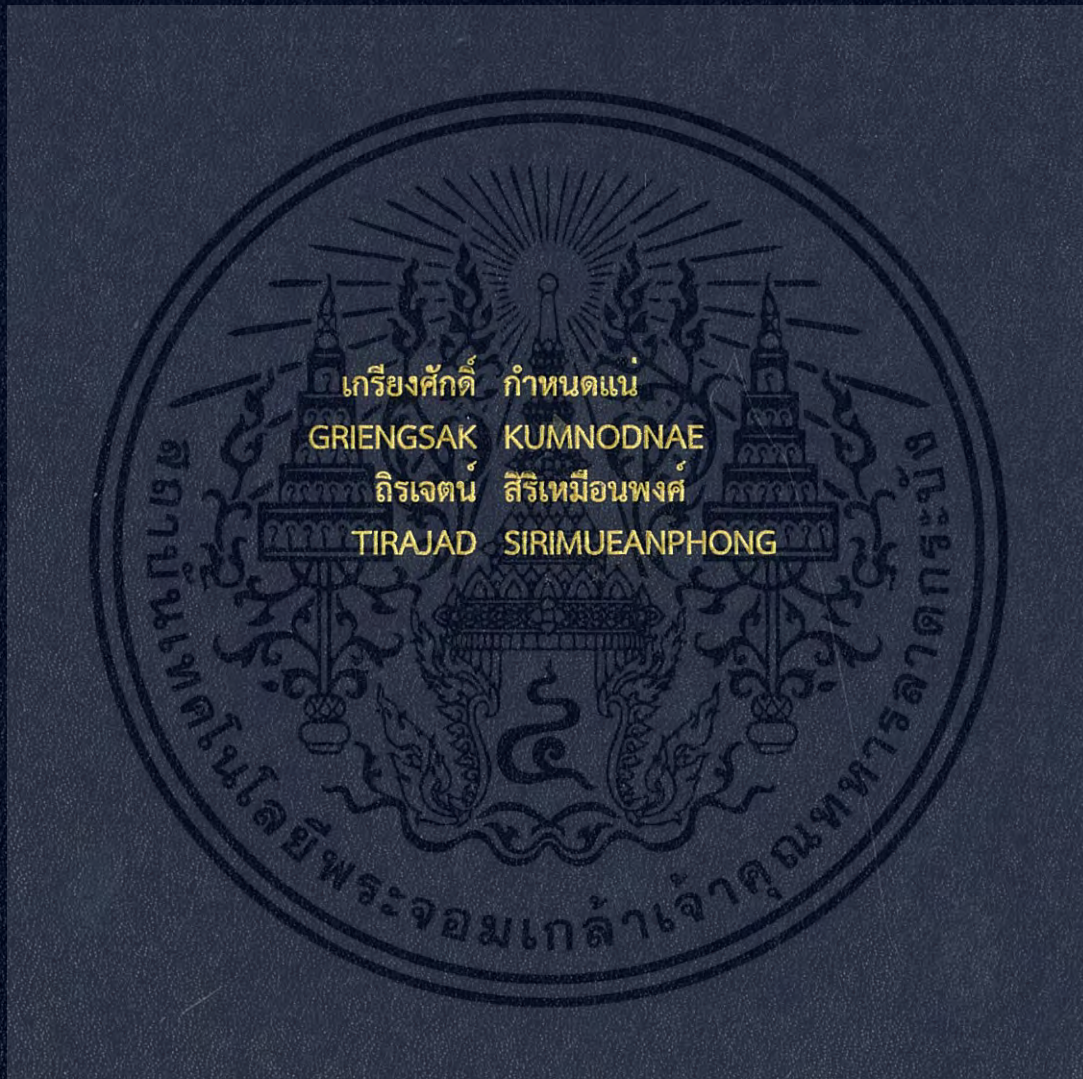


ระบบลงคะแนนเสียงแบบไม่มีศูนย์กลางบนระบบบล็อกเชน

DECENTRALIZED VOTING APPLICATION BASED ON  
BLOCKCHAIN SYSTEM



ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

ระบบลงคะแนนเสียงแบบไม่มีศูนย์กลางบนระบบบล็อกเชน

DECENTRALIZED VOTING APPLICATION BASED ON  
BLOCKCHAIN SYSTEM



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DECENTRALIZED VOTING APPLICATION BASED ON  
BLOCKCHAIN SYSTEM



THIS IS SUBMITTED IN PARTIAL FULFILLMENT  
OF THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR OF ENGINEERING IN INFORMATION ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG  
ACADEMIC YEAR 2017

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ใบรับรองปริญญาานิพนธ์

หัวข้อปริญญาานิพนธ์ ระบบลงคะแนนเสียงแบบไม่มีศูนย์กลางบนระบบ  
บล็อกเชน  
Thesis Title DECENTRALIZED VOTING APPLICATION BASED ON  
BLOCKCHAIN SYSTEM  
ชื่อนักศึกษา นายเกรียงศักดิ์ กำหนดแน่  
นายถิรเจตน์ สิริเหมือนพงศ์  
ระดับปริญญา วิศวกรรมศาสตรบัณฑิต  
สาขาวิชา วิศวกรรมสารสนเทศ  
ปริญญาานิพนธ์ปีการศึกษา 2560

(..... กฤตกร กล่อมการ .....)  
ผศ.ดร.กฤตกร กล่อมการ  
อาจารย์ที่ปรึกษาปริญญาานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	ระบบลงคะแนนเสียงแบบไม่มีศูนย์กลางบนระบบบล็อกเชน		
Thesis Title	DECENTRALIZED VOTING APPLICATION BASED ON BLOCKCHAIN SYSTEM		
ชื่อนักศึกษา	นายเกรียงศักดิ์ กำหนดแน่	รหัสนักศึกษา	57010116
	นายถิรเจตน์ สิริเหมือนพงศ์	รหัสนักศึกษา	57010497
ระดับปริญญา	วิศวกรรมศาสตรบัณฑิต		
สาขาวิชา	วิศวกรรมสารสนเทศ		
ปีการศึกษา	2560		
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผศ.ดร.กฤตากร กล่อมการ		

## บทคัดย่อ

วิทยานิพนธ์นี้มีจุดประสงค์เพื่อศึกษาและพัฒนาระบบจาก ระบบแบบที่มีศูนย์กลางในการจัดการ (Centralization) เป็นแบบกระจายอำนาจแบบไม่มีศูนย์กลางในการจัดการ (Decentralization) โดยมีการนำระบบบล็อกเชนมาใช้แทนระบบแทนระบบดาต้าเบส โดยใช้ระบบบล็อกเชนแบบบล็อกเชนส่วนตัวที่สร้างและจัดการด้วย Go-Ethereum ในการเชื่อมต่อ จัดการข้อมูล และประมวลผลบนระบบบล็อกเชน อีเทอเรียม แพลตฟอร์ม โดยมีการนำบล็อกเชนส่วนตัวที่สร้างนั้นมาสร้างแอปพลิเคชันการโหวตแบบไม่มีศูนย์กลางในการจัดการ โดยที่การนำระบบบล็อกเชนมาใช้นั้นจะทำให้มีความน่าเชื่อถือและปลอดภัยมากขึ้น เนื่องจากสิ่งที่อยู่บนบล็อกเชนไม่สามารถเปลี่ยนแปลงแก้ไขและสามารถตรวจสอบได้ โดยระบบการโหวตของแอปพลิเคชันแบบไม่มีศูนย์กลางนั้นผู้ใช้สามารถโหวตผู้เข้าแข่งขันได้ตามจำนวนเงินของผู้ใช้ ซึ่งสามารถโหวตเท่าไรก็ได้ถ้าหากจำนวนคะแนนส่วนกลางและเวลายังไม่หมด และในอนาคตสามารถนำแอปพลิเคชันพัฒนาและต่อยอดให้เป็นระบบเป็นแบบประชาธิปไตยหนึ่งคนหนึ่งโหวตเพื่อใช้ในการเลือกตั้งในอนาคต

<b>Thesis Title</b>	DECENTRALIZED VOTING APPLICATION BASED ON BLOCKCHAIN SYSTEM		
<b>Student</b>	Mr.Griengsak Kumnodnae	Student ID.	57010116
	Mr.Tirajad Sirimueanphong	Student ID.	57010497
<b>Degree</b>	Bachelor of Engineering		
<b>Program</b>	Information Engineering		
<b>Academic Year</b>	2017		
<b>Thesis Advisor</b>	Asst.Prof.Dr.Kitdakorn Klomkarn		

## ABSTRACT

This thesis is aim to study and develop the system from centralized system to be the decentralized by utilizing blockchain technology instead of using database system, by using private blockchain created and managed by Go-Ethereum for creating connection, managing data and processing on Ethereum Blockchain Platform. We are using the created private blockchain to build decentralized voting application. The blockchain system is make the system more reliable and safer because the content on the blockchain system is uneditable and traceable. The decentralized voting system of application make the user can vote the contestant as many times as the fund of user, depends on the central point and the time limit. In the near future, this application could be developed and furthered to be the ballot for democracy election system.

## กิตติกรรมประกาศ

ปริญญาโทเล่มนี้สำเร็จได้ด้วยความรู้มาจากอาจารย์ที่ ผศ. กฤดากร กล่อมการ ที่ให้ความช่วยเหลือ ให้คำชี้แนะช่วยแก้ปัญหาตลอดจนให้ความรู้และประสบการณ์ที่ดีแก่ผู้จัดทำ ทางผู้จัดทำขอขอบพระคุณที่ได้กรุณาให้คำปรึกษาแนะนำ และตรวจสอบ แก้ไข ข้อบกพร่องทุกขั้นตอนของการจัดทำโครงการ คณะผู้จัดทำโครงการขอขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ บิดา มารดา เพื่อนนักศึกษา ตลอดจนผู้ที่เกี่ยวข้องทุกท่านที่ไม่ได้กล่าวนามไว้ ณ ที่นี้ ที่ได้ให้กำลังใจและมีส่วนช่วยเหลือให้โครงการฉบับนี้สำเร็จลุล่วงได้ด้วยดี

ท้ายที่สุด คณะผู้จัดทำโครงการหวังว่าโครงการฉบับนี้จะเป็นประโยชน์กับผู้สนใจไม่มากก็น้อย

เกรียงศักดิ์ กำหนดแน  
ถิรเจตน์ สิริเหมือนพงศ์

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารตาราง.....	VIII
สารบัญรูป.....	IX
บทที่ 1 บทนำ.....	1
1.1 แนวคิดและที่มาของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 แนวคิดที่ใช้ในการออกแบบ.....	2
1.4 ขอบเขตของโครงการ.....	2
1.5 ขั้นตอนการท ำโครงการ.....	2
1.6 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีและเครื่องมือที่เกี่ยวข้อง.....	3
2.1 Blockchain.....	3
2.2 Ethereum.....	4
2.3 Go -ethereum.....	7
2.3.1 Geth console .....	8
2.4 Smartcontract.....	8
2.4.1 Solidity .....	8
2.4.2 JSON .....	9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และทำอย่างอื่นถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.5 Web3.js .....	10
2.6 Node.js .....	10
2.7 Node Package manager .....	11
2.8 เว็บแอปพลิเคชัน .....	11
2.8.1 HTML .....	11
2.8.2 HTTP (Hypertext Transfer Protocol) .....	12
2.8.2 CSS .....	12
2.8.4 Javascript .....	13
2.9 Truffle .....	14
2.10 Metamask .....	14
2.11 Sublime text .....	15
บทที่ 3 การออกแบบระบบและการดำเนินงาน .....	16
3.1 การออกแบบโครงสร้างของระบบการลงคะแนนเสียงแบบไม่มีศูนย์กลาง .....	16
3.1.1 ระบบลงคะแนนเสียง (Vote) .....	17
3.1.2 ระบบซื้อขายโทเคน (Token sale) .....	17
3.1.3 ระบบตรวจสอบคะแนนเสียง (Look up) .....	17
3.1.4 ระบบตรวจสอบข้อมูลในบล็อกเชน (Check Transaction) .....	17
3.2 Use Case Diagram .....	18
3.2.1 ระบบการสร้างผู้สมัครลงคะแนนเสียง .....	18
3.2.2 ระบบการซื้อขายโทเคน .....	19
3.2.3 ระบบการลงคะแนนเสียง .....	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

หน้า

3.2.4 ระบบตรวจสอบคะแนนเสียง .....	21
3.2.5 ระบบตรวจสอบข้อมูลในบล็อกเชน .....	22
3.3 Sequence Diagram .....	23
3.4 ภาพรวมขั้นตอนการดำเนินงานการสร้างของระบบ .....	24
3.5 ขั้นตอนการดำเนินงานสร้างบล็อกเชนส่วนตัว .....	25
3.5.1 ขั้นตอนการติดตั้ง brew .....	25
3.5.2 ขั้นตอนการติดตั้ง Go – Ethereum .....	25
3.5.3 ขั้นตอนการสร้างไฟล์เตอร์เพื่อจัดเก็บข้อมูลและกระเป๋าของบล็อกเชนส่วนตัว .....	26
3.5.4 ขั้นตอนการสร้างเจเนซิสบล็อก (genesisblock) .....	27
3.5.5 ขั้นตอนการดำเนินงานสร้างบล็อกเชนส่วนตัว .....	27
3.5.6 ขั้นตอนการสร้างแอคเคาท์ .....	28
3.5.7 ขั้นตอนการตั้งค่าการทางานของโหนดอีเทอเรียมบล็อกเชน .....	29
3.5.8 ขั้นตอนการสร้างไฟล์ที่ใช้ในการเปิดใช้งานโหนดอีเทอเรียมบล็อกเชน .....	30
3.5.9 ขั้นตอนการเปิดการทางานของโหนดอีเทอเรียมบล็อกเชน .....	30
3.5.10 ขั้นตอนการเชื่อมต่ออีเทอเรียมบล็อกเชน .....	31
3.5.11 ขั้นตอนการใช้คำสั่งประมวลผลโหนดอีเทอเรียมบล็อกเชน .....	32
3.6 ขั้นตอนการดำเนินการแอปพลิเคชันแบบไม่มีศูนย์กลาง .....	32
3.6.1 ขั้นตอนการติดตั้ง truffle .....	32
3.6.2 ขั้นตอนการสร้างไฟล์เตอร์ส สำหรับจัดการแอปพลิเคชันแบบไม่มีศูนย์กลาง ..	33
3.6.3 ขั้นตอนการตั้งค่า truf .....	34
3.6.4 ขั้นตอนการติดตั้งสัญญาอัจฉริยะลงในบล็อกเชนส่วนตัว .....	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ VI อย่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
บทที่ 4 การทำงานของเว็บแอปพลิเคชัน .....	36
4.1 การเข้าสู่หน้าหลัก .....	36
4.2 การเข้าสู่หน้าเมนู Tokensale .....	37
4.3 การเข้าสู่หน้าเมนู Home ลงคะแนนเสียง .....	39
4.4 การเข้าสู่หน้าเมนู Look up .....	41
4.5 การเข้าสู่หน้าเมนู Check transaction .....	42
4.5.1 เมนู Home ของ Check transaction .....	42
4.5.2 เมนู Blockchain Info ของ Check transaction .....	45
บทที่ 5 สรุปผลการดำเนินงาน .....	46
5.1 สรุปผลการทดลอง .....	46
5.2 ปัญหาและอุปสรรคของการดำเนินงาน .....	46
5.2.1 แหล่งที่มาอ้างอิงข้อมูลยังไม่เพียงพอ .....	46
5.2.2 รหัสเบอริ์พายไม่สามารถท ากาต านวณประมวลผลได้ .....	46
5.3 แนวทางการพัฒนาโครงการ .....	47
5.3.1 น าไปพัฒนาต่อบนบล็อกเชนหลัก .....	47
5.3.2 เพิ่มการใช้งานรูปแบบการลงคะแนนเสียง .....	47
เอกสารอ้างอิง .....	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ VII อ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง

หน้า

ตารางที่ 3.1 แสดงรายละเอียด Use Case ของระบบสร้างผู้สมัครลงคะแนนเสียง .....	18
ตารางที่ 3.2 แสดงรายละเอียด Use Case ของระบบการซื้อขายโทเคน .....	19
ตารางที่ 3.3 แสดงรายละเอียด Use Case ของระบบการลงคะแนนเสียง.....	20
ตารางที่ 3.4 แสดงรายละเอียด Use Case ของระบบตรวจสอบคะแนนเสียง .....	21
ตารางที่ 3.5 แสดงรายละเอียด Use Case ของระบบตรวจสอบข้อมูลในบล็อกเชน.....	22



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ VIII อ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

หน้า

รูปที่ 2.1 ความแตกต่างระหว่าง centralized , Decentralized , Distributed .....	3
รูปที่ 2.2 แสดงการเชื่อมต่อของEthereum Blockchain Network โดยใช้ Geth protocol .....	7
รูปที่ 2.3 ตัวอย่างภาษา JSON .....	9
รูปที่ 2.4 แสดงการเชื่อมต่อของโปรแกรมที่ใช้ไปยัง Geth protocol .....	10
รูปที่ 2.5 ภาพแสดงสัญลักษณ์ของ node.js .....	11
รูปที่ 2.6 ภาพแสดงสัญลักษณ์ของ Node package manager .....	11
รูปที่ 2.7 ภาพแสดงสัญลักษณ์ของ TRUFFLE .....	14
รูปที่ 2.8 ภาพแสดงสัญลักษณ์ของ Metamask .....	15
รูปที่ 2.9 ภาพแสดงสัญลักษณ์ของ Sublime Text .....	15
รูปที่ 3.1 แสดงภาพรวมการออกแบบของระบบ.....	16
รูปที่ 3.2 ระบบสร้างผู้สมัครลงคะแนนเสียง .....	18
รูปที่ 3.3 ระบบการซื้อขายโทเคน.....	19
รูปที่ 3.4 ระบบการลงคะแนนเสียง .....	20
รูปที่ 3.5 ระบบตรวจสอบคะแนนเสียง.....	21
รูปที่ 3.6 ระบบตรวจสอบข้อมูลในบล็อกเชน .....	22
รูปที่ 3.7 Sequence Diagram ของระบบ.....	23
รูปที่ 3.8 แสดงขั้นตอนการสร้างบล็อกเชนส่วนตัว.....	24
รูปที่ 3.9 ขั้นตอนการจัดการเว็บแอปพลิเคชัน .....	25
รูปที่ 3.10 ขั้นตอนการติดตั้ง brew .....	25
รูปที่ 3.11 ขั้นตอนการโหลด Go-ethereum .....	26
รูปที่ 3.12 ขั้นตอนการติดตั้ง Go-ethereum .....	26
รูปที่ 3.13 ขั้นตอนการสร้างโพลเดอร์.....	26
รูปที่ 3.14 แสดงไฟล์ genesis.json .....	27
รูปที่ 3.15 แสดงคำสั่งย้ายโพลเดอร์.....	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ IX อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ 3.16 แสดงค าสั่ง สร้างบล็อกเชนส่วนตัว.....	28
รูปที่ 3.17 แสดงโพลเดอร์ที่จากการใช้ค าสั่งในรูป 3.9 .....	28
รูปที่ 3.18 แสดงค าสั่งสร้างแอคเคาท์ .....	28
รูปที่ 3.19 แสดงไฟล์password.sec .....	29
รูปที่ 3.20 แสดงค าสั่งที่ใช้ในการท งานโหนด.....	29
รูปที่ 3.21 แสดงไฟล์start.sh และ ค าสั่งภายในไฟล์.....	30
รูปที่ 3.22 แสดงการท างานของโหนดอีเทอเรียมบล็อกเชนที่เปิดการใช้งานแล้ว .....	30
รูปที่ 3.23 แสดงค าสั่งแสดงไอพี.....	31
รูปที่ 3.24 แสดงค าสั่งแสดงเลขประจำ ำตัวโหนด .....	31
รูปที่ 3.25 แสดงไฟล์ที่ใส่เลขประจำ ำตัวโหนดและไอพี .....	31
รูปที่ 3.26 แสดงการท างานของโหนดอีเทอเรียมบล็อกเชนที่เริ่มการประมวลผลทรานแซคชัน .....	32
รูปที่ 3.27 แสดงการติดตั้ง t .....	33
รูปที่ 3.28 แสดงค าสั่งย้ายไปโพลเดอร์ลงคะแนน .....	33
รูปที่ 3.29 แสดงค าสั่งสร้าง tr ึ่งในโพลเดอร์.....	33
รูปที่ 3.30 แสดงโพลเดอร์ติดตั้ง tr .....	34
รูปที่ 3.31 แสดงไฟล์ tr ึ่ง js .....	34
รูปที่ 3.32 แสดงค าสั่งtr ึ่ง compile .....	35
รูปที่ 3.33 แสดงค าสั่งtr ึ่ง migrate .....	35
รูปที่ 4.1 หน้าแรกของเว็บแอปพลิเคชัน.....	36
รูปที่ 4.2 หน้าเมนู Token sale .....	37
รูปที่ 4.3 ขั้นตอนการซื้อโทเค็น จ ำนวน 300 โทเค็น .....	37
รูปที่ 4.4 ยืนยันการซื้อโทเค็นผ่าน Metamask .....	38
รูปที่ 4.5 แสดงหน้าที่มีการซื้อโทเค็นเพิ่มไปจ ำนวน 300 เค็น.....	38
รูปที่ 4.6 แสดงหน้า Home ส่วนการลงคะแนนเสียง .....	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ X อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ 4.7 ลงคะแนนเสียงให้ผู้สมัคร Jay จำนวน 100 โทเค้น .....	39
รูปที่ 4.8 ยืนยันการลงคะแนนผ่าน Metamask .....	40
รูปที่ 4.9 แสดงผลคะแนนของ Jay เพิ่มขึ้น 100 คะแนน .....	40
รูปที่ 4.10 แสดงหน้า Look up .....	41
รูปที่ 4.11 แสดงผล Look up .....	41
รูปที่ 4.12 แสดงหน้า Home ในส่วนของ Check transection .....	42
รูปที่ 4.13 แสดงหน้า Home ในส่วนของ Check transection(ต่อ) .....	43
รูปที่ 4.14 แสดงผลการค้นหาด้วย Tx Hash .....	43
รูปที่ 4.15 แสดงผลการค้นหาด้วย Adress .....	44
รูปที่ 4.16 แสดงผลการค้นหาด้วย Block .....	44
รูปที่ 4.17 แสดงหน้า Blockchain Info ในส่วนของ Chech transection .....	45
รูปที่ ก.1 Poster Decentralized Voting Appilcation Based On Blockchain System .....	51
รูปที่ ข.1 แสดงหน้าเว็บไซต์ Homebrew .....	53
รูปที่ ข.2 แสดงหน้าเว็บไซต์ github ของ Go-ethereum .....	54
รูปที่ ข.3 แสดงหน้าTerminal ติดตั้ง Go-ethereum .....	54
รูปที่ ค.1 แสดงหน้าเว็บ Node.js .....	56
รูปที่ ค.2 หน้าต่างอธิบาย Node.js ที่ติดตั้ง.....	56
รูปที่ ค.3 หน้าต่างเงื่อนไข และการยอมรับการใช้งานของ Node.js .....	57
รูปที่ ค.4 กดยอมรับเงื่อนไขของ Node.js .....	57
รูปที่ ค.5 เลือกไดรฟ์ในการเก็บข้อมูล .....	58
รูปที่ ค.6 ยืนยันการเลือกไดรฟ์ .....	58
รูปที่ ค.7 รอการติดตั้ง Node.js .....	59
รูปที่ ค.8 แสดงการติดตั้ง Node.js เสร็จสมบูรณ์.....	59
รูปที่ ง.1 แสดงหน้าเว็บไซต์ T Framework .....	61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ XI อ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ ง.2 แสดงคำสั่งและการติดตั้ง T Framework .....	61
รูปที่ จ.1 เว็บไซต์ตัวอย่างไฟล์ Web3.JS .....	63
รูปที่ จ.2 แสดงคำสั่ง npm install web3 .....	63
รูปที่ จ.3 แสดงการติดตั้ง Web3.JS เสร็จสมบูรณ์ .....	64
รูปที่ จ.4 แสดงโพลเดอร์ ที่ติดตั้ง Web3.JS .....	64
รูปที่ ฉ.1 เว็บไซต์ของโปรแกรม Sublime Text .....	66
รูปที่ ฉ.2 แสดงการลาก Sublime Text ลงในโพลเดอร์ Application .....	66
รูปที่ ฉ.3 แสดงการติดตั้ง Sublime Text เสร็จสมบูรณ์ .....	67
รูปที่ ช.1 เว็บไซต์ของส่วนขยายเบราว์เซอร์ Metamask .....	69
รูปที่ ช.2 หน้าติดตั้งของส่วนขยายเบราว์เซอร์ Metamask .....	69
รูปที่ ช.3 หน้ายืนยันการติดตั้งของส่วนขยายเบราว์เซอร์ Metamask .....	70
รูปที่ ช.4 หน้าแสดงการติดตั้ง Metamask เสร็จสมบูรณ์ .....	70
รูปที่ ช.5 หน้าแสดงการ Metamask เข้ากับบล็อกเชนส่วนตัว .....	71
รูปที่ ช.6 หน้าแสดงการแก้ไขพอร์ตให้ตรงกับบล็อกเชนส่วนตัว .....	71
รูปที่ ช.7 หน้าแสดงการเชื่อมต่อ Metamask กับบล็อกเชนส่วนตัวเสร็จสมบูรณ์ .....	72

# บทที่ 1

## บทนำ

### 1.1 แนวคิดและที่มาของโครงการ

บล็อกเชน (Blockchain) เป็นเทคโนโลยีใหม่ที่กำลังพูดถึงในโลกปัจจุบันเป็นอย่างมาก ด้วยเทคโนโลยี ทำให้ผู้คนสามารถเขียน บันทึกเรื่องราว หรือเก็บข้อมูล (Database) ในรูปแบบหนึ่งของระบบที่ไม่มีศูนย์กลาง แต่ด้วยระบบของ บล็อกเชน นั้นถูกพัฒนาออกมา ให้เชื่อถือได้ และทำให้ปลอมหรือปลอมแปลงข้อมูลต่างๆที่อยู่ในระบบนั้นๆ แบบเป็นไปได้ยาก โดยแนวคิดของบล็อกเชนนั้นคือการให้ทุกคนถือเอกสารชุดเดียวกัน เมื่อมีการอัปเดตก็จะอัปเดตด้วยกันเสมอ โดยมั่นใจได้ว่าเอกสารเหล่านั้นเชื่อถือได้แน่นอนไม่มีการปลอมแปลง

ในปัจจุบันนั้นได้มีการนำระบบบล็อกเชน มาใช้เป็นระบบฐานข้อมูลของแอปพลิเคชันแบบไม่มีศูนย์กลาง หรือที่เรียกว่า DApp ซึ่งย่อมาจาก Decentralized Application โดยสามารถพัฒนาออกไปได้หลากหลายรูปแบบ โดยใจกลางของแอปพลิเคชันนั้น มีโปรแกรมที่เชื่อมเพื่อติดต่อระหว่างหน้าบ้าน และหลังบ้าน เพื่อรับคำสั่งและประมวลผล หรือที่เรียกว่า สัญญาอัจฉริยะ (Smart contract) โดยระบบ แอปพลิเคชันที่ได้นำระบบบล็อกเชนมาใช้เป็นฐานข้อมูลนั้น จะได้รับความปลอดภัย และความน่าเชื่อถือมา

### 1.2 วัตถุประสงค์ของโครงการ

1. ศึกษาและค้นคว้าการทำงานของระบบบล็อกเชน
2. ต้องการทราบถึงกระบวนการทั้งหมดของการทำงานของบล็อกเชนในการสร้าง แอปพลิเคชันแบบไม่มีศูนย์กลาง (Decentralized Application)
3. ค้นคว้าและพัฒนาระบบบล็อกเชนและพัฒนาแอปพลิเคชันแบบไม่มีศูนย์กลาง
4. สร้างแอปพลิเคชันการลงคะแนนเสียงแบบไม่มีศูนย์กลางโดยใช้บล็อกเชนเป็นตัวจัดการและจัดเก็บข้อมูล
5. สามารถพัฒนาแอปพลิเคชันแบบไม่มีศูนย์กลาง เพื่อเป็นต้นแบบนำไปพัฒนา และเป็นแหล่งการเรียนรู้การสอน เพื่อประโยชน์ในการเรียนรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 1.3 แนวคิดที่ใช้ในการออกแบบ

1. สร้างระบบบล็อกเชนส่วนตัว (Private Blockchain) เพื่อใช้เป็นฐานข้อมูลระดับองค์กร
2. สร้างแอปพลิเคชันแบบไม่มีศูนย์กลางโดยใช้บล็อกเชนในการจัดการและเก็บข้อมูลเพื่อความปลอดภัย และโปร่งใสขึ้นมากกว่าระบบปัจจุบัน

### 1.4 ขอบเขตของโครงการ

แอปพลิเคชันลงคะแนนแบบไม่มีศูนย์กลาง (decentralized voting application) จะต้องใช้งานได้อย่างมีประสิทธิภาพโดยผ่านบล็อกเชนที่เป็นระบบฐานข้อมูล

### 1.5 ขั้นตอนการทำโครงการ

1. ศึกษาค้นคว้าข้อมูลบล็อกเชน, แอปพลิเคชันแบบไม่มีศูนย์กลาง, สัญญาอัจฉริยะ
2. สร้างบล็อกเชนส่วนตัว โดยใช้ Go-ethereum
3. ทดสอบระบบ บล็อกเชนส่วนตัว
4. สร้าง สัญญาอัจฉริยะ, แอปพลิเคชันการลงคะแนนเสียงแบบไม่มีศูนย์กลาง, หน้าเว็บไซต์งานแอปพลิเคชัน
5. ทดสอบการทำงานของบล็อกเชน, แอปพลิเคชันการลงคะแนนเสียงแบบไม่มีศูนย์กลาง หน้าเว็บไซต์งานแอปพลิเคชัน และ สัญญาอัจฉริยะ
6. ปรับปรุง แก้ไข การทำงานในส่วนต่าง ๆ

### 1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถวางแผนการพัฒนาระบบบล็อกเชน และแอปพลิเคชันได้อย่างเป็นขั้นตอน
2. สามารถพัฒนาบล็อกเชน และแอปพลิเคชัน
3. ได้รับความรู้เกี่ยวกับเทคโนโลยีต่าง ๆ ที่เลือกใช้ในการพัฒนาบล็อกเชนและแอปพลิเคชัน
4. ผู้ใช้งานระบบสามารถวางแผนการให้บริการเดินรถได้อย่างมีประสิทธิภาพ
5. เพื่อเป็นแหล่งการเรียนการสอน เพื่อให้นำไปศึกษา และเป็นแนวทางพัฒนาต่อยอดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

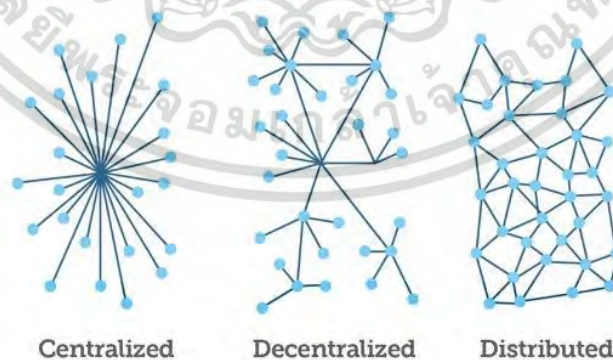
# ทฤษฎีและเครื่องมือที่เกี่ยวข้อง

ในการจัดสร้างโครงงาน Blockchain จำเป็นจะต้องศึกษาทฤษฎีที่เกี่ยวข้องดังนี้

### 2.1 Blockchain

บล็อกเชน (Blockchain) เปรียบเสมือน ฐานข้อมูล หรือ สมุดบัญชี ที่คอยควบคุมดูแลการ จัดเก็บ ทุกๆข้อมูลหรือทุกๆกิจกรรมของการติดต่อสื่อสาร (Transaction) ที่เติบโตขึ้นตลอดเวลา แต่ บล็อกเชน มี 5 จุดเด่นที่แตกต่าง จากระบบฐานข้อมูลทั่วไปคือ

1. แคร่ข้อมูลออกสู่สาธารณะ – เซิร์ฟเวอร์ หรือ โหนด ต่างๆจะมีหน้าที่ดูแลข้อมูล “แหล่งเก็บบันทึกข้อมูล” หรือที่เราเรียกว่า บล็อก และทุกๆ โหนด จะมีความสามารถที่ดูได้ว่า ข้อมูล ทรานเซคชัน นั้นๆถูกจัดเก็บใน “บล็อก” ไต (ข้อมูลจะมีการเข้ารหัส แฮช)
2. ไม่มีศูนย์กลาง (Decentralized) ในการ ตรวจสอบข้อมูล ทรานเซคชัน ใดๆ ที่เกิดขึ้น
3. มีความปลอดภัย (Secure) ข้อมูลที่จัดเก็บใน บล็อก จะไม่มีการเปลี่ยนแปลง, เอาค่าคืนกลับมาได้ หรือแม้กระทั่งทำลายข้อมูลนั้นก็ตาม
4. มีความน่าเชื่อถือ (Trusted) ด้วยธรรมชาติของการไม่มีศูนย์กลางจัดเก็บข้อมูล ดังนั้นระบบจึงต้องพึ่งความคิดเห็นของคนส่วนใหญ่ เมื่อมี ทรานเซคชัน เกิดขึ้นจากคนที่ไม่รู้จัก
5. ทำงานอัตโนมัติ (Automated) ซอฟต์แวร์ จะถูกเขียนขึ้นให้ป้องกันการเขียนข้อมูลซ้ำซ้อน และการจัดเก็บข้อมูล ทรานเซคชัน ก็ดำเนินการแบบอัตโนมัติเช่นกัน



รูปที่ 2.1 ความแตกต่างระหว่าง centralized , Decentralized , Distributed

(อ้างอิงโดย <https://www.linkedin.com/pulse/blockchain-distributed-ledger-technology-david-com>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกเชน มีหน้าที่หลักๆ 2 อย่างคือ ควบคุมข้อมูลและป้องกันการซ้ำข้อมูล ซึ่งใน 1 บล็อก จะประกอบด้วย 4 ส่วนหลักๆ คือ

1. ไอดี ประจำ บล็อก ซึ่งเป็นตัวเลขสุ่ม ที่เข้ารหัส แฮช ไว้
2. ไอดี ของ บล็อก ก่อนหน้านี้ ซึ่งเข้ารหัส แฮช ไว้เช่นกัน
3. ข้อมูล ทรานแซคชัน ซึ่งอาจจะมีเพียง 1 ทรานแซคชัน หรือมากกว่านั้นก็ได้
4. คีย์สาธารณะ ที่บอกว่า บล็อก นี้เป็นของใคร ใครส่งให้ใคร ใครเป็นผู้รับ

ทุก บล็อก จะถูกสร้างขึ้น ก็ต่อเมื่อมีหลายๆ โหนด ตกลงว่าจะยอมรับและมีการตรวจสอบว่า ทรานแซคชัน นั้นถูกต้อง กระบวนการนี้จึงถูกเรียกว่า “Distributed Ledger” ซึ่งเป็นกระบวนการที่ไม่ต้องใช้ การเก็บข้อมูลไว้ส่วนกลาง เพื่อใช้ในการตรวจสอบ จึงมีคำถามถัดมาว่าแล้วถ้า โหนด นั้นๆ เกิดปิดเครื่องขึ้นมาจะอย่างไร มันเลยมีคนสร้าง เซิร์ฟเวอร์ ไว้ให้กระบวนการนี้นั่นเอง หรือเรียกว่าการ ไม่มีการตรวจสอบความถูกต้อง บล็อก นั้นจะมีอยู่ด้วยกัน 2 ขั้นตอนคือ

1. PROOF OF WORK (POW) กระบวนการตรวจสอบว่า สมุดบัญชี หรือ บัญชีข้อมูลนี้ ไม่ได้ถูกแทรกเข้ามา ซึ่งกระบวนการตรวจสอบนี้จะต้องใช้ คอมพิวเตอร์ เข้ามาช่วย (เครื่องที่ ไม่นั่นเอง)
2. PROOF OF STAKE ในขณะที่ คอมพิวเตอร์ ดำเนินการตรวจสอบข้อมูลที่เข้ารหัส เจ้า Proof of stake ก็จะตรวจสอบความเป็นเจ้าของ ด้วยจำนวนที่มีการเพิ่มขึ้นอย่างแน่นอน

ภายในข้อมูลบัญชี (Ledger) นั้นจะไม่มีมีการเก็บข้อมูลเจ้าของ แต่จะเก็บเพียงข้อมูล ทรานแซคชัน ทั้งหมดที่ผ่านมาเท่านั้น ซึ่งในส่วนนี้ กระเป๋าตั้ง (Wallet) จะสามารถอ้างอิงไปยังเจ้าของผ่านทาง ข้อมูล Transaction ที่ผ่านมานั่นเอง

## 2.2 Ethereum

อีเทอเรียม เป็นแพลตฟอร์มแบบเปิดของ บล็อกเชน ที่ทำให้ทุกคนสามารถสร้างและใช้งาน แอปพลิเคชันแบบกระจายข้อมูล (decentralized) ซึ่งทำงานบนเทคโนโลยี บล็อกเชน ได้ อีเทอเรียม มีความคล้าย Bitcoin ตรงที่ไม่มีใครสามารถควบคุมหรือเป็นเจ้าของ อีเทอเรียม ได้ เนื่องจาก อีเทอเรียม เป็นโครงการแบบโอเพนซอร์ส (open-source project) ที่สร้างขึ้นโดยผู้คนเป็นจำนวนมาก จากทั่วโลก แต่ อีเทอเรียม มีความแตกต่างจากโปรโตคอล บิทคอยน์ เนื่องจาก อีเทอเรียม ถูกออกแบบมาเพื่อให้สามารถปรับตัวได้และมีความยืดหยุ่น การสร้างแอปพลิเคชันใหม่ ๆ บนแพลตฟอร์ม อีเทอเรียม เป็นเรื่องง่าย และการเปิดตัว ที่จัดเก็บ(Homestead) ก็ทำให้ทุกคนสามารถใช้งานแอปพลิเคชันเหล่านี้ได้อย่างปลอดภัย

อีเทอเรียม เป็น บล็อกเชน ที่เขียนชุดคำสั่งได้ ซึ่งแทนที่จะให้ผู้ใช้ชุดปฏิบัติการที่ถูกกำหนดไว้ล่วงหน้าแล้ว (pre-defined operations) (เช่น ธุรกิจเกี่ยวกับ บิทคอยน์) อีเทอเรียม กลับช่วยให้ผู้ใช้สามารถสร้างปฏิบัติการให้มีความซับซ้อนตามที่ผู้ใช้ต้องการได้ ด้วยวิธีนี้ อีเทอเรียม จะทำหน้าที่

เป็นแพลตฟอร์มสำหรับแอปพลิเคชัน บล็อกเชน ที่มีรูปแบบหลากหลาย ซึ่งสามารถใช้สกุลเงินดิจิทัลได้อย่างไม่จำกัด

อีเทอริยม ในความหมายอย่างแคบ (narrow sense) หมายถึงชุดของโปรโตคอลที่กำหนดแพลตฟอร์มสำหรับแอปพลิเคชันแบบกระจายข้อมูล หัวใจสำคัญของมันคือ Ethereum Virtual Machine (“EVM”) ซึ่งสามารถประมวลผลซึ่งมีอัลกอริทึมที่ซับซ้อนได้เอง ในแง่ของวิทยาการคอมพิวเตอร์ อีเทอริยม นั้นอยู่ในระดับ “Turing complete” (แก้ปัญหาได้เทียบเท่าเครื่องคอมพิวเตอร์) นักพัฒนาซอฟต์แวร์สามารถสร้างแอปพลิเคชันที่ทำงานบน EVM โดยใช้ภาษาโปรแกรมที่รองรับได้ซึ่งมีตัวแบบเป็นภาษาที่มีอยู่ เช่น JavaScript และ Python

อีเทอริยม ยังมีโปรโตคอลเครือข่ายแบบ ตัวต่อตัว(peer-to-peer) เช่นเดียวกับ บล็อกเชน ทั้งนี้ ฐานข้อมูล บล็อกเชน ของ อีเทอริยม ถูกเก็บรักษาและพัฒนาปรับปรุงโดยโหนดหลายโหนดที่เชื่อมต่อกันเป็นเครือข่าย โหนดแต่ละโหนดในเครือข่ายจะเรียกใช้ EVM และใช้คำสั่งเดียวกัน ด้วยเหตุผลนี้ อีเทอริยม จึงถูกอธิบายว่าเป็น “คอมพิวเตอร์ระดับโลก (world computer)”

การประมวลผลขนานขนาดใหญ่ในเครือข่าย อีเทอริยม ไม่ได้ทำให้การคำนวณมีประสิทธิภาพมากขึ้น เพราะในความเป็นจริงแล้วขั้นดังกล่าวทำให้การคำนวณบน อีเทอริยม ช้าลง และมีค่าใช้จ่ายสูงกว่าการคำนวณบน “คอมพิวเตอร์” แบบดั้งเดิม ดังนั้น โหนด อีเทอริยม ทุกโหนด จะประมวลผลโดยใช้ EVM เพื่อรักษาระดับความความสัมพันธ์ระหว่าง บล็อกเชน ทั้งนี้ การกระจายข้อมูลจะช่วยให้ อีเทอริยม สามารถทนข้อบกพร่องต่าง ๆ ในระดับรุนแรงได้ (extreme levels of fault tolerance) ทำให้มั่นใจได้ว่าไม่มีการหยุดทำงานของระบบ (ensures zero downtime) และทำให้ข้อมูลที่เก็บอยู่ใน บล็อกเชน ไม่เปลี่ยนแปลงและสามารถกันการดักจับต่าง ๆ ได้ (censorship-resistant) ตลอดไป

ตัวของแพลตฟอร์ม อีเทอริยม เองนั้น เป็นสิ่งที่ไม่มีความลักษณะพิเศษใด ๆ (featureless) หรือไม่มีมูลค่า (value-agnostic) เช่นเดียวกับภาษาโปรแกรม (programming languages) ซึ่งก็ขึ้นอยู่กับผู้ประกอบการและนักพัฒนาจะตัดสินใจว่าควรจะใช้ภาษานั้นสำหรับทำอะไร อย่างไรก็ตาม เห็นได้ชัดว่าแอปพลิเคชันบางประเภทใช้ประโยชน์จากความสามารถของ Ethereum มากกว่าแอปพลิเคชันอื่น

ซึ่ง อีเทอริยม นั้นเหมาะอย่างยิ่งสำหรับแอปพลิเคชันที่ก่อให้เกิดการปฏิสัมพันธ์ทางตรงโดยอัตโนมัติระหว่างผู้ใช้ด้วยกัน หรืออำนวยความสะดวกในการประสานงานของกลุ่มต่าง ๆ ในเครือข่าย อีเทอริยม ตัวอย่างเช่น แอปพลิเคชันสำหรับการประสานงานในตลาดซื้อขายแบบ peer-to-peer (peer-to-peer marketplaces) หรือระบบอัตโนมัติเพื่อทำสัญญาทางการเงินที่ไม่ซับซ้อน บิทคอยน์ ช่วยให้บุคคลสามารถแลกเปลี่ยนเงินตราได้โดยไม่ต้องผ่านพ่อค้าคนกลาง เช่น สถาบันการเงิน ธนาคาร หรือรัฐบาล ซึ่งเครือข่าย อีเทอริยม อาจส่งผลกระทบแผ่ขยายออกไปได้มาก

โดยในทางทฤษฎีแล้วการปฏิสัมพันธ์ด้านการเงินหรือการซื้อขายแลกเปลี่ยนที่มีความซับซ้อนก็สามารถทำได้โดยอัตโนมัติและมีความน่าเชื่อถือผ่านการใช้รหัสที่ประมวลผลบน Ethereum นอกเหนือจากการเป็นแอปพลิเคชันทางการเงินแล้ว การดำเนินงาน อาทิ การลงทะเบียนสินทรัพย์ (asset-registries) การลงคะแนน (voting) การกำกับดูแล (governance) และการเชื่อมโยงเครือข่ายอินเทอร์เน็ตของสิ่งต่าง ๆ (internet of things) อาจได้รับอิทธิพลจากอย่างมากจากแพลตฟอร์ม Ethereum เนื่องจากการดำเนินการเหล่านี้ที่เน้นความสำคัญของปัจจัยแวดล้อมซึ่งมีความน่าเชื่อถือ (trust) ความปลอดภัย (security) และความคงที่ (permanence)

อีเธอเรียม นั้นรวบรวมคุณลักษณะและเทคโนโลยีมากมายที่ผู้ใช้ บิทคอยน์ คั่นเคย์ ในขณะเดียวกันก็มีการปรับปรุงคุณสมบัติและมีนวัตกรรมมากมายเป็นของตนเอง

ในขณะที่ บล็อกเชน ของ บิทคอยน์ เป็นรายการการทำธุรกรรมอย่างแท้จริง (list of transactions) โดยเป็นหน่วยพื้นฐานของ Ethereum คือบัญชี (account) ซึ่ง บล็อกเชน ของ อีเธอเรียม จะติดตามสถานะของทุกบัญชี และการเปลี่ยนสถานะของบัญชีบน บล็อกเชน ของ อีเธอเรียม จะเป็นการโอนค่าและข้อมูลระหว่างบัญชี ซึ่งแบ่งประเภทของบัญชีได้ 2 ประเภท ได้แก่

1. บัญชีที่มีเจ้าของจากภายนอก (Externally Owned Accounts, EOAs) ซึ่งถูกควบคุมโดยกุญแจส่วนตัว (private key)
2. บัญชีสัญญา (Contract Accounts) ซึ่งควบคุมโดยรหัสสัญญา (contract code) และสามารถใช้งานได้ (activate) โดย EOA เท่านั้น

สำหรับผู้ใช้ส่วนใหญ่ ความแตกต่างพื้นฐานระหว่างสองบัญชีนี้คือ ผู้ใช้ซึ่งเป็นมนุษย์ (human users) จะเป็นผู้ควบคุม EOA เนื่องจากผู้ใช้เหล่านี้เป็นผู้ควบคุมกุญแจส่วนตัวซึ่งควบคุม EOA อยู่ในทางตรงกันข้าม บัญชีสัญญา (Contract Accounts) จะถูกควบคุมโดยรหัสภายในบัญชีนั้น หากจะกล่าวว่าบัญชีสัญญาถูก “ควบคุม” โดยผู้ใช้ที่เป็นมนุษย์ ก็สามารถอธิบายได้ในแง่ที่ว่าบัญชีสัญญาเหล่านี้ได้รับการกำหนดชุดคำสั่งให้ถูกควบคุมโดย EOA ซึ่ง EOA นั้นก็ถูกควบคุมโดยใครก็ตามที่ถือกุญแจส่วนตัวซึ่งควบคุม EOA ไว้อีกที คำว่า “สัญญาอัจฉริยะ (smart contracts)” หมายถึง รหัสในบัญชีสัญญา ซึ่งเป็นโปรแกรมที่จะดำเนินการเมื่อมีการส่งธุรกรรมไปยังบัญชีนั้น ทั้งนี้ ผู้ใช้สามารถสร้างสัญญาใหม่ได้ด้วยการปรับรหัสที่ บล็อกเชน

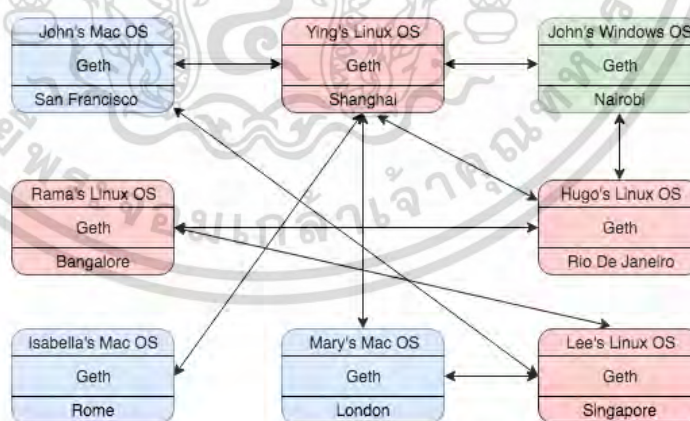
บัญชีสัญญาจะมีการดำเนินงานเฉพาะเมื่อได้รับคำสั่งให้ดำเนินงานจาก EOA เท่านั้น ดังนั้นจึงเป็นไปได้ที่บัญชีสัญญาจะดำเนินการด้วยตัวเอง เช่น การสร้างหมายเลขแบบสุ่มหรือชุดคำสั่ง API ซึ่งบัญชีสัญญาจะทำเช่นนั้นได้ในกรณีที่ได้รับคำสั่งจาก EOA เท่านั้น เนื่องจาก อีเธอเรียมต้องการให้โหนดมีความสอดคล้องกันในผลลัพธ์จากการคำนวณ ซึ่งจำเป็นต้องรับประกันให้เกิดการดำเนินการตามที่กำหนดอย่างเคร่งครัด

ผู้ใช้จะต้องเสียค่าธรรมเนียมการทำธุรกรรมจำนวนเล็กน้อยให้แก่เครือข่าย อีเทอริยม เช่นเดียวกับ บิทคอยน์ ซึ่งจะช่วยป้องกันไม่ให้ บล็อกเชน ของ อีเทอริยม ประมวลผลสิ่งไม่สำคัญหรือ ป้องกันระบบจากการประสงคร้าย เช่น การโจมตีแบบ Distributed Denial of Service (DDoS) เพื่อให้ระบบหยุดการทำงาน หรือเกิดรูปแบบไม่สิ้นสุด ผู้ส่งธุรกรรมจะต้องจ่ายเงินสำหรับขั้นตอนแต่ละขั้นตอนของ “โปรแกรม” ที่ใช้งาน รวมถึงการคำนวณและการจัดเก็บหน่วยความจำ ค่าธรรมเนียมเหล่านี้จะต้องจ่ายเป็นเงิน Ether ซึ่งเป็นสกุลเงินหลักของ อีเทอริยม

ค่าธรรมเนียมการทำธุรกรรมเหล่านี้จะถูกเก็บรวบรวมโดยโหนดที่ตรวจสอบความถูกต้องของเครือข่าย โดย “miner” ซึ่งเป็นโหนดในเครือข่าย อีเทอริยม จะได้รับข้อมูล ขยาย พิสูจน์ และประมวลผลธุรกรรม หลังจากนั้นก็จัดกลุ่มการทำธุรกรรม รวมถึงปรับปรุง “สถานะ” จำนวนมากของ บัญชีต่าง ๆ ใน บล็อกเชน ของ อีเทอริยม หรือสร้างบล็อก และ ประมวลผล ต่างก็แข่งขันกันกัน สร้างบล็อกเพิ่มเข้าไปใน บล็อก โดย ประมวลผล ที่ประสบความสำเร็จในการสร้างบล็อกจะได้รับรางวัลเป็นเงิน Ether ซึ่งถือว่าเป็นแรงจูงใจทางเศรษฐกิจเพื่อกระตุ้นให้คนใช้อุปกรณ์ฮาร์ดแวร์ของตน และใช้กระแสไฟเพื่อการดำเนินงานในเครือข่าย อีเทอริยม

### 2.3 Go-ethereum

Go-ethereum หรือ Geth เป็นเครื่องมือที่ใช้ในการรันโหนด อีเทอริยม แบบโหนดเต็มใน Go มีอินเทอร์เฟซ 3 ตัว ได้แก่ command line subcommands และ options เชีร์ฟเวอร์ JSON-rpc และคอนโซล Go-ethereumถูกสร้างขึ้นโดยมูลนิธิ อีเทอริยมซึ่งถูกเขียนโดยภาษา Go



Ethereum Blockchain Network

รูปที่ 2.2 แสดงการเชื่อมต่อของEthereum Blockchain Network โดยใช้ Geth protocol

(อ้างอิงโดย <https://medium.com/blockchannel>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.1. Geth console

เป็นเครื่องมือในการใช้งานคำสั่งซึ่งจะช่วยให้เชื่อมต่อกับ โหนด การทำงานของบล็อกเชน และดำเนินการต่างๆเช่นการสร้างและจัดการบัญชีแบบสอบถาม บล็อกเชน ในลงชื่อและส่งธุรกรรม บล็อกเชน และอื่น ๆ

## 2.4 Smartcontract

สัญญาอัจฉริยะ(Smartcontract) คือการนำเอาเงื่อนไขในสัญญาและกระบวนการทางธุรกิจ มาเขียนเป็นโค้ดโปรแกรมคอมพิวเตอร์ ซึ่งจะช่วยให้การบังคับใช้สัญญาตามข้อตกลงถูกดำเนินการโดยอัตโนมัติ ถูกต้อง คล่องตัว และโปร่งใส เพิ่มความน่าเชื่อถือ และเพิ่มประสิทธิภาพการทำงาน ช่วยลดขั้นตอนการทำงาน และลดข้อผิดพลาดที่อาจเกิดจากมนุษย์ (Human Error) ได้ ตัวอย่างเช่น ผู้ซื้อขายสินค้าออนไลน์ และผ่านโซเชียลสามารถลดความเสี่ยงของการชำระเงินลงได้ การโอนเงินข้ามประเทศจะสามารถทำได้ในเวลา 15 นาที จากเดิมที่ต้องใช้เวลาหลายวัน การอนุมัติสินเชื่อที่สามารถทำได้อย่างรวดเร็ว หรือการเคลมประกันแบบอัตโนมัติโดยไม่ต้องใช้เอกสาร รวมถึงยังเพิ่มความโปร่งใสในการทำธุรกรรม

ถ้าเปรียบเทียบกับง่าย ๆ ระหว่าง สัญญาอัจฉริยะ กับ เซอร์วิสหลังบ้าน ก็คือ เว็บเซอร์วิสจะรันอยู่บน เซิร์ฟเวอร์ เครื่องใดเครื่องหนึ่ง หรือไม่ก็รันบน คลาวด์ แต่หลักๆก็คือว่ามันรันอยู่ ณ คอมพิวเตอร์ใดเครื่องหนึ่งในระบบเครือข่ายอินเทอร์เน็ต ส่วน สัญญาอัจฉริยะ ของ อีเทอริยม นั้น เมื่อเราอัปโหลด แล้วจะเปรียบเสมือนว่ามันรันอยู่ในคอมทุกๆเครื่องที่เชื่อมต่อกับเครือข่าย บล็อกเชน ของ อีเทอริยม แต่ว่าที่รันอยู่นี้จะไม่ได้มีแค่ สัญญาอัจฉริยะ อย่างเดียวอาจจะมีการแปลภาษาเพื่อให้ code ฝั่ง หน้าบ้าน สามารถเรียก สัญญาอัจฉริยะ เพื่อดึงข้อมูลหรือเก็บข้อมูลลง บล็อกเชน

### 2.4..1 Solidity

Solidityเป็นภาษาที่นิยมที่สุดในการเขียน สัญญาอัจฉริยะ สำหรับ อีเทอริยม บล็อกเชน เป็นภาษาระดับสูงซึ่งเมื่อรวบรวมได้รับแปลงเป็นรหัสไบนารี EVM (Ethereum Virtual Machine) ภาษา solidity นั้นจะคล้ายกับ Scala, Groovy, Clojure, JRuby และอื่นๆ โดยในการคอมไพล์สร้างรหัสไบนารีซึ่งทำงานใน JVM (Java Virtual Machine)

## 2.4.2 JSON

JSON (JavaScript Object Notation) คือ รูปแบบของข้อมูลที่ใช้สำหรับแลกเปลี่ยนข้อมูลที่มีขนาดเล็ก ซึ่งคนสามารถทำความเข้าใจได้ง่าย และสามารถถูกสร้างและอ่านโดยเครื่องได้ง่าย มันถูกกำหนดภายใต้ภาษา JavaScript (JavaScript Programming Language, Standard ECMA-262 3rd Edition – December 1999.) JSON เป็นรูปแบบข้อมูลตัวอักษรที่มีความเป็นอิสระอย่างสมบูรณ์ แต่จะมีหลักการการเขียนที่คุ้นเคยกับนักเขียนโปรแกรมภาษาต่างๆ ได้ ไม่ว่าจะเป็น ภาษา C, C++, C#, Java, JavaScript, Perl, Python และอื่นๆ คุณสมบัติเหล่านี้ทำให้ JSON เป็นภาษาแลกเปลี่ยนข้อมูลที่มีสมบูรณ์แบบ

ในการทำงานหลายอย่างกับ JavaScript จะพบ JSON เข้ามาเกี่ยวข้องด้วย ยกตัวอย่างการทำงานกับ script หลายๆตัวที่มีการเรียกข้อมูลแบบ AJAX จะส่งข้อความตอบกลับมาในรูปแบบ JSON

มาตรฐานของฟอร์แมต JSON คือ RFC 4627 มี Internet media type เป็น application/json และมีนามสกุลของไฟล์เป็น .json

ปัจจุบัน JSON นิยมใช้ในเว็บแอปพลิเคชัน โดยเฉพาะ AJAX โดย JSON เป็นฟอร์แมตทางเลือกในการส่งข้อมูล นอกเหนือไปจาก XML ซึ่งนิยมใช้กันอยู่แต่เดิม สาเหตุที่ JSON เริ่มได้รับความนิยมเป็นเพราะกระชับและเข้าใจง่ายกว่า XML

```

1  {
2  "nonce": "0x0000000000000042",
3  "mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000",
4  "difficulty": "0x400",
5  "alloc": {},
6  "coinbase": "0x000000000000000000000000000000000000000000000000",
7  "timestamp": "0x00",
8  "parentHash": "0x000000000000000000000000000000000000000000000000",
9  "extraData": "0x436861696e536b696c6c732047656e6573697320426c66636b",
10 "gasLimit": "0xffffffff",
11 "config": {
12   "chainId": 42,
13   "homesteadBlock": 0,
14   "eip155Block": 0,
15   "eip158Block": 0
16 }
17 }

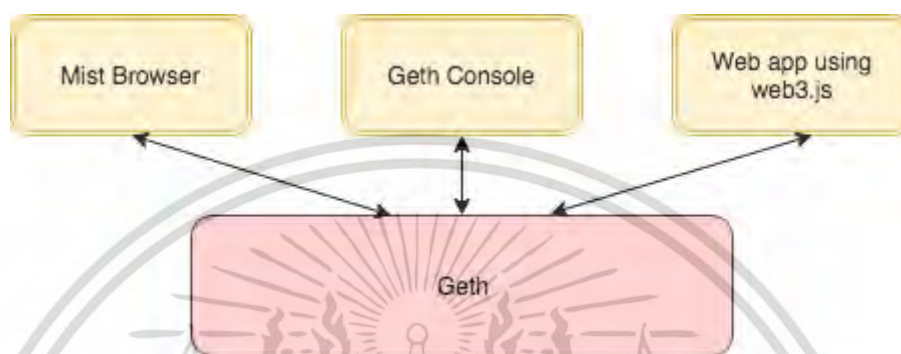
```

รูปที่ 2.3 ตัวอย่างภาษา JSON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 Web3.Js

Web3.Js จะคล้ายกับ geth , mist browser และอื่นๆที่เอาไว้ส่งข้อมูลไปยัง อีเทอริยม โหนด โดย Web3.Js เป็น Javascript library ที่สามารถสื่อสารกับ อีเทอริยม โหนด โดยสามารถสร้าง Web3.js ไลบรารี แอปพลิเคชันแบบไม่มีศูนย์กลางตาม เว็บเบราว์เซอร์เพื่อตอบโต้กับ อีเทอริยม โหนด ได้ ดังภาพ 2.4



รูปที่ 2.4 แสดงการเชื่อมต่อของโปรแกรมที่ใช้ไปยัง Geth protocol  
(อ้างอิงโดย <https://medium.com/blockchannel>)

## 2.6 Node.js

Node.js คือ open source server environment เพื่อให้เราเขียนซอฟต์แวร์มารันตามที่ต้องการ ภายใต้สิ่งแวดล้อมของ โหนด นี้ โดยภาษาที่ใช้เขียน คือภาษา JavaScript เหมือนที่ใช้เขียนบนหน้าเว็บ โดยจะมีตัว compiler คือ google JavaScript engine V8 ก็คือตัวประมวลผลภาษา JavaScript ที่ทาง กูเกิล พัฒนาขึ้นมา (เป็น open source) สรุปคือ การเขียนโปรแกรมด้วยภาษา JavaScript โดยความสามารถมันทำได้หลายอย่างมาก ไม่ว่าจะเป็นประมวลผลงานต่างๆ หรือติดต่อผ่าน socket หรือไปจนถึงสั่ง command line บนเครื่อง

Node.js ใช้ V8 มาเป็นตัว compiler เพราะว่า มันประมวลผลได้รวดเร็ว ผลที่ได้คือมันทำตัวคล้ายๆกับเป็น parallel execution คือ ทำงานหลายอย่างพร้อมกันในเวลาเดียวกัน แต่ความจริงคือไม่ใช่ เพราะว่ามันเป็น single thread คริ๊บ แต่มันปรับกระบวนการโดยเอาแต่ละงานไปเข้าคิวเวลาเอาไว้ก่อน (ยังไม่ประมวลผล) แล้วค่อยประมวลผลงานตามคิวเวลา ไม่ได้เรียงตามขั้นตอนที่เสร็จหรือไม่เสร็จ ซึ่งกระบวนการเข้าคิวเวลามันใช้เวลาน้อยมากๆ เราเลยมองเห็นเป็นว่า อ้อ มันเริ่มต้นทำงานได้พร้อมกัน สัญลักษณ์ของโปรแกรม Node.js แสดงดังภาพที่ 2.5



รูปที่ 2.5 ภาพแสดงสัญลักษณ์ของ node.js  
(อ้างอิงโดย <https://nodejs.org/en/>)

## 2.7 Node Package Manager

Node Package Manager(NPM) คือ เครื่องมือ สำหรับจัดการ แพ็คเกจ หรือ โมดูล ที่มีการสร้างไว้แล้วมาใช้งานใน แอปพลิเคชัน ที่สร้างขึ้นใหม่ เมื่อมี แพ็คเพจ ถูกนำมาใช้ใน แอปพลิเคชัน จะเรียก แพ็คเกจ เหล่านั้นว่า dependency ของ แอปพลิเคชัน โดย NPM ถูกติดตั้งมาพร้อมกับ Node.js และสามารถใช้งานได้ผ่านทาง command-line สัญลักษณ์ของโปรแกรม Node package manager แสดงดังภาพที่ 2.6



รูปที่ 2.6 ภาพแสดงสัญลักษณ์ของ Node package manager  
(อ้างอิงโดย <https://www.npmjs.com/>)

## 2.8 เว็บแอปพลิเคชัน

### 2.8.1 HTML

ภาษาหลักที่ใช้ในการเขียนเว็บเพจ โดยใช้ Tag ในการกำหนดการแสดงผล HTML ย่อมาจากคำว่า Hypertext Markup Language โดย Hypertext หมายถึง ข้อความที่เชื่อมต่อกันผ่านลิงก์ (Hyperlink) Markup language หมายถึงภาษาที่ใช้ Tag ในการกำหนดการแสดงผลสิ่งต่าง ๆ ที่แสดงอยู่บนเว็บเพจ ดังนั้น HTML จึงหมายถึง ภาษาที่ใช้ Tag ในการกำหนดการแสดงผลเว็บเพจที่ต่างก็เชื่อมถึงกันใน Hyperspace ผ่าน Hyperlink นั่นเอง

ความเป็นมาของ HTML เริ่มขึ้นเมื่อปี 1980 เมื่อ Tim Berners Lee เสนอต้นแบบสำหรับนักวิจัยใน CERN เพื่อแลกเปลี่ยนเอกสาร ข้อมูลด้านการวิจัย โดยใช้ชื่อว่า Enquire ในปี 1990 คำเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้เขียนโปรแกรมเบราวเซอร์ และทดลองรันบนเซิร์ฟเวอร์ที่เค้าพัฒนาขึ้น HTML ได้รับการรู้จักจาก HTML Tag ซึ่งมีอยู่ 18 Tag ในปี 1991

HTML ถูกพัฒนาจาก SGML และ Tim ก็คิดเสมือนว่า HTML เป็นโปรแกรมย่อยของ SGML อยู่ในตอนนั้น ต่อมาในปี 1996 เพื่อกำหนดมาตรฐานให้ตรงกัน W3C World Wide Web Consortium จึงเป็นผู้กำหนดสเปคทั้งหมดของ HTML และปี 1999 HTML 4.01 ก็ถือกำเนิดขึ้น โดยมี HTML 5 ซึ่งเป็น Web Hypertext Application ถูกพัฒนาต่อมาในปี 2004 นอกจากนี้ยังมีการพัฒนาไปเป็น XHTML ซึ่ง คือ Extended HTML ซึ่งมีความสามารถและมาตรฐานที่รัดกุมกว่าอีกด้วย โดยอยู่ภายใต้การควบคุมของ W3C (World Wide Web Consortium)

## 2.8.2 HTTP (Hypertext Transfer Protocol)

เป็นกลไกหรือโปรโตคอลหลักที่ใช้แลกเปลี่ยนข้อมูลกันระหว่างเซิร์ฟเวอร์และไคลเอนต์ของ เวิลด์ไวด์เว็บ โดยถูกออกแบบมาให้มีความกระชับรัด สามารถทำงานได้รวดเร็ว มีกระบวนการทำงานที่ไม่ซับซ้อน และมีคำสั่งที่ใช้กันไม่มากนัก แต่สามารถรองรับข้อมูลได้ทุกแบบ ไม่ว่าจะเป็นข้อมูลทั่วไปที่เข้ารหัสแบบ MIME หรือข้อมูลที่เป็นกราฟิก

หลักการการทำงานทั่ว ๆ ไปของ HTTP คือ จะแบบการทำงานออกเป็น 2 ด้านคือ ด้านเว็บเซิร์ฟเวอร์ และด้านไคลเอนต์ โดยไคลเอนต์จะติดต่อเข้ามายังเซิร์ฟเวอร์โดยใช้โปรแกรมบราวเซอร์ และอ้างถึงแอดเดรสของเซิร์ฟเวอร์โดยใช้รูปแบบของ URL ส่วนด้านเซิร์ฟเวอร์จะส่งข้อมูลกลับมาในรูปแบบที่เป็น HTML โดยที่โปรโตคอล HTTP ใช้วิธีการเข้ารหัสในแบบ MIME เป็นมาตรฐานของการทำงาน

โครงสร้างข้อมูลของ HTTP จะแบ่งออกเป็น 2 ส่วนใหญ่ ๆ คือ ส่วนเฮดเดอร์ หรือเรียกว่า metadata จะเป็นส่วนเก็บข้อมูลที่จำเป็นต้องใช้ภายในโปรโตคอล ส่วนที่สองเป็นส่วนข้อมูลจริงที่ต้องการรับส่ง ทั้งนี้ HTTP ถูกออกแบบมาให้สามารถรับส่งข้อมูลผ่าน Proxy หรือ Firewall ต่าง ๆ ได้ โดยการทำงาน HTTP จะอาศัยโปรโตคอลพื้นฐาน TCP/IP ซึ่งทั่วไปจะใช้หมายเลขพอร์ตที่ 80

โปรโตคอล HTTP ในปัจจุบันได้พัฒนาขึ้นมาเป็นเวอร์ชัน 1.1 (จากเดิมคือ เวอร์ชัน 1.0) ซึ่งโปรแกรมบราวเซอร์ที่แพร่หลายทั่วไปนั้นจะสามารถรองรับโปรโตคอลในเวอร์ชันใหม่นี้ได้ และได้กำหนดไว้เป็นมาตรฐานใน RFC2068 แล้ว โดยใน HTTP เวอร์ชัน 1.1 นี้ได้เพิ่มประสิทธิภาพการทำงานให้สูงขึ้น และปรับปรุงในด้านต่าง ๆ ที่ทำให้ความสามารถมากขึ้น

## 2.8.3 CSS

CSS (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 ไม่ขึ้นอยู่กับเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื้อหาของเอกสาร 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

### ประโยชน์ของ CSS

1.CSS มีคุณสมบัติมากกว่า tag ของ html เช่น การกำหนดกรอบให้ข้อความ รวมทั้งสีรูปแบบของข้อความที่กล่าวมาแล้ว

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

3.CSS สามารถกำหนดแยกไว้ต่างหากจาก ไฟล์เอกสาร html และสามารถนำมาใช้ร่วม กับเอกสารหลายไฟล์ได้ การแก้ไขก็แก้เพียง จุดเดียวก็มีผลกับเอกสารทั้งหมด

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

### 2.8.4 JavaScript

เป็นภาษาสคริปต์ ที่มีลักษณะการเขียนแบบโปรโตไทป์ (Prototyped-based Programming) ส่วนมากใช้ในหน้าเว็บเพื่อประมวลผลข้อมูลที่ฝั่งของผู้ใช้งาน แต่ก็ยังมีใช้เพื่อเพิ่มเติมความสามารถในการเขียนสคริปต์โดยฝังอยู่ในโปรแกรมอื่น ๆ

JavaScript ถูกพัฒนาขึ้นโดย เน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator2.0 เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคปจึงได้ร่วมมือกับ บริษัทซันไมโครซิสเต็มส์ ปรับปรุงระบบของบราวเซอร์เพื่อให้สามารถติดต่อใช้งานกับภาษาจาวาได้ และได้ปรับปรุง LiveScript ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript สามารถทำให้ การสร้างเว็บเพจ มีลูกเล่นต่าง ๆ มากมาย และยังสามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม เป็นต้น

เนื่องจาก JavaScript ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการ และมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยม

เป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA การทำงานเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของ JavaScript จะต้องมี การแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้น JavaScript จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุน JavaScript แล้ว อย่างไรก็ตาม สิ่งที่ต้องระวังคือ JavaScript มีการพัฒนาเป็นเวอร์ชันใหม่ ๆ ออกมาด้วย ดังนั้นถ้าโค้ดของเวอร์ชันใหม่ ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิด error ได้

## 2.9 Truffle

เป็นเครื่องมือในการพัฒนาแอปพลิเคชันเหมือนกับ such ruby on rail, python/Django และอื่นๆ แต่Truffle นั้นจะเป็นที่นิยมในการพัฒนาสำหรับ แอปพลิเคชันแบบไม่มีศูนย์กลาง โดย Truffle นั้นย่อกระบวนการทั้งหมดในการนำ สัญญาอัจฉริยะ ไปไว้ใน บล็อกเชน ให้เหลือแค่ compiling and deploying สัญลักษณ์ของเครื่องมือ Truffle แสดงดังภาพที่ 2.7



รูปที่ 2.7 ภาพแสดงสัญลักษณ์ของ TRUFFLE

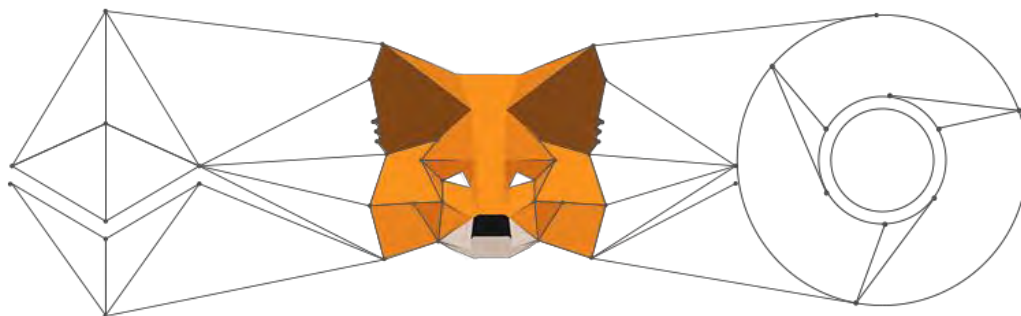
(อ้างอิงโดย <http://truffleframework.com/docs/>)

## 2.10 Metamask

MetaMask เป็นตัวที่ช่วยให้คุณสามารถเยี่ยมชมเว็บและใช้จ่าย อีเทอริยมในเบราว์เซอร์ของคุณได้ โดยช่วยให้คุณสามารถเรียกใช้ อีเทอริยม แอปพลิเคชันแบบไม่มีศูนย์กลาง ได้จากเบราว์เซอร์ของคุณโดยไม่ต้องใช้ อีเทอริยมโหนด เต็มรูปแบบ

MetaMask มีระบบรักษาความปลอดภัยเฉพาะตัวที่มีการรักษาความปลอดภัยให้อินเทอร์เน็ตเพซสำหรับผู้ใช้ในการจัดการข้อมูลเฉพาะตัวของคุณในไซต์ต่างๆและลงนามในการทำธุรกรรม บล็อกเชนคุณสามารถติดตั้งส่วนเสริม MetaMask ใน Chrome, Firefox, Opera และเบราว์เซอร์ Brave ใหม่ได้ หากคุณเป็นนักพัฒนาซอฟต์แวร์คุณสามารถเริ่มพัฒนาเมตาทุกวันได้ สัญลักษณ์ของโปรแกรม Metamask แสดงดังภาพที่ 2.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ภาพแสดงสัญลักษณ์ของ Metamask  
(อ้างอิงโดย <https://metamask.io>)

### 2.11 Sublime text

Sublime Text : คือโปรแกรมแก้ไขข้อความ และ เขียนโค้ด สำหรับโปรแกรมเมอร์ และ นักพัฒนา ที่มีขนาดเล็ก เพียงแค่ 7 เมกะบิต แต่ยังคงประสิทธิภาพและฟังก์ชันไว้มากมาย ใช้งานง่าย และทำงานได้อย่างรวดเร็ว

ด้วย โปรแกรม Sublime Text นี้ รองรับการใช้งานหลากหลายภาษา ไม่ว่าจะเป็น ภาษาหลักๆ อย่าง C,C++, C# เขียนเว็บไซต์ HTML,PHP โค้ด CSS เขียนภาษา JAVA, Python หรืออื่นๆ อีกมากมาย ด้วยหน้าต่างที่ใช้งานเขียน ส่วนโค้ดแบ่งแต่ละส่วนอย่างชัดเจน สามารถแก้ไขข้อความหลายๆ บรรทัดพร้อมกันได้ ค้นหา Text แต่ละส่วนได้ มีแสดงโค้ดทั้งหมด ให้ผู้ใช้สามารถเลื่อนดู ด้านข้างได้

สำหรับ โปรแกรมเขียนโค้ด Sublime Text เป็นโปรแกรมเขียน แก้ไขโค้ด ที่มีประสิทธิภาพสูง ด้วยประสิทธิภาพจาก Python API ที่ช่วยให้คุณสามารถติดตั้งปลั๊กอินเสริมต่างๆ และ แพ็คเกจ ให้โปรแกรมมีประสิทธิภาพในการทำงานมากยิ่งขึ้น ช่วยให้งานเขียนโค้ด ทำได้อย่างรวดเร็วและประหยัดเวลา สามารถสลับการทำงานบน โปรเจค ที่คุณทำอยู่ได้อย่างรวดเร็ว โปรแกรมนี้ก็รองรับการใช้งานบนระบบปฏิบัติการ Windows Linux และ macOS



รูปที่ 2.9 ภาพแสดงสัญลักษณ์ของ Sublime Text

(อ้างอิงโดย <https://www.sublimetext.com/>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพียงการศึกษาเท่านั้น เมื่อผู้ยูเห็นหาไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

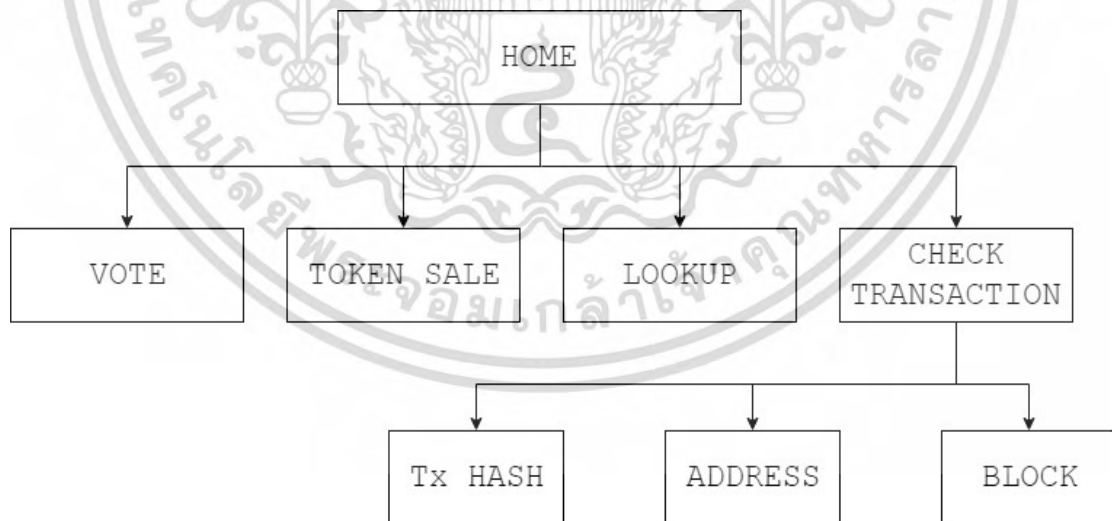
### การออกแบบระบบและการดำเนินงาน

#### 3.1 การออกแบบโครงสร้างของระบบการลงคะแนนเสียงแบบไม่มีศูนย์กลาง

ผู้พัฒนาได้ศึกษาค้นคว้าทฤษฎีและหลักการในการลงคะแนนเสียงโดยใช้บล็อกเชนในการจัดการเพื่อใช้ในการดำเนินงานของเว็บแอปพลิเคชันเพื่อใช้ในการลงคะแนนเสียง

โดยระบบที่ได้ออกแบบนั้น เป็นระบบซอฟต์แวร์ แบ่งออกเป็น 4 ระบบย่อย ได้แก่

1. ระบบลงคะแนนเสียง (Vote)
2. ระบบซื้อขายโทเคน (Token sale)
3. ระบบตรวจสอบคะแนนเสียง (Look up)
4. ระบบตรวจสอบข้อมูลในบล็อกเชน (Check Transaction)



รูปที่ 3.1 แสดงภาพรวมการออกแบบของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.1 ระบบลงคะแนนเสียง (Vote)

ระบบลงคะแนนเสียงจะสามารถลงคะแนนเสียงให้ผู้สมัครโดยสามารถกรอกจำนวนโทเค้นได้ตามจำนวนที่ผู้ลงคะแนนเสียงต้องการจะลงโดยผู้สมัครจะคะแนนเสียงตามโทเค้นที่ลงคะแนนเสียงไปโดยจะมีการแสดงหน้าผลคะแนนของผู้เข้าสมัครแบบตลอดเวลาผ่านการประมวลผลบนบล็อกเชน

### 3.1.2 ระบบซื้อขายโทเค้น (Token sale)

ระบบซื้อขายโทเค้นโดยทางผู้จัดทำกรโหวตนั้นสามารถกำหนดจำนวนขั้นสูงสุดของโทเค้นว่ามีจำนวนเท่าไร และ ราคาต่อโทเค้นหน่วยอีเทอริยมในการจำหน่ายโดยจะมีหน้าแสดงผลว่ามีจำนวนโทเค้นสูงสุดจำนวนเท่าไรและระบบได้ทำการขายโทเค้นไปจำนวนเท่าไรแล้วบ้าง รวมถึงแสดงยอดการซื้อของโทเค้นในหน่วยของอีเทอริยมที่ทางผู้จัดทำได้รับ

### 3.1.3 ระบบตรวจสอบคะแนนเสียง (Lookup)

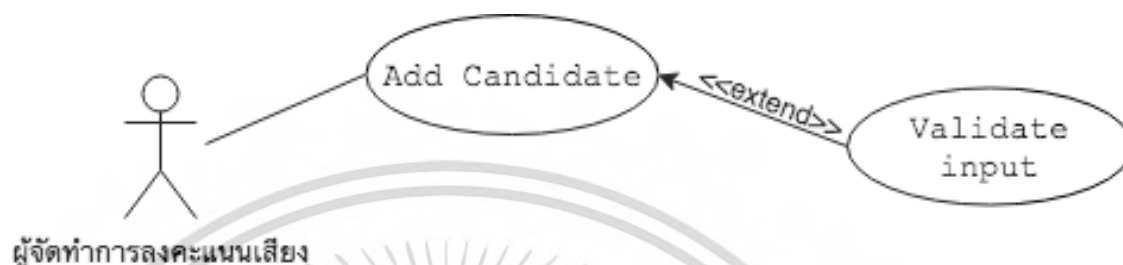
ระบบตรวจสอบคะแนนเสียง โดยระบบจะทำการแสดงผลการซื้อขายโทเค้น ผลการโหวตของเลขกุญแจสาธารณะโดยผู้ใช้งานสามารถนำเลขกุญแจสาธารณะที่ต้องการทราบจำนวนการซื้อขายโทเค้นของเลขกุญแจสาธารณะนั้นรวมถึงจะแสดงผลว่าเลขกุญแจสาธารณะเลขหมายนั้นได้ทำการลงคะแนนเสียงให้กับผู้สมัครคนใดบ้างและจำนวนกี่คะแนนเสียง เพื่อแสดงความโปร่งใสของการลงคะแนนเสียง

### 3.1.4 ระบบตรวจสอบข้อมูลในบล็อกเชน (Check Transaction)

ระบบตรวจสอบข้อมูลในบล็อกเชนโดยระบบแสดงผลการทำงานของบล็อกเชนทั้งหมดทั้งปัจจุบันและย้อนหลังเพื่อให้สามารถตรวจสอบข้อมูลได้ โดยสามารถตรวจสอบข้อมูลในบล็อกเชนผ่านการกรอกเลขของค่าแฮชของบล็อกที่ต้องการตรวจสอบ เลขกุญแจสาธารณะที่ต้องการตรวจสอบ และเลขของบล็อกที่ต้องการตรวจสอบ โดยทางผู้ใช้งานสามารถใช้ตัวระบบของการตรวจสอบข้อมูลบล็อกเชนได้ทั้งหมดรวมถึงการซื้อขายโทเค้นในระบบ และ รวมถึงสามารถนำเลขกุญแจสาธารณะที่มีในบล็อกเชนนั้นไปตรวจสอบได้ว่าการซื้อโทเค้นไปจำนวนเท่าใดและ โหวตผู้เข้าสมัครคนใดไปบ้าง

## 3.2 Use Case Diagram

### 3.2.1 ระบบการสร้างผู้สมัครลงคะแนนเสียง



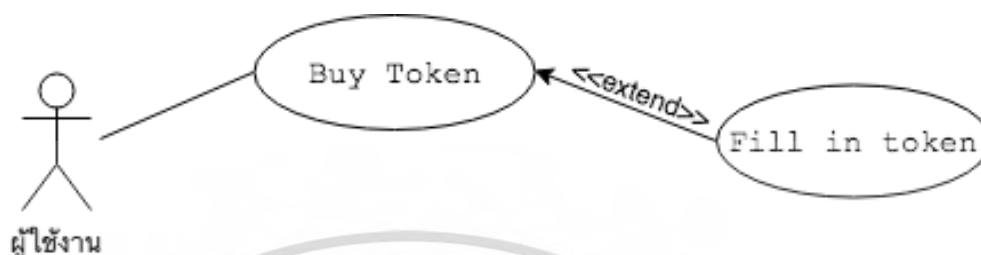
รูปที่ 3.2 ระบบสร้างผู้สมัครลงคะแนนเสียง

ตารางที่ 3.1 แสดงรายละเอียด Use Case ของระบบสร้างผู้สมัครลงคะแนนเสียง

Use Case ID:	1
Use Case Name:	ระบบการสร้างผู้สมัครลงคะแนนเสียง
Actors:	ผู้จัดทำการลงคะแนนเสียง
Description:	ผู้จัดทำการสร้างระบบและระบุผู้เข้ารับการลงคะแนนเสียง
Trigger:	ผู้ใช้งานอัปโหลดข้อมูลลงในบล็อกเชน
Pre-conditions:	-
Post-conditions:	ระบบการลงคะแนนเสียงถูกสร้างขึ้น
Normal Flow:	<ol style="list-style-type: none"> <li>1. ระบุผู้เข้าสมัคร</li> <li>2. อัปโหลดข้อมูลลงในบล็อกเชน</li> </ol>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.2 ระบบการซื้อขายโทเคน

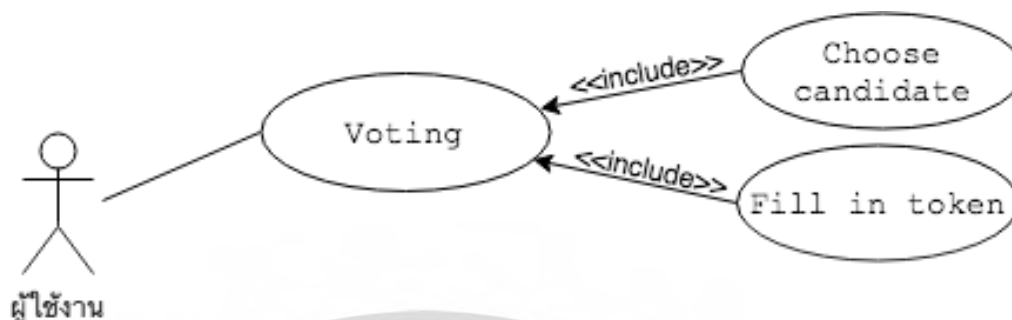


รูปที่ 3.3 ระบบการซื้อขายโทเคน

ตารางที่ 3.2 แสดงรายละเอียด Use Case ของระบบการซื้อขายโทเคน

Use Case ID:	2
Use Case Name:	ระบบการซื้อขายโทเคน
Actors:	ผู้ใช้งานแอปพลิเคชัน
Description:	ผู้ใช้งานทำการซื้อโทเคนเพื่อใช้ในการลงคะแนนเสียง
Trigger:	ผู้ใช้งานทำการกรอกโทเคนละซื้อโทเคน
Pre-conditions:	มีกุญแจสาธารณะในและจำนวนเงินดิจิทัลในการซื้อโทเคน
Post-conditions:	ผู้ใช้งานแอปพลิเคชันได้รับโทเคนสำหรับลงคะแนนเสียง
Normal Flow:	<ol style="list-style-type: none"> <li>1. ผู้ใช้งานกรอกจำนวนโทเคนที่ต้องการซื้อ</li> <li>2. ผู้ใช้งานกดซื้อโทเคน</li> <li>3. ผู้ใช้งานทำการยืนยันการซื้อโทเคน</li> </ol>

### 3.2.3 ระบบการลงคะแนนเสียง



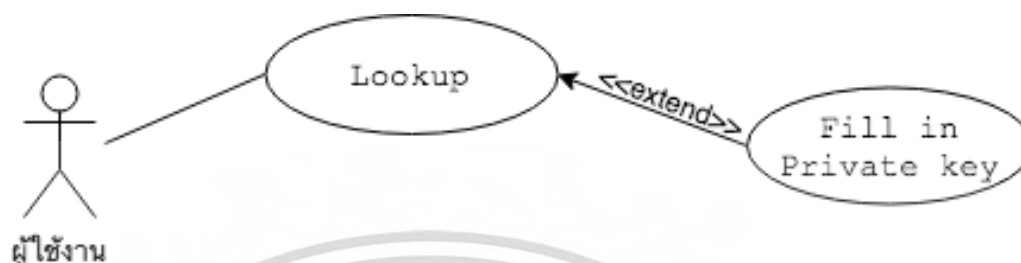
รูปที่ 3.4 ระบบการลงคะแนนเสียง

ตารางที่ 3.3 แสดงรายละเอียด Use Case ของระบบการลงคะแนนเสียง

Use Case ID:	3
Use Case Name:	ระบบการลงคะแนนเสียง
Actors:	ผู้ใช้งานแอปพลิเคชัน
Description:	ผู้ใช้งานทำการลงคะแนนเสียงด้วยโทเค้น
Trigger:	ผู้ใช้งานทำการกรอกโทเค้นและลงคะแนนเสียง
Pre-conditions:	มีโทเค้นในการลงคะแนนเสียง
Post-conditions:	ผู้สมัครได้รับคะแนนเสียงตามโทเค้นที่ผู้ใช้งานได้ลงคะแนนเสียงไป
Normal Flow:	<ol style="list-style-type: none"> <li>1. ผู้ใช้งานเลือกผู้สมัครที่ต้องการลงคะแนนเสียง</li> <li>2. ผู้ใช้งานกรอกจำนวนโทเค้นที่ต้องการลงคะแนนเสียง</li> <li>3. ผู้ใช้งานกดลงคะแนนเสียง</li> <li>4. ผู้ใช้งานทำการยืนยันการซื้อโทเค้น</li> </ol>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.4 ระบบตรวจสอบคะแนนเสียง



รูปที่ 3.5 ระบบตรวจสอบคะแนนเสียง

ตารางที่ 3.4 แสดงรายละเอียด Use Case ของระบบตรวจสอบคะแนนเสียง

Use Case ID:	4
Use Case Name:	ระบบตรวจสอบคะแนนเสียง
Actors:	ผู้ใช้งานแอปพลิเคชัน
Description:	ผู้ใช้งานตรวจสอบการซื้อขายโทเคนและการลงคะแนนเสียง
Trigger:	ผู้ใช้งานกรอกข้อมูลกุญแจสาธารณะที่ต้องการตรวจสอบ
Pre-conditions:	มีเลขกุญแจสาธารณะที่ต้องการตรวจสอบ
Post-conditions:	ผู้ใช้งานได้รับผลของเลขกุญแจสาธารณะที่ตรวจสอบ
Normal Flow:	<ol style="list-style-type: none"> <li>1. กรอกกุญแจสาธารณะ</li> <li>2. ผู้ใช้งานกดแสดงข้อมูล</li> </ol>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.5 ระบบตรวจสอบข้อมูลในบล็อกเชน



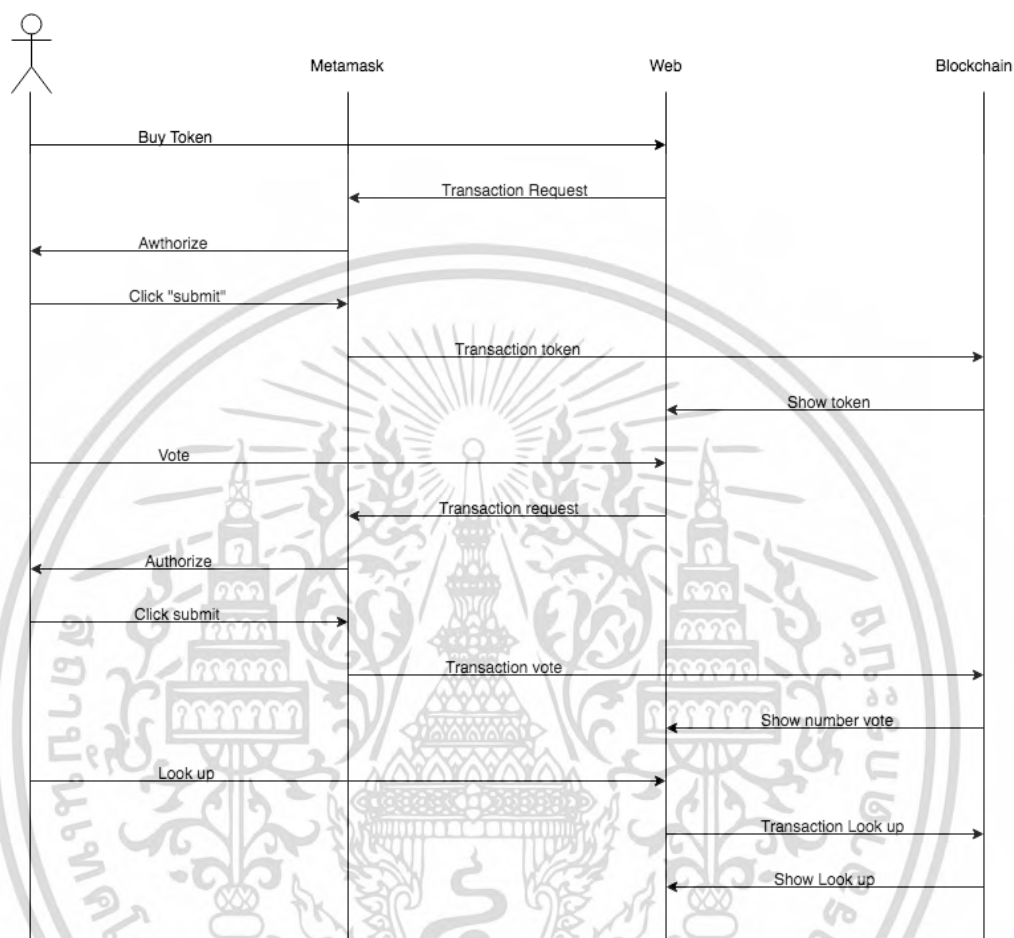
รูปที่ 3.6 ระบบตรวจสอบข้อมูลในบล็อกเชน

ตารางที่ 3.5 แสดงรายละเอียด Use Case ของระบบตรวจสอบข้อมูลในบล็อกเชน

Use Case ID:	5
Use Case Name:	ระบบตรวจสอบข้อมูลในบล็อกเชน
Actors:	ผู้ใช้งานแอปพลิเคชัน
Description:	ผู้ใช้งานตรวจสอบข้อมูลของบล็อกเชนทั้งในอดีตและปัจจุบัน
Trigger:	ผู้ใช้งานสามารถดูการทำงานของบล็อกเชน และสามารถกรอกข้อมูลกุญแจสาธารณะ บล็อก หรือ เลขแฮช ของบล็อกที่ต้องการจะตรวจสอบ
Pre-conditions:	-
Post-conditions:	ผู้ใช้งานได้รับผลของข้อมูลบล็อกเชนที่ตรวจสอบ
Normal Flow:	<ol style="list-style-type: none"> <li>1. ผู้ใช้งานตรวจสอบข้อมูลของบล็อกเชน</li> <li>2. ผู้ใช้งานกรอกข้อมูล ข้อมูลกุญแจสาธารณะ บล็อก หรือ เลขแฮช</li> <li>3. ผู้ใช้งานกดยืนยันเพื่อตรวจสอบ</li> </ol>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 Sequence Diagram



รูปที่ 3.7 Sequence Diagram ของระบบ

หลักการทำงานของ Sequence Diagram

- 1) ผู้ใช้งานซื้อโทเคน
- 2) เว็บส่งทรานแซคชันไป Metamask
- 3) Metamask ส่งคำยืนยันการซื้อโทเคนไปยังผู้ใช้งาน
- 4) ผู้ใช้งานยืนยันการซื้อโทเคน
- 5) Metamask ส่งข้อมูลการซื้อโทเคนไปบล็อกเชน
- 6) บล็อกเชนแสดงข้อมูลการซื้อโทเคนหน้าเว็บเบราว์เซอร์
- 7) ผู้ใช้งานทำการลงคะแนนเสียง
- 8) เว็บส่งทรานแซคชันไปยัง Metamask

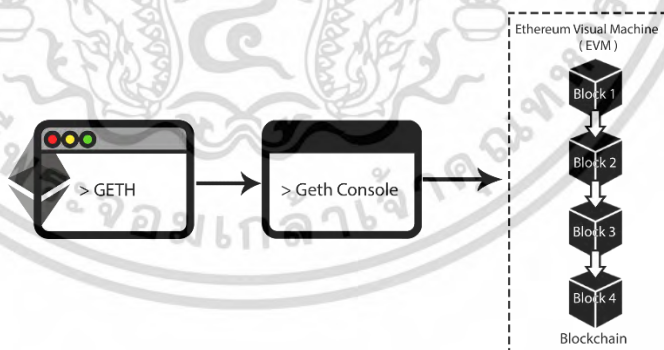
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการทํางานของ Sequence Diagram(ต่อ)

- 9) Metamask ส่งคํายืนยันการโหวตไปยังผู้ใช้งาน
- 10) ผู้ใช้งานยืนยันการลงคะแนนเสียง
- 11) Metamask ส่งข้อมูลการลงคะแนนเสียงไปบล็อกเชน
- 12) บล็อกเชนแสดงข้อมูลการลงคะแนนเสียงหน้าเว็บเบราว์เซอร์
- 13) ผู้ใช้งานตรวจสอบการลงคะแนนเสียง
- 14) เว็บเบราว์เซอร์ส่งทรานแซคชันไปยังบล็อกเชน
- 15) บล็อกเชนแสดงข้อมูลการตรวจสอบบนหน้าเว็บเบราว์เซอร์

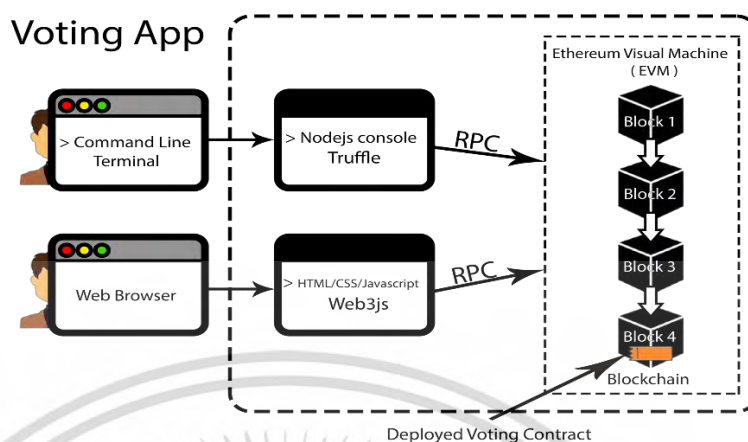
### 3.4 ภาพรวมขั้นตอนการดำเนินการสร้างของระบบ

ภาพรวมขั้นตอนการดำเนินการสร้างของระบบ โดยจะแบ่งขั้นตอนการดำเนินการเป็น2ส่วนหลักๆ คือขั้นตอนการสร้างบล็อกเชนส่วนตัว(private blockchain) และ ขั้นตอนการดำเนินการแอปพลิเคชันแบบไม่มีศูนย์กลาง(decentralized applications) โดยภาพที่ 3.8 จะแสดงส่วนของการจัดการการสร้างบล็อกเชนโดย Go-ethereum ผ่าน Gethคอนโซล และ ภาพที่ 3.9 แสดงในส่วน ติดตั้งสัญญาอัจฉริยะที่เขียนขึ้นลงบล็อกเชนส่วนตัวผ่าน truffle และ ส่วนของการใช้งานของผู้ใช้งานเว็บแอปพลิเคชันผ่านเว็บเบราว์เซอร์



รูปที่ 3.8 แสดงขั้นตอนการสร้างบล็อกเชนส่วนตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 ขั้นตอนการจัดการเว็บแอปพลิเคชัน

### 3.5 ขั้นตอนการดำเนินงานสร้างบล็อกเชนส่วนตัว

#### 3.5.1 ขั้นตอนการติดตั้ง brew

ติดตั้งbrew ด้วยคำสั่ง `/usr/bin/ruby -e "$(curl -fsSL`

`https://raw.githubusercontent.com/Homebrew/install/master/install)"` ผ่าน เทอมินอล ดังรูป 3.10

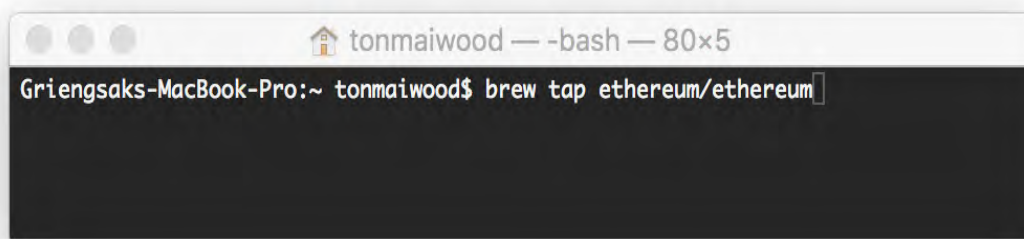
```
tonmaiwood — bash — 80x5
Griengsaks-MacBook-Pro:~ tonmaiwood$ /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

รูปที่ 3.10 ขั้นตอนการติดตั้งbrew

#### 3.5.2 ขั้นตอนการติดตั้ง Go-ethereum

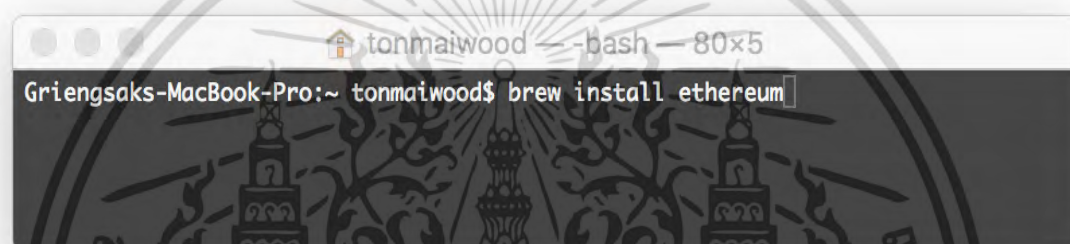
โหลด Go-ethereum ด้วยคำสั่ง `brew tap ethereum/Ethereum` ดังรูป 3.11 และ ติดตั้งGo-ethereum ด้วยคำสั่ง `brew install Ethereum` ผ่าน เทอมินอล ดังรูปที่ 3.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```
tonmaiwood — -bash — 80x5
Griengsaks-MacBook-Pro:~ tonmaiwood$ brew tap ethereum/ethereum
```

รูปที่ 3.11 ขั้นตอนการโหลด Go-ethereum

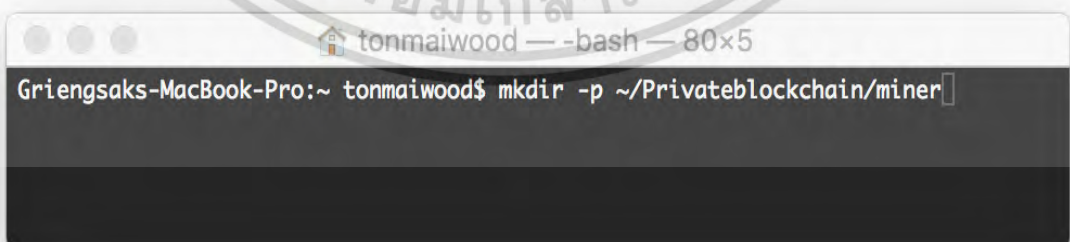


```
tonmaiwood — -bash — 80x5
Griengsaks-MacBook-Pro:~ tonmaiwood$ brew install ethereum
```

รูปที่ 3.12 ขั้นตอนการติดตั้ง Go-ethereum

### 3.5.3 ขั้นตอนการสร้างโพลเดอร์เพื่อจัดเก็บข้อมูลและกระเป๋าของบล็อกเชนส่วนตัว

สร้างโพลเดอร์เพื่อจัดเก็บข้อมูลและกระเป๋าของบล็อกเชนส่วนตัว ด้วยคำสั่ง `mkdir -p ~/Privateblockchain/miner` ผ่าน เทอมีนอล ดังรูป 3.13



```
tonmaiwood — -bash — 80x5
Griengsaks-MacBook-Pro:~ tonmaiwood$ mkdir -p ~/Privateblockchain/miner
```

รูปที่ 3.13 ขั้นตอนการสร้างโพลเดอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.4 ขั้นตอนการสร้าง เจเนซิสบล็อกเชน

โดยแต่ละบล็อกเชนจะมีบล็อกเชนบล็อกกำเนิดที่เรียกว่า เจเนซิสบล็อกเพื่อกำหนดเงื่อนไขของ บล็อกเชนและการเชื่อมต่อของบล็อกเชนที่สร้างขึ้น โดยสร้างไฟล์ชื่อ genesis.json ดังรูป 3.14 ไว้ใน โฟลเดอร์ Privateblockchain โดยแต่ละคำสั่งมีความหมายดังนี้



```

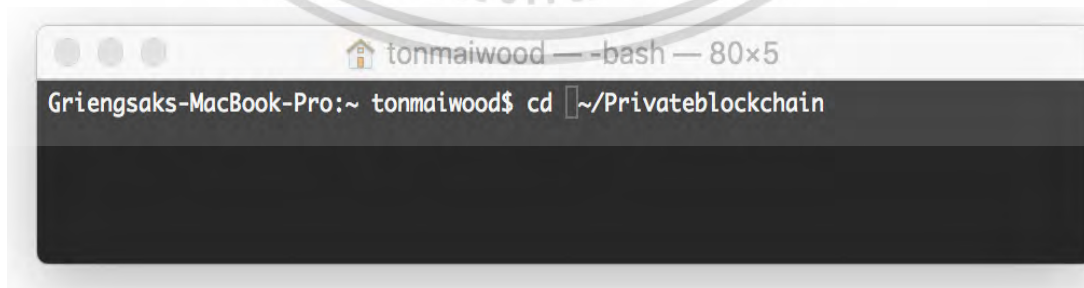
1 {
2   "nonce": "0x0000000000000042",
3   "mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000",
4   "difficulty": "0x400",
5   "alloc": {},
6   "coinbase": "0x000000000000000000000000000000000000000000000000",
7   "timestamp": "0x00",
8   "parentHash": "0x000000000000000000000000000000000000000000000000",
9   "extraData": "0x436861696e536b696c6c732047656e65733697320426c6f636b",
10  "gasLimit": "0xffffffff",
11  "config": {
12    "chainId": 42,
13    "homesteadBlock": 0,
14    "eip155Block": 0,
15    "eip158Block": 0
16  }
17 }

```

รูปที่ 3.14 แสดงไฟล์ genesis.json

### 3.5.5 ขั้นตอนการดำเนินการสร้างบล็อกเชนส่วนตัว

ใช้คำสั่ง `cd ~/Privateblockchain` ผ่าน เทอมินอล ดังรูปที่ 3.15 เพื่อย้ายตำแหน่งไปยัง โฟลเดอร์ Privateblockchain และ ใช้คำสั่ง `geth --datadir ~/ChainSkills/miner1 init genesis.json` ผ่าน terminal ดังรูป 3.16 เพื่อสร้างบล็อกเชนส่วนตัว จะได้โฟลเดอร์ geth และ keystore ดังรูป 3.17



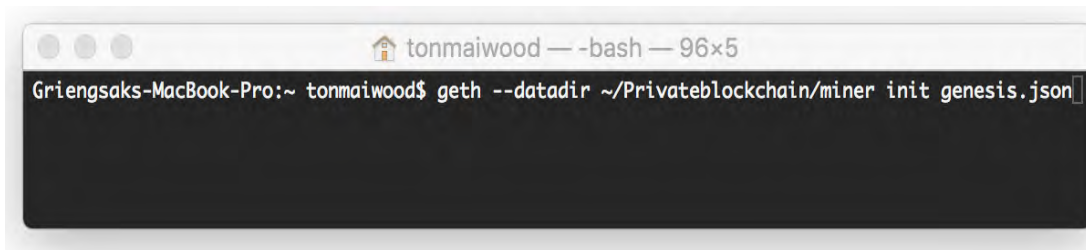
```

tonmaiwood --bash -- 80x5
Griengsaks-MacBook-Pro:~ tonmaiwood$ cd ~/Privateblockchain

```

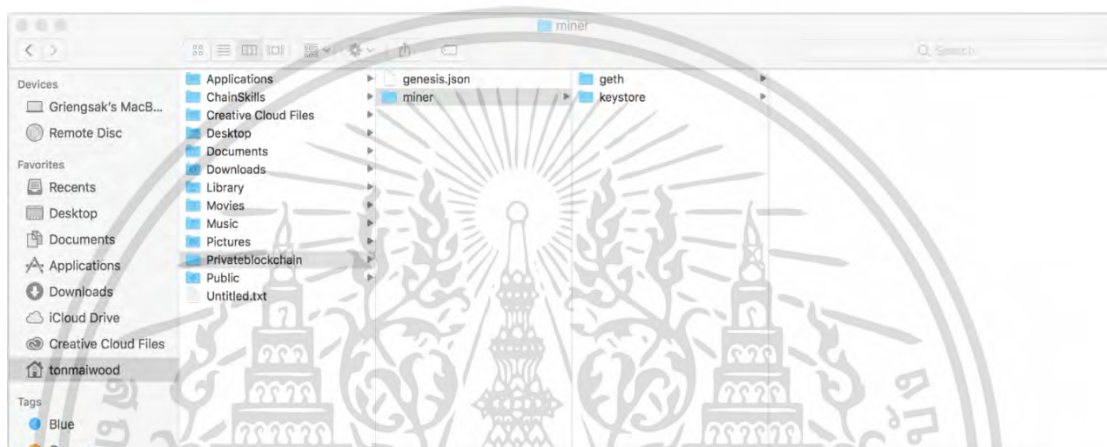
รูปที่ 3.15 แสดงคำสั่งย้ายโฟลเดอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```
tonmaiwood — bash — 96x5
Griengsaks-MacBook-Pro:~ tonmaiwood$ geth --datadir ~/Privateblockchain/miner init genesis.json
```

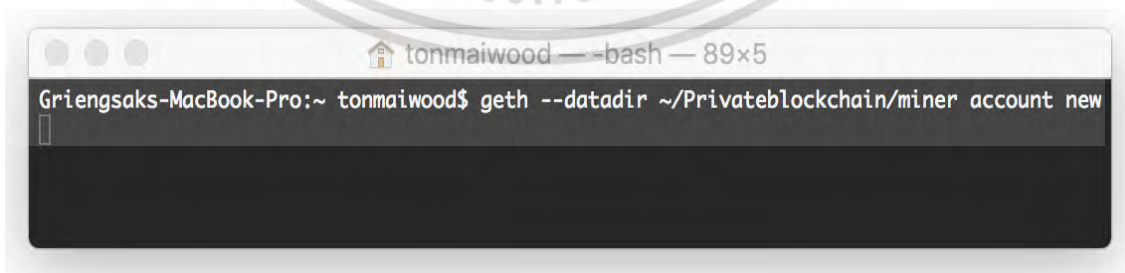
รูปที่ 3.16 แสดงคำสั่ง สร้างบล็อกเชนส่วนตัว



รูปที่ 3.17 แสดงโฟลเดอร์ที่จากการใช้คำสั่งในรูป 3.9

### 3.5.6 ขั้นตอนการสร้างแอดเคาท์

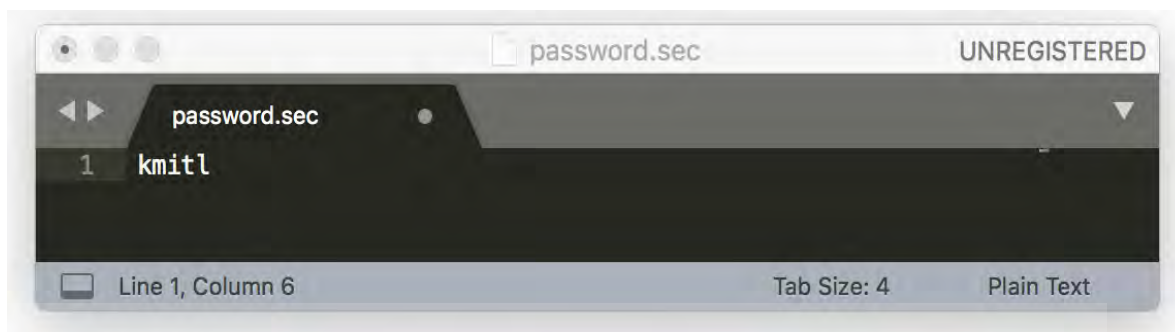
ขั้นตอนการสร้างแอดเคาท์เพื่อสร้างโทนดอีเทอริยม และประมวลผลยืนยันทรานแซคชัน (mining) โดยใช้คำสั่ง `geth --datadir ~/Privateblockchain/miner account new` ผ่าน เทอมินอล และตั้ง passphrase ในการเข้าแอดเคาท์ โดยแอดเคาท์ที่ถูกสร้างขึ้นจะไปในโฟลเดอร์ keystore และสร้างไฟล์ password.sec ไว้ในโฟลเดอร์ miner เพื่อเป็นรหัสในการปลดล็อคแอดเคาท์ที่สร้างดังรูป 3.19



```
tonmaiwood — bash — 89x5
Griengsaks-MacBook-Pro:~ tonmaiwood$ geth --datadir ~/Privateblockchain/miner account new
```

รูปที่ 3.18 แสดงคำสั่งสร้างแอดเคาท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

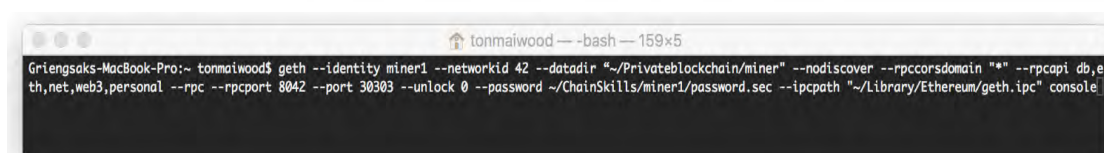


รูปที่ 3.19 แสดงไฟล์password.sec

### 3.5.7 ขั้นตอนการตั้งค่าการทำงานของโหนดอีเทอร์ียมบล็อกเชน

การตั้งค่าการทำงานของโหนดอีเทอร์ียมผ่าน เทอมินอล มีคำสั่งดังรูป 3.20 โดยแต่ละคำสั่งมีความหมายดังนี้

1. geth : เปิดใช้งานคำสั่ง
2. -- identity : ชื่อของโหนดที่สร้างขึ้น
3. -- networkid : ค่าที่ใช้ระบุเครือข่าย ถ้าต้องการจะเชื่อมต่อเครือข่ายเดียวกัน
4. -- datadir : โฟลเดอร์ที่เก็บข้อมูลของโหนดที่สร้างขึ้น
5. -- rpc / -- rpcport : หมายเลขพอร์ตที่ใช้ในการเชื่อมต่อกับเซิร์ฟเวอร์
6. -- port : หมายเลขพอร์ตที่ใช้ในการเชื่อมต่อกับโหนดอื่นๆเพื่อกระจายธุรกรรม
7. -- nodescover : ปิดการใช้งานเชื่อมต่อกับโหนดอื่น
8. -- unlock : เลือกแอคเคาท์ที่ใช้ในการทำงานของโหนด (0 คือแอคเคาท์เริ่มต้น)
9. -- password : โฟลเดอร์ที่เก็บ password.sec ที่ใช้ในการปลดล็อกแอคเคาท์
10. -- ipcpath : โฟลเดอร์ที่เก็บไฟล์ซ็อกเกต
11. -- rpccosdomain : รายชื่อโดเมนที่ใช้
12. -- rpcapi : apiที่จะใช้แสดงผ่าน HTTP-RPC
13. Console เปิดใช้ console ของGo-ethereum



รูปที่ 3.20 แสดงคำสั่งที่ใช้ในการทำงานโหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.8 ขั้นตอนการสร้างไฟล์ที่ใช้ในการเปิดใช้งานโหนดอีเทอร์เรียมบล็อกเชน

สร้างไฟล์ start.sh มีคำสั่งดังรูป 3.21 เพื่อไว้ใช้ในการเปิดการใช้งานโหนดอีเทอร์เรียมบล็อกเชน โดยภายในคำสั่งเหมือนกับรูปที่ 3.20 แต่เพิ่มคำสั่ง #!/bin/bash

```
start.sh
1 #!/bin/bash
2
3 geth --identity miner1 --networkid 42 --datadir "~/Privateblockchain/miner" --nodiscover --rpcor
```

รูปที่ 3.21 แสดงไฟล์start.sh และ คำสั่งภายในไฟล์

### 3.5.9 ขั้นตอนเปิดการทำงานของโหนดอีเทอร์เรียมบล็อกเชน

ต้องย้ายตำแหน่งของเทอมินอลไปยังโพลเดออร์ที่ใช้ในการเปิดใช้งานโหนดอีเทอร์เรียมบล็อกเชน แล้วพิมพ์คำสั่ง ./start.sh เพื่อเปิดการใช้งานไฟล์ในรูป 3.21 ที่ตั้งค่าไว้ โหนดอีเทอร์เรียมบล็อกเชนจะทำงาน พร้อมหน้าคอนโซลในการใช้งาน ดังรูป 3.22 โดยโหนดที่เปิดใช้งานแล้วยังไม่มีการคำนวณประมวลผลทรานแซคชัน

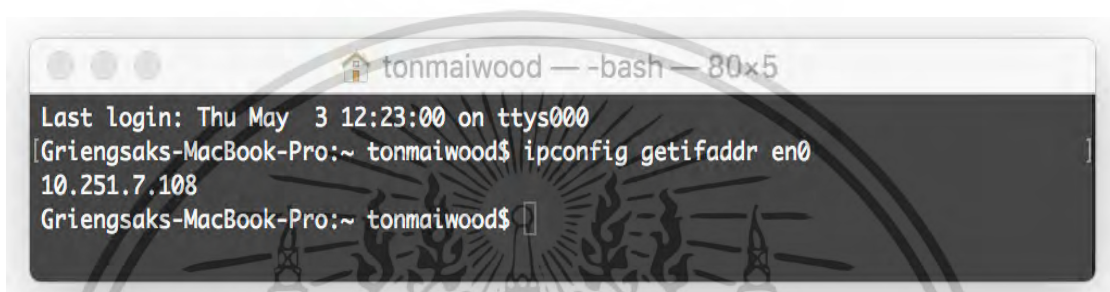
```
miner — geth | start.sh — 137x25
Last login: Thu May 3 12:19:44 on ttys000
Griengsaks-MacBook-Pro:~ tonmaiwood$ cd ~/Privateblockchain/miner
Griengsaks-MacBook-Pro:miner tonmaiwood$ ./start.sh
INFO [05-03|12:23:06] Maximum peer count ETH=25 LES=0 maxPeers=25
INFO [05-03|12:23:06] Starting peer-to-peer node instance=Geth/miner/v1.8.2-stable/darwin-amd64/go1.10
INFO [05-03|12:23:06] Allocated cache and file handles database=/Users/tonmaiwood/Privateblockchain/miner/geth/chaindata/cache=76
8 handles=128
INFO [05-03|12:23:06] Initialised chain configuration {ChainID: 42 Homestead: 0 DAO: <nil> DAOSupport: false EIP150: <nil
Engine: unknown"}
INFO [05-03|12:23:06] Disk storage enabled for ethash caches dir=/Users/tonmaiwood/Privateblockchain/miner/geth/ethash count=3
INFO [05-03|12:23:06] Disk storage enabled for ethash DAGS dir=/Users/tonmaiwood/.ethash count=2
INFO [05-03|12:23:06] Initialising Ethereum protocol version=0 [63 62] network=42
INFO [05-03|12:23:06] Loaded most recent local header number=0 hash=c0990b_cdd1ab td=1024
INFO [05-03|12:23:06] Loaded most recent local full block number=0 hash=c0990b_cdd1ab td=1024
INFO [05-03|12:23:06] Loaded most recent local fast block number=0 hash=c0990b_cdd1ab td=1024
INFO [05-03|12:23:06] Loaded local transaction journal transactions=0 dropped=0
INFO [05-03|12:23:06] Regenerated local transaction journal transactions=0 accounts=0
INFO [05-03|12:23:06] Starting P2P networking
INFO [05-03|12:23:06] RLPx listener up
INFO [05-03|12:23:06] IPC endpoint opened
INFO [05-03|12:23:06] HTTP endpoint opened
WARN [05-03|12:23:06] Referring to accounts by order in the keystore folder is dangerous!
WARN [05-03|12:23:06] This functionality is deprecated and will be removed in the future!
WARN [05-03|12:23:06] Please use explicit addresses! (can search via "geth account list")
INFO [05-03|12:23:07] Unlocked account address=0x948d8e79b3A6E66769B417b284Da22b8F724655e
Welcome to the Geth JavaScript console!
instance: Geth/miner/v1.8.2-stable/darwin-amd64/go1.10
INFO [05-03|12:23:07] Etherbase automatically configured address=0x948d8e79b3A6E66769B417b284Da22b8F724655e
coinbase: 0x948d8e79b3A6E66769B417b284Da22b8F724655e
at block: 0 (Thu, 01 Jan 1970 07:00:00 +07)
datadir: /Users/tonmaiwood/Privateblockchain/miner
modules: admin:1.0 debug:1.0 eth:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0
```

รูปที่ 3.22 แสดงการทำงานของโหนดอีเทอร์เรียมบล็อกเชนที่เปิดการใช้งานแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.10 ขั้นตอนการเชื่อมต่ออีโหนดเรียงบล็อกเชนกับโหนดอื่น

ในการเชื่อมเชื่อมต่ออีโหนดเรียงบล็อกเชนกับโหนดอื่นจะต้องมีเลขประจำตัวโหนดของเครื่องที่ต้องการจะเชื่อมต่อ และ ไอพี ของโหนดที่จะเชื่อมต่อโดยใช้คำสั่ง `ipconfig getifaddr en0` ดังรูป 3.23 ในการแสดงไอพีผ่านเทอร์มินอล และคำสั่ง `admin.nodeInfo.enode` ผ่านคอนโซลของรูป 3.22 ดังรูป 3.24 ในการแสดงเลขประจำตัวของโหนด และสร้างไฟล์ `static-nodes.json` ไว้ในโพลเดอร์ `miner` เพื่อใส่เลขประจำตัวโหนดและไอพี ดังรูปที่ 3.25



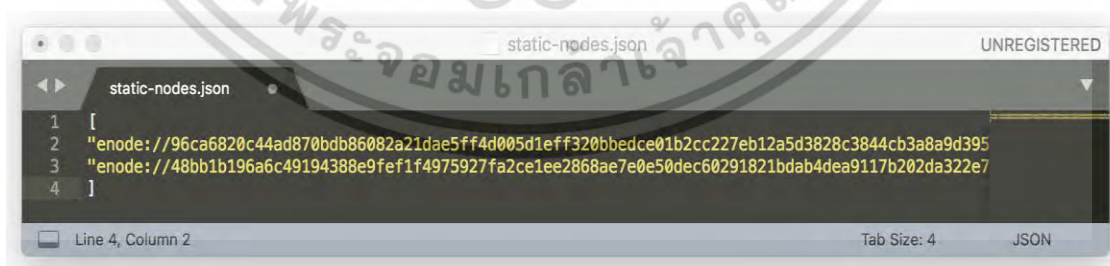
```
tonmaiwood — -bash — 80x5
Last login: Thu May 3 12:23:00 on ttys000
[Griengsaks-MacBook-Pro:~ tonmaiwood$ ipconfig getifaddr en0
10.251.7.108
Griengsaks-MacBook-Pro:~ tonmaiwood$
```

รูปที่ 3.23 แสดงคำสั่งแสดงไอพี



```
miner — geth — start.sh — 137x8
at block: 0 (Thu, 01 Jan 1970 07:00:00 +07)
datadir: /Users/tonmaiwood/Privateblockchain/miner
modules: admin:1.0 debug:1.0 eth:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0
[> admin.nodeInfo.enode
"enode://96ca6820c44ad870bdb86082a21dae5ff4d005d1eff320bbe01b2cc227eb12a5d3828c3844cb3a8a9d395
@[:]:30303?discport=0"
>
```

รูปที่ 3.24 แสดงคำสั่งแสดงเลขประจำตัวโหนด



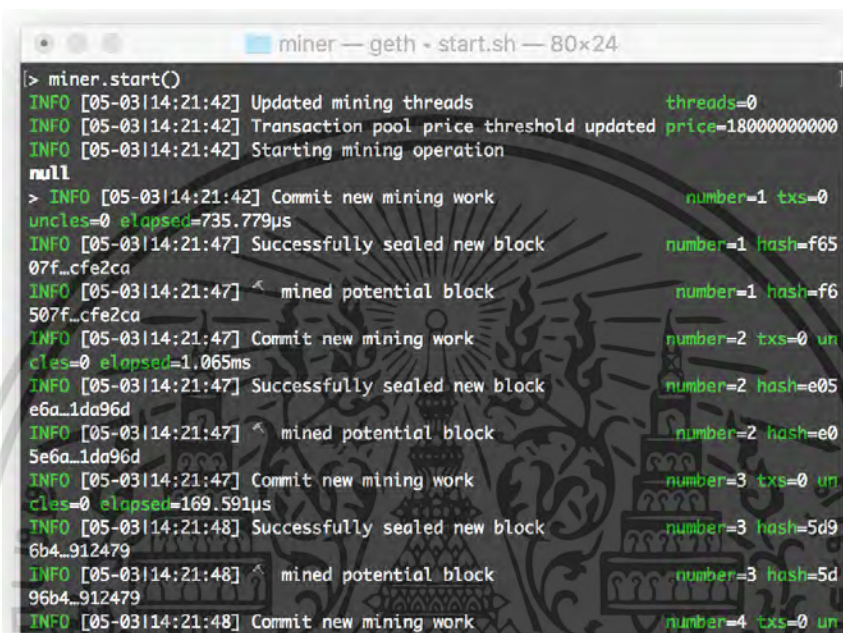
```
static-nodes.json UNREGISTERED
static-nodes.json
1 [
2   "enode://96ca6820c44ad870bdb86082a21dae5ff4d005d1eff320bbe01b2cc227eb12a5d3828c3844cb3a8a9d395
3   "enode://48bb1b196a6c49194388e9fef1f4975927fa2ce1ee2868ae7e0e50dec60291821bdab4dea9117b202da322e7
4 ]
Line 4, Column 2 Tab Size: 4 JSON
```

รูปที่ 3.25 แสดงไฟล์ที่ใส่เลขประจำตัวโหนดและไอพี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.11 ขั้นตอนการใช้คำสั่งประมวลผลโหนดอีเทอร์เรียมบล็อกเชน

การเปิดใช้งานคำสั่งประมวลผลทรานแซคชันเพื่อคอมเพิร์มทรานแซคชันในแต่ละบล็อก ผ่านคอนโซลในรูป 3.22 โดยใช้คำสั่ง `miner.start()` ระบบจะเริ่มทำการประมวลผลคอนเพิร์ม ทรานแซคชัน ดังรูป 3.26



```

> miner.start()
INFO [05-03|14:21:42] Updated mining threads                threads=0
INFO [05-03|14:21:42] Transaction pool price threshold updated price=18000000000
INFO [05-03|14:21:42] Starting mining operation
null
> INFO [05-03|14:21:42] Commit new mining work                number=1 txs=0
uncles=0 elapsed=735.779µs
INFO [05-03|14:21:47] Successfully sealed new block                number=1 hash=f65
07f...cfe2ca
INFO [05-03|14:21:47] ^ mined potential block                        number=1 hash=f6
507f...cfe2ca
INFO [05-03|14:21:47] Commit new mining work                number=2 txs=0 un
cles=0 elapsed=1.065ms
INFO [05-03|14:21:47] Successfully sealed new block                number=2 hash=e05
e6a...1da96d
INFO [05-03|14:21:47] ^ mined potential block                        number=2 hash=e0
5e6a...1da96d
INFO [05-03|14:21:47] Commit new mining work                number=3 txs=0 un
cles=0 elapsed=169.591µs
INFO [05-03|14:21:48] Successfully sealed new block                number=3 hash=5d9
6b4...912479
INFO [05-03|14:21:48] ^ mined potential block                        number=3 hash=5d
96b4...912479
INFO [05-03|14:21:48] Commit new mining work                number=4 txs=0 un

```

รูปที่ 3.26 แสดงการทำงานของโหนดอีเทอร์เรียมบล็อกเชนที่เริ่มการประมวลผลทรานแซคชัน

## 3.6 ขั้นตอนการดำเนินการแอปพลิเคชันแบบไม่มีศูนย์กลาง

### 3.6.1 ขั้นตอนการติดตั้ง truffle

ขั้นตอนการติดตั้ง truffle เฟรมเวิร์คที่ใช้ในการจัดการแอปพลิเคชันแบบไม่มีศูนย์กลางโดยใช้ node package manager ในการติดตั้ง โดยใช้คำสั่ง `npm install -g truffle` ดังรูป 3.27

```
tonmaiwood — -bash — 80x5
Last login: Thu May 3 18:04:16 on ttys001
Griengsaks-MacBook-Pro:~ tonmaiwood$ npm install -g truffle
```

รูปที่ 3.27 แสดงการติดตั้ง truffle

### 3.6.2 ขั้นตอนการสร้างโพลเดอร์สำหรับจัดการแอปพลิเคชันแบบไม่มีศูนย์กลาง

โดยใช้คำสั่ง `mkdir voting` ในการสร้างโพลเดอร์ และย้ายตำแหน่งเทมिनอลไปยังโพลเดอร์ที่สร้างขึ้นผ่านเทมिनอลด้วยคำสั่ง `cd voting` ดังรูป 3.28 และใช้คำสั่ง `truffle unbox webpack` ดังรูป 3.29 เพื่อสร้างtruffle งานลงในโพลเดอร์ที่สร้างไว้จะได้ โพลเดอร์ดังรูป 3.30

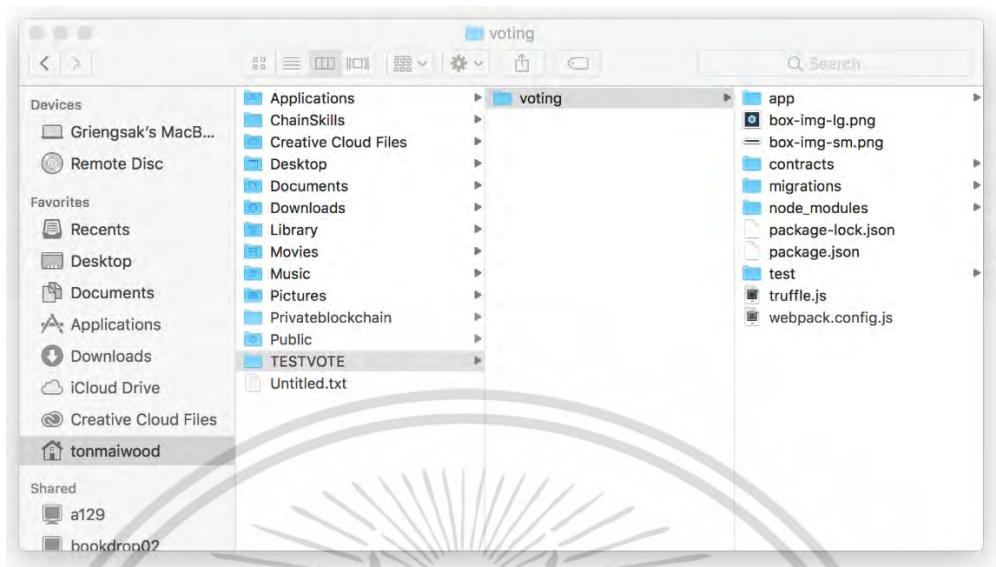
```
TESTVOTE — -bash — 80x5
Last login: Thu May 3 18:00:08 on ttys000
Griengsaks-MacBook-Pro:~ tonmaiwood$ cd ~/TESTVOTE
Griengsaks-MacBook-Pro:TESTVOTE tonmaiwood$ mkdir voting
Griengsaks-MacBook-Pro:TESTVOTE tonmaiwood$ cd ~/TESTVOTE/voting
```

รูปที่ 3.28 แสดงคำสั่งย้ายไปโพลเดอร์ลงคะแนน

```
voting — npm • node /usr/local/bin/truffle unbox webpack — 80x5
Last login: Thu May 3 18:04:16 on ttys001
Griengsaks-MacBook-Pro:~ tonmaiwood$ cd ~/TESTVOTE/voting
Griengsaks-MacBook-Pro:voting tonmaiwood$ truffle unbox webpack
Downloading...
Unpacking...
```

รูปที่ 3.29 แสดงคำสั่งสร้าง truffle ลงในโพลเดอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.30 แสดงโฟลเดอร์ติดตั้ง truffle

### 3.6.3 ขั้นตอนการตั้งค่า truffle

ขั้นตอนการตั้งค่า truffle นั้นจะตั้งโดยแก้ไขไฟล์ truffle.js โดยค่าตั้งต้นของพอร์ตที่ truffle ใช้ นั้นจะเป็น 8545 คือค่าพอร์ตที่ต่ออีเทอเรียมบล็อกเชนโหนดหลักของอีเทอเรียมบล็อกเชนแต่ในโปรเจกต์นี้จะใช้ในบล็อกเชนส่วนตัว จึงแก้ไขพอร์ตให้ตรงกับโหนดอีเทอเรียมบล็อกเชนที่สร้างไว้คือ 8042 ดังรูป 3.31

```

1 // Allows us to use ES6 in our migrations and tests.
2 require('babel-register')
3
4 module.exports = {
5   networks: {
6     development: {
7       host: 'localhost',
8       port: 8042,
9       network_id: '*' // Match any network id
10    }
11  }
12 }
13

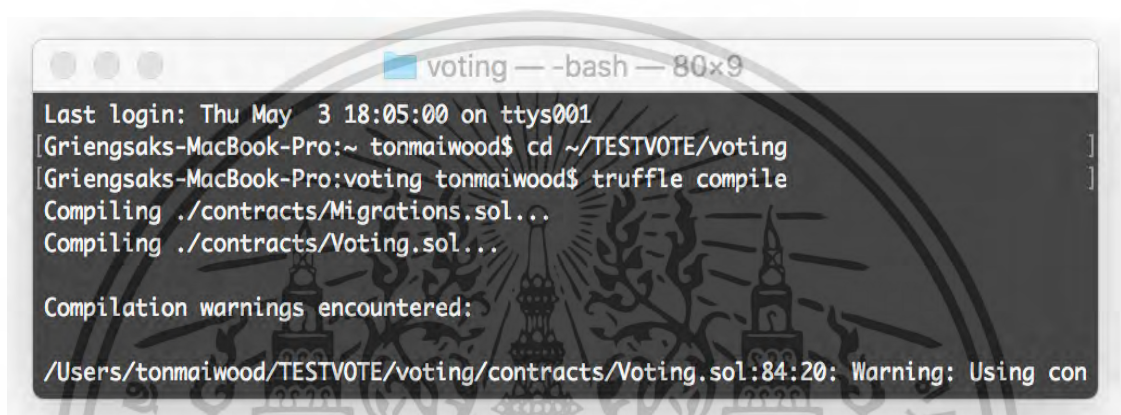
```

รูปที่ 3.31 แสดงไฟล์ truffle.js

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.6.4 ขั้นตอนการติดตั้งสัญญาอัจฉริยะลงในบล็อกเชนส่วนตัว

ขั้นตอนการติดตั้งสัญญาอัจฉริยะลงในบล็อกเชนหลังจากที่เขียนสัญญาอัจฉริยะด้วยภาษา solidity จะต้องทำการแปลงไฟล์จากสกุล SOL เป็นไฟล์สกุล JSON ด้วยคำสั่ง `truffle compile` จะได้ไฟล์เตอร์พร้อมไฟล์ที่แปลงเป็นสกุล JSON ดังภาพที่ 3.32 และทำการติดตั้งสัญญาอัจฉริยะลงในบล็อกเชนด้วยคำสั่ง `truffle migrate` ดังรูปที่ 3.33 ตัวของสัญญาอัจฉริยะจะอัปโหลดลงในบล็อกเชนส่วนตัวพร้อมใช้งาน

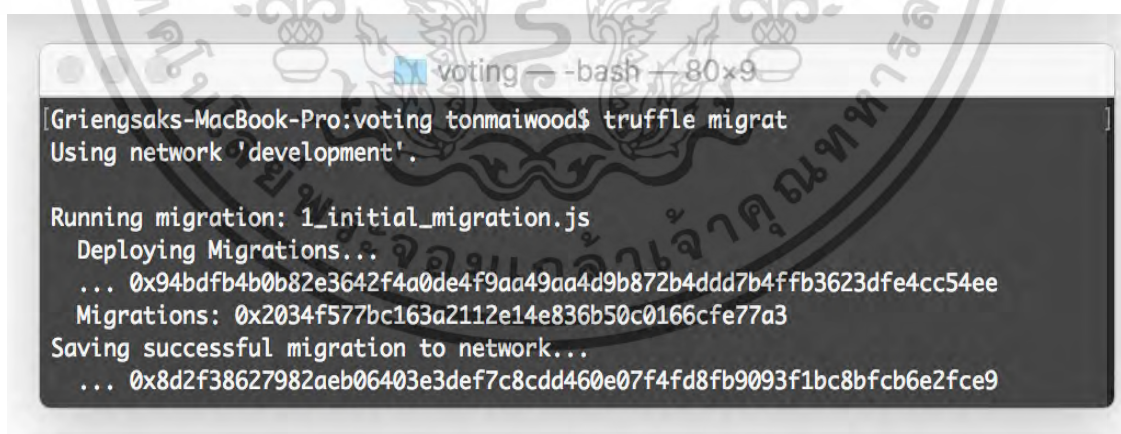


```
voting -- -bash -- 80x9
Last login: Thu May 3 18:05:00 on ttys001
[Griengsaks-MacBook-Pro:~ tonmaiwood$ cd ~/TESTVOTE/voting ]
[Griengsaks-MacBook-Pro:voting tonmaiwood$ truffle compile ]
Compiling ./contracts/Migrations.sol...
Compiling ./contracts/Voting.sol...

Compilation warnings encountered:

/Users/tonmaiwood/TESTVOTE/voting/contracts/Voting.sol:84:20: Warning: Using con
```

รูปที่ 3.32 แสดงคำสั่ง `truffle compile`



```
voting -- -bash -- 80x9
[Griengsaks-MacBook-Pro:voting tonmaiwood$ truffle migrate
Using network 'development'.

Running migration: 1_initial_migration.js
Deploying Migrations...
... 0x94bdfb4b0b82e3642f4a0de4f9aa49aa4d9b872b4ddd7b4ffb3623dfe4cc54ee
Migrations: 0x2034f577bc163a2112e14e836b50c0166cfe77a3
Saving successful migration to network...
... 0x8d2f38627982aeb06403e3def7c8cdd460e07f4fd8fb9093f1bc8bfc6e2fce9
```

รูปที่ 3.33 แสดงคำสั่ง `truffle migrate`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

# การทำงานของเว็บแอปพลิเคชัน

ผลสรุปเว็บแอปพลิเคชันจากการที่นำเดินการจากบทที่บนี้จะแสดงถึงขั้นตอนการทำงานของเว็บแอปพลิเคชัน

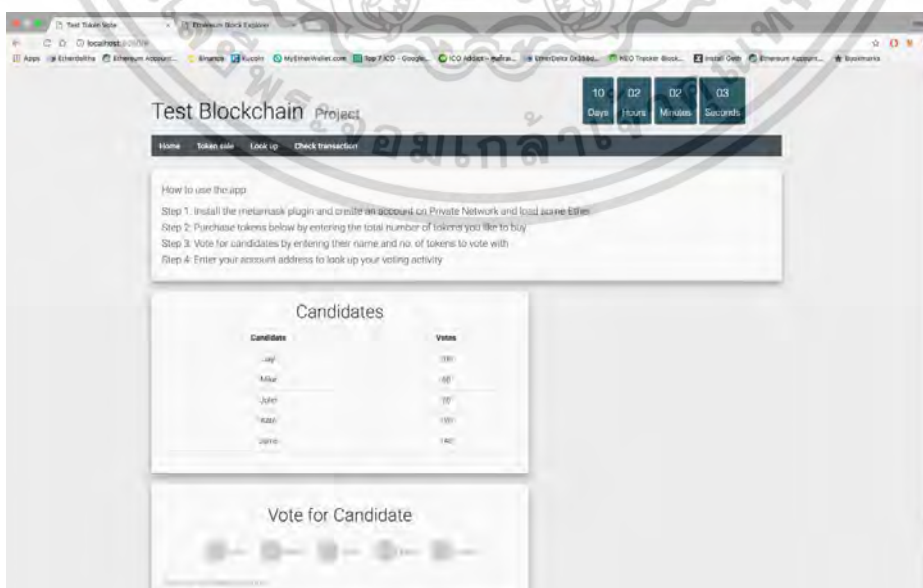
## การทำงานของระบบลงคะแนนเสียง

### 4.1 การเข้าสู่หน้าหลัก

เมื่อเปิดแอปพลิเคชันขึ้นมาจะได้หน้าหลักดังรูปที่ 4.1 โดยจะมีเมนูหลักรวมกัน 4 เมนู

1. Home
2. Token sale
3. Look up
4. Check Transaction

โดยหน้า Home จะแสดงการใช้งานเว็บแอปพลิเคชัน ชื่อผู้สมัคร และจำนวนคะแนนเสียงของผู้สมัครแต่ละคนรวมถึงในส่วนของการลงคะแนนเสียงให้ผู้สมัครแต่ละคน หน้า Token sale จะแสดงจำนวนโทเค็น ทั้งหมด จำนวนโทเค็นที่มีการซื้อไป จำนวนราคาต่อโทเค็น และจำนวนเงินที่ทางผู้สร้างได้รับรวมถึงหน้าต่างการซื้อTokenที่ใช้สำหรับการลงคะแนนเสียง หน้า Look up ใช้การตรวจสอบเลขกระเป๋าว่ากระเป๋านั้นได้มีการซื้อโทเค็นไปจำนวนเท่าไรและลงคะแนนเสียงให้ผู้สมัครคนไหนจำนวนเท่าไร และ Check Transaction จะเชื่อมต่อไปยังหน้าแสดงรายละเอียดทรานซัซชันทั้งหมด

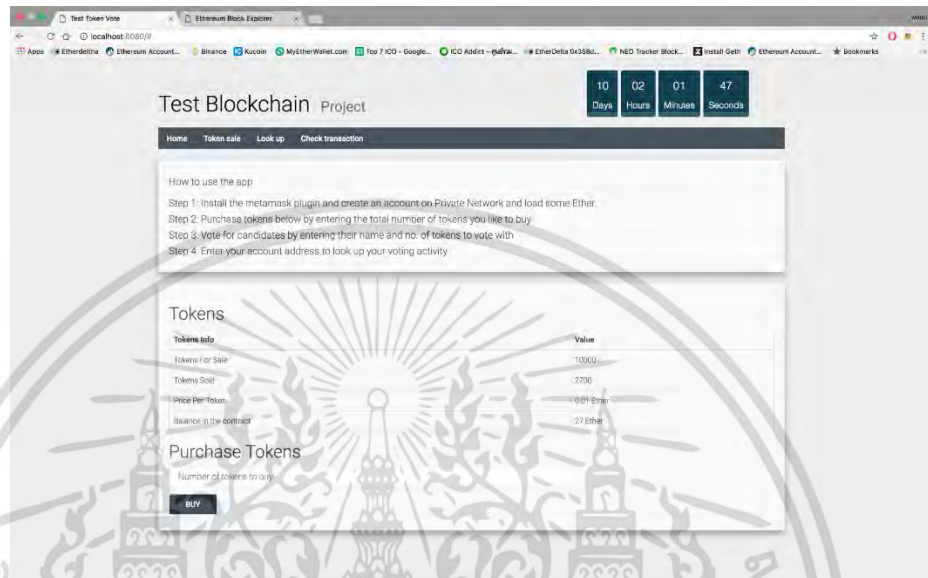


(รูปที่ 4.1 หน้าแรกของเว็บแอปพลิเคชัน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

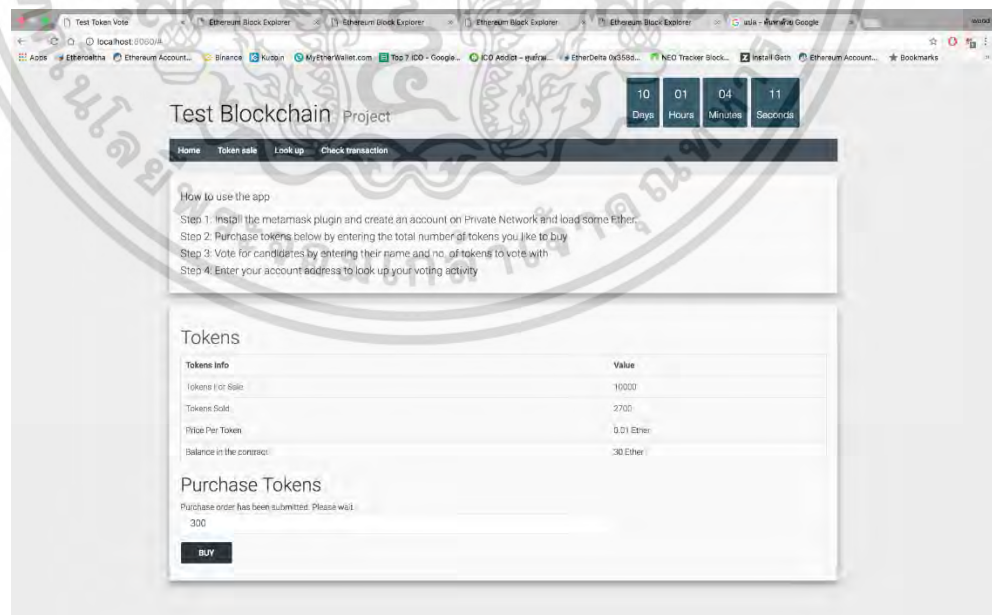
## 4.2 การเข้าสู่หน้าเมนู Token sale

เมื่อเข้าสู่หน้าเมนู Token sale จะแสดงหน้าต่างจำนวนของโทเคนทั้งหมด จำนวนของโทเคนที่มีการซื้อไปแล้ว จำนวนราคาต่อโทเคน จำนวนเงินที่ผู้จัดตั้งการลงคะแนนได้รับทั้งหมด และระบบการซื้อโทเคนที่ใช้ในการลงคะแนนเสียง ดังรูป 4.2



(รูป 4.2 หน้าเมนู Token sale)

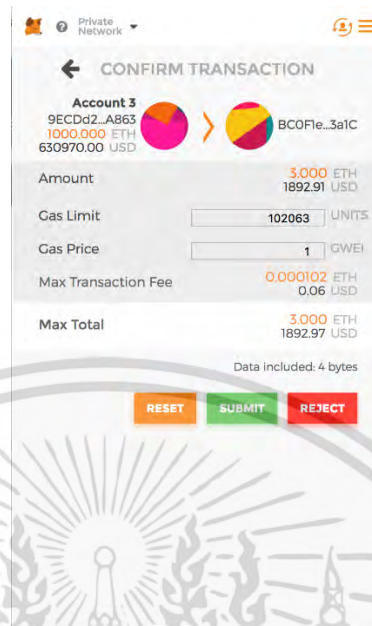
ขั้นตอนการซื้อโทเคนที่ใช้ในการลงคะแนนเสียงเพื่อนำไปลงคะแนนเสียง



(รูป 4.3 ขั้นตอนการซื้อโทเคน จำนวน 300 โทเคน)

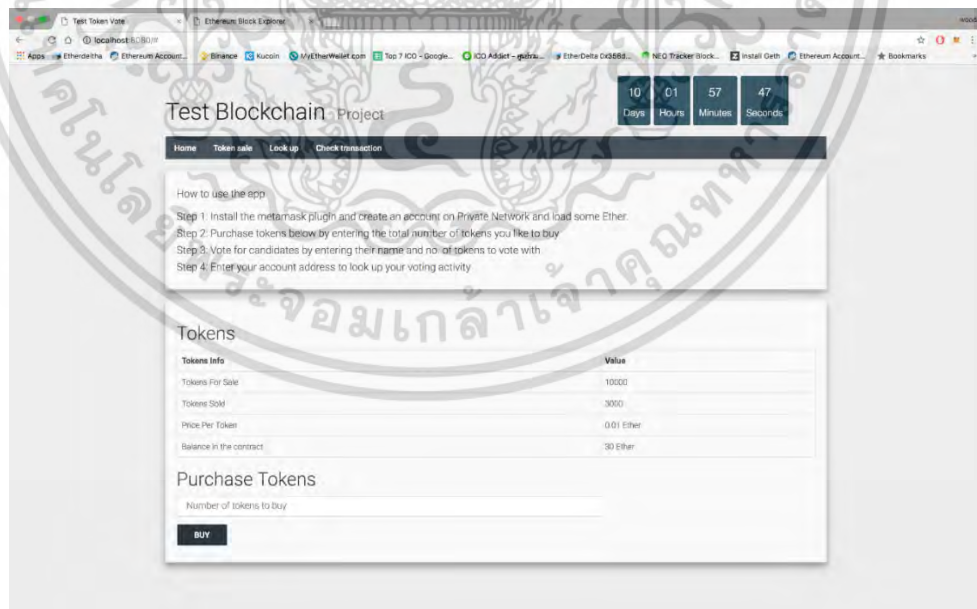
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ยืนยันการซื้อโทเค็นผ่าน Meatomask



(รูปที่ 4.4 ยืนยันการซื้อโทเค็นผ่าน Metamask)

แสดงหน้าโทเค็นที่มีการซื้อเพิ่มไปจำนวน 300 โทเค็น

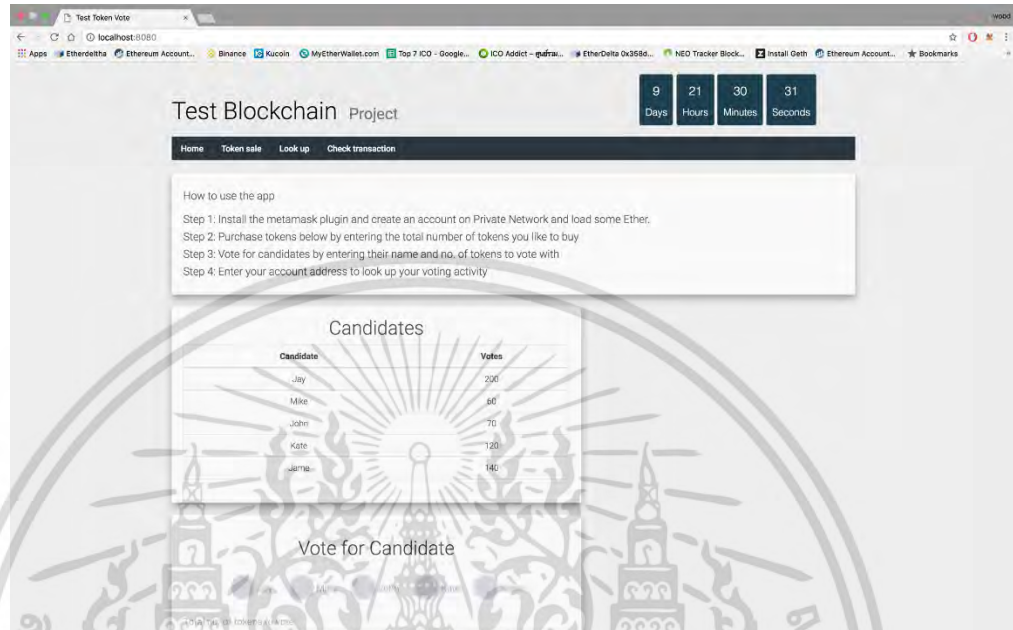


(รูปที่ 4.5 แสดงหน้าโทเค็นที่มีการซื้อเพิ่มไปจำนวน 300 โทเค็น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 การเข้าสู่หน้าเมนู Home ลงคะแนนเสียง

เมื่อเข้าสู่หน้าเมนู Home จะแสดงจำนวนผู้ลงสมัคร จำนวนคะแนนเสียงของผู้เข้าสมัครแต่ละคน และ ส่วนการลงคะแนนเสียงให้ผู้สมัคร ดังรูป 4.6



(รูปที่ 4.6 แสดงหน้า Home ส่วนการลงคะแนนเสียง)

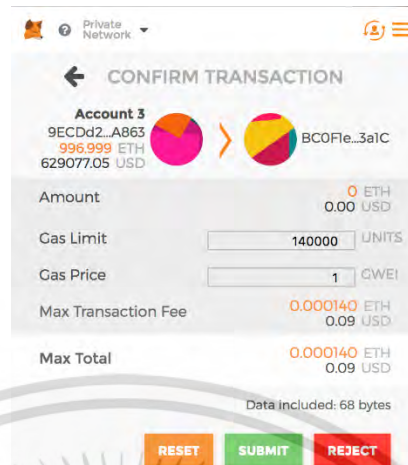
ขั้นตอนการลงคะแนนเสียงให้ผู้สมัคร



(รูปที่ 4.7 ลงคะแนนเสียงให้ผู้สมัคร Jay จำนวน 100 โทเค็น )

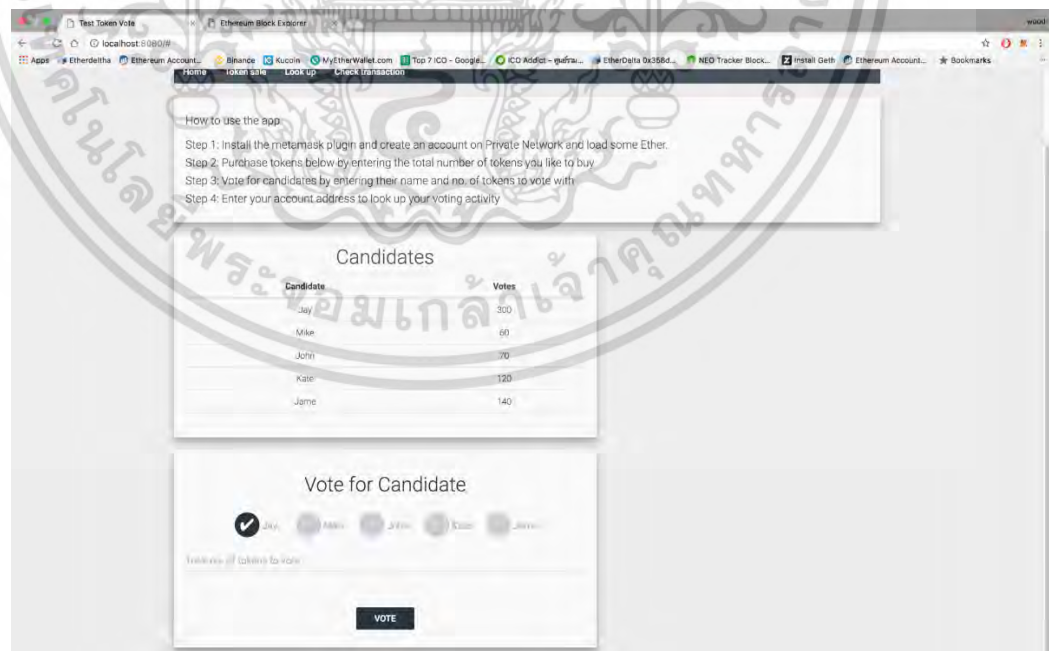
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ยืนยันการโหวตผ่าน Metamask



(รูปที่ 4.8 ยืนยันการลงคะแนนผ่าน Metamask)

แสดงหน้าการโหวต Jay ที่เพิ่มมาจำนวน 100 โทเค้น

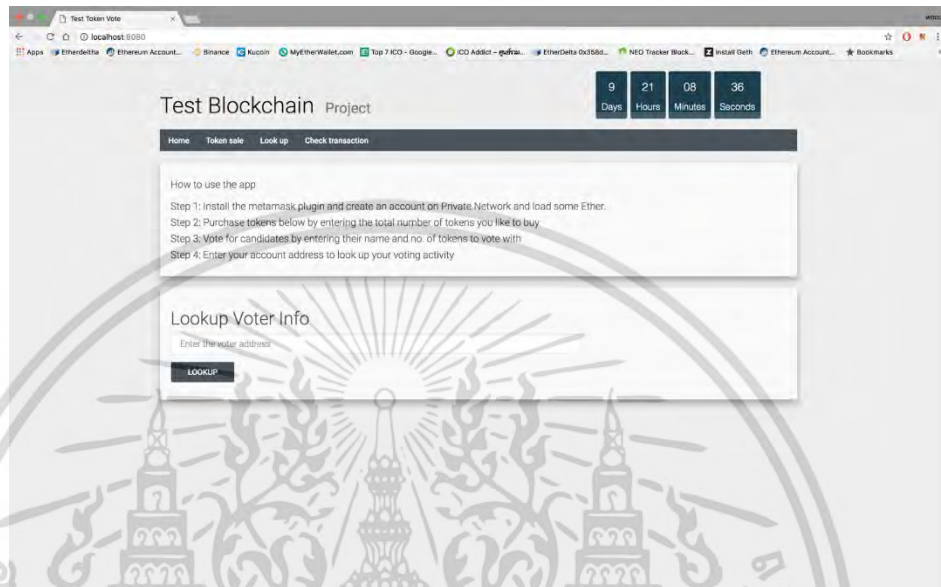


(รูปที่ 4.9 แสดงผลคะแนนของ Jay เพิ่มขึ้น 100 คะแนน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

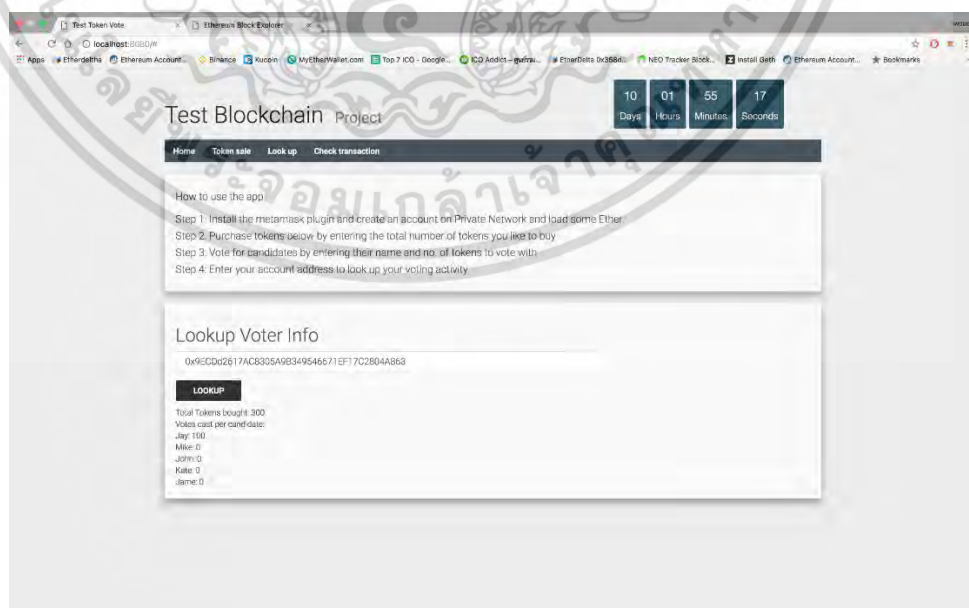
#### 4.4 การเข้าสู่หน้าเมนู Look up

เมื่อเข้าสู่หน้าเมนู Look up ภายในหน้าจะมีให้กรอกเลขกระเป๋าในช่อง Lookup Voter Info เพื่อตรวจสอบว่าเลขกระเป๋านั้นได้มีการซื้อ โทเค็นไปจำนวนเท่าไร และ ได้ลงคะแนนเสียงให้ใครจำนวนเท่าไร ดังรูป 4.10



(รูปที่ 4.10 แสดงหน้า Look up)

ขั้นตอนการ Look up เพื่อเช็คแอดเดสว่ามีกรซื้อโทเค็นและลงคะแนนเสียงไปเท่าไรโดยการกรอกเลขกระเป๋าเพื่อเช็ค



(รูปที่ 4.11 แสดงผล Look up)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้หน้าไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

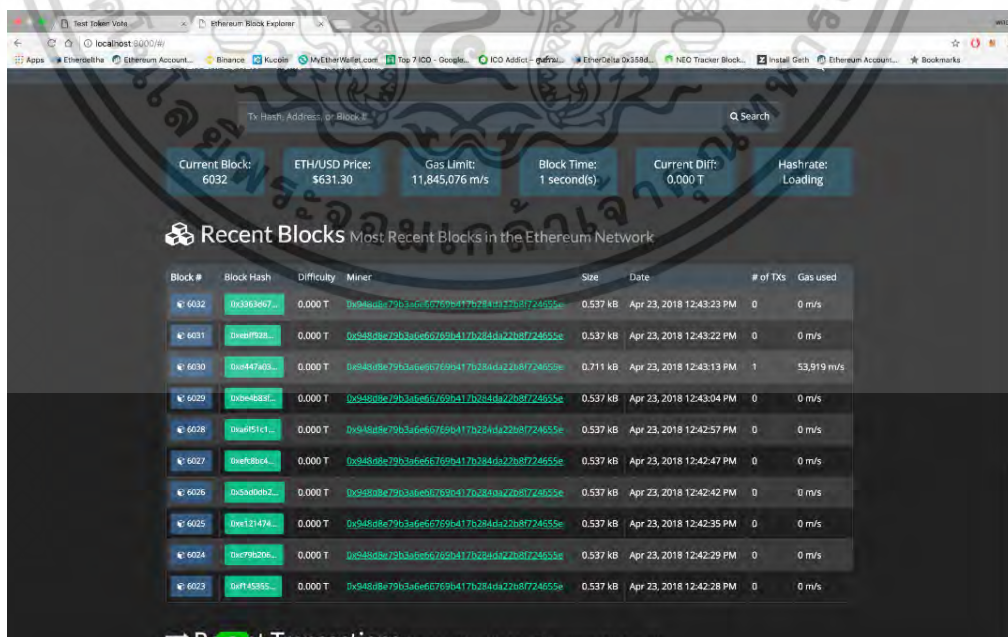
## 4.5 การเข้าสู่หน้าเมนู Check Transaction

ในส่วนของหน้า Check transaction ในหน้านี้จะเป็นส่วนในการแสดงผลทรานแซคชันทั้งหมดของบล็อกเชนที่ใช้ในแอปพลิเคชัน โดยภายในเมนู Check transaction เมื่อเปิดขึ้นมาจะได้หน้าดังรูปที่ 4.12 โดยจะมีเมนูหลักรวมกัน 2 เมนู คือ

1. Home
2. Blockchain Info

### 4.5.1 เมนู Home ของ Check transaction

จะประกอบไปด้วย ช่องค้นหาข้อมูลโดยสามารถใส่หาข้อมูล Tx Hash, Address or Block เพื่อทำการตรวจสอบข้อมูล ส่วนของหน้าบอกสถานะของบล็อกเชน ว่าปัจจุบันบล็อกเชนอยู่บล็อกที่เท่าไร ราคาอีเธอเมียมเทียบกับเยสดอลลาร์สหรัฐปัจจุบันราคาเท่าไร แก๊สลิ้มิตปัจจุบันใช้งานเท่าไร บล็อกทามล่าสุดกี่วินาที ค่าดีฟปัจจุบัน และค่าแฮชปัจจุบัน ส่วนของ Recent Blocks จะแสดงค่า Block เลขบล็อกเชน, Block Hash ค่าของฟังก์ชันแฮชของบล็อก, Difficulty ค่าความยากของฟังก์ชันแฮชในการคอนเฟิร์มทรานแซคชัน, Miner แสดงเลขกระเป๋าของผู้ที่สามารถคอนเฟิร์มทรานแซคชันได้, Size ขนาดของข้อมูลในบล็อกนั้น, Date เวลาที่บล็อกนั้นถูกคอนเฟิร์มทรานแซคชัน, # of TXs จำนวนทรานแซคชันที่มีการเคลื่อนไหวในบล็อกนั้น, Gas Used จำนวนค่าแก๊สที่ถูกใช้ทั้งหมดในทรานแซคชันนั้น โดยจะแสดง 10 บล็อกล่าสุดที่มีการประมวลผล และสามารถคลิก ที่Block, Block Hash, address miner เพื่อเช็คข้อมูลได้ ดังรูป 4.12 และสุดท้ายเป็นส่วน ของ Recent Transactions จะแสดงผลบล็อกที่มีทรานแซคชันที่เคลื่อนไหวในส่วนนี้ ดังรูป4.13

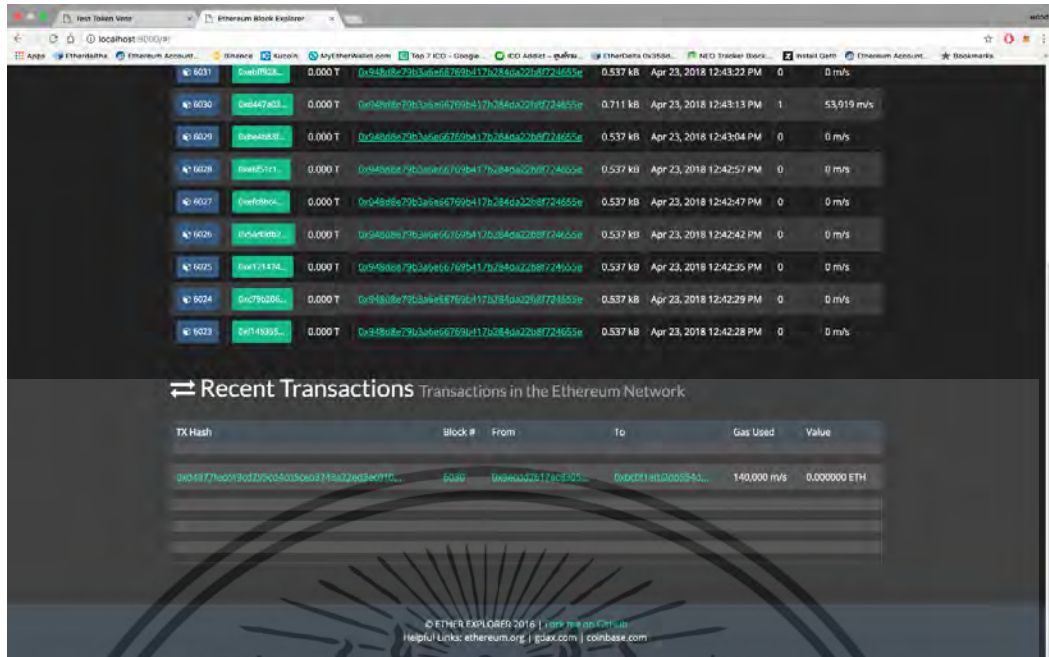


The screenshot shows the Ethereum Block Explorer interface. At the top, there are several key metrics: Current Block: 6032, ETH/USD Price: \$631.30, Gas Limit: 11,845,076 m/s, Block Time: 1 second(s), Current Diff: 0.000 T, and Hashrate: Loading. Below this is a section titled 'Recent Blocks' with a table listing the most recent blocks in the Ethereum network.

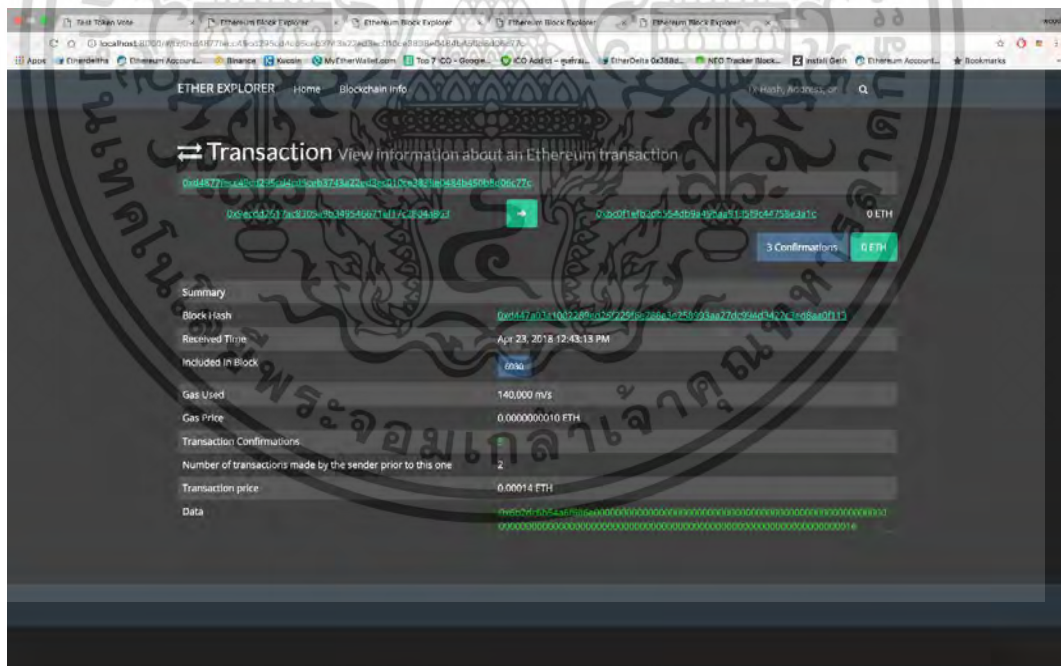
Block #	Block Hash	Difficulty	Miner	Size	Date	# of TXs	Gas used
6032	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:43:23 PM	0	0 m/s
6031	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:43:22 PM	0	0 m/s
6030	0x0447603...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.711 kB	Apr 23, 2018 12:43:13 PM	1	33,919 m/s
6029	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:43:04 PM	0	0 m/s
6028	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:42:57 PM	0	0 m/s
6027	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:42:47 PM	0	0 m/s
6026	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:42:42 PM	0	0 m/s
6025	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:42:35 PM	0	0 m/s
6024	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:42:29 PM	0	0 m/s
6023	0x0963607...	0.000 T	0x04a08e79b3a6e6769b417b2844a22b8724655e	0.537 kB	Apr 23, 2018 12:42:28 PM	0	0 m/s

(รูปที่ 4.12 แสดงหน้า Home ในส่วนของ Check transaction)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



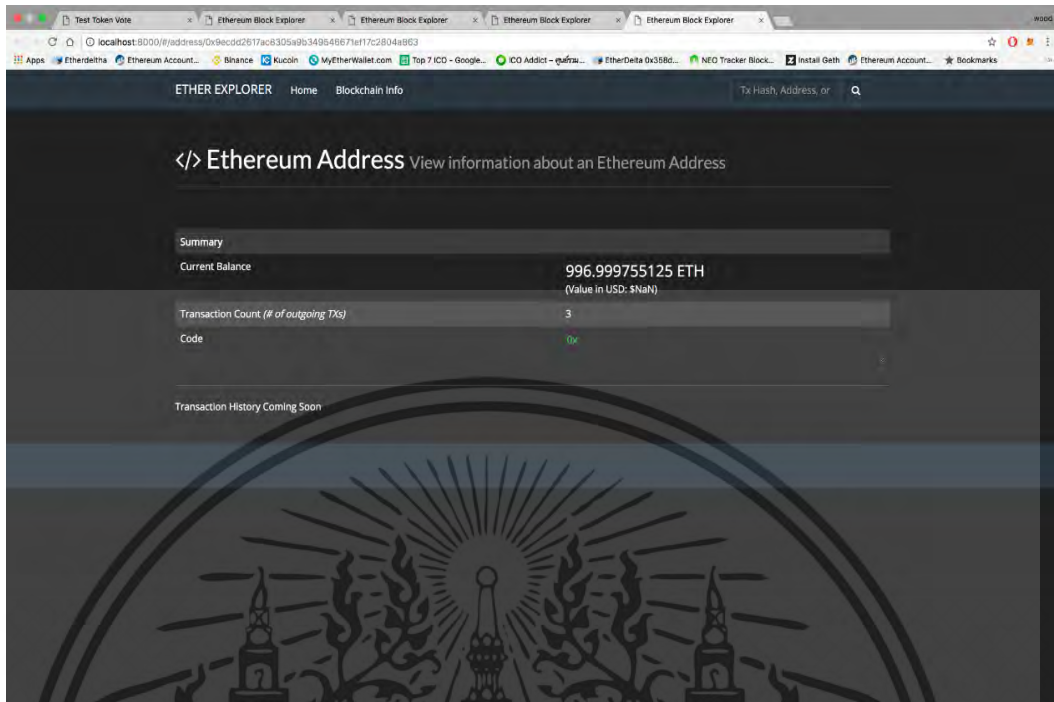
(รูปที่ 4.13 แสดงหน้า Home ในส่วนของ Check transaction(ต่อ))  
 แสดงผลของการช้ค้นหาค้นหาโดยการค้นหาค้นหาด้วย Tx Hash



(รูปที่ 4.14 แสดงผลการค้นหาค้นหาด้วย Tx Hash)

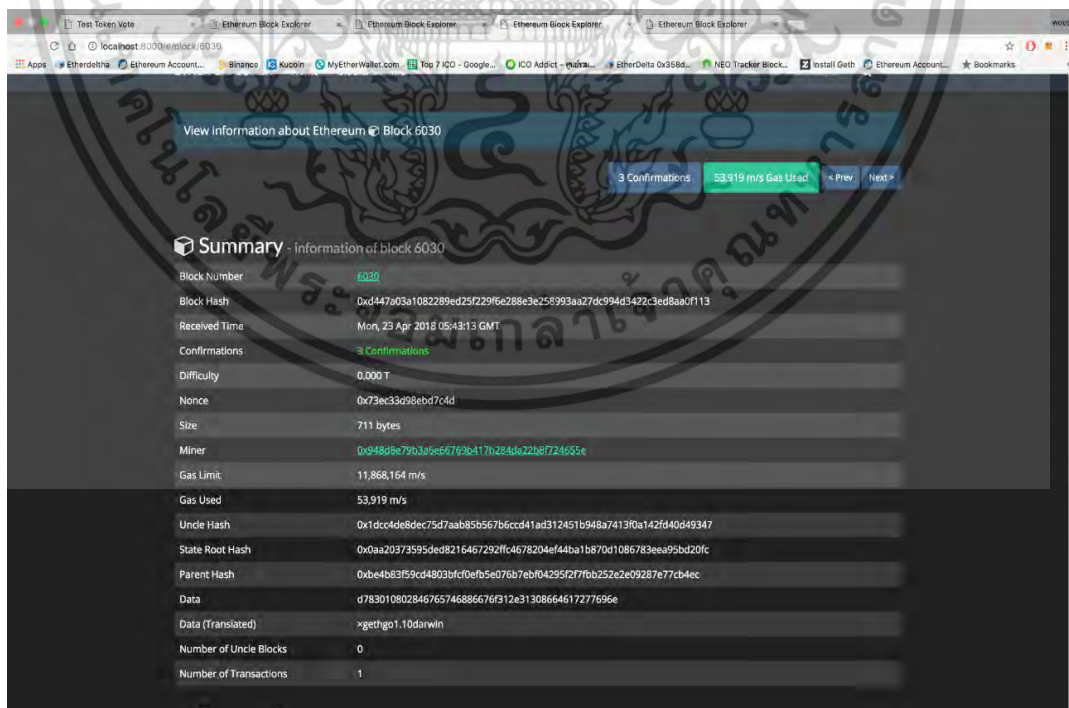
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## แสดงผลของการข่งค้นหาโดยการค้นหาด้วย Address



(รูปที่ 4.15 แสดงผลการค้นหาด้วย Address)

## แสดงผลของการข่งค้นหาโดยการค้นหาด้วย Block

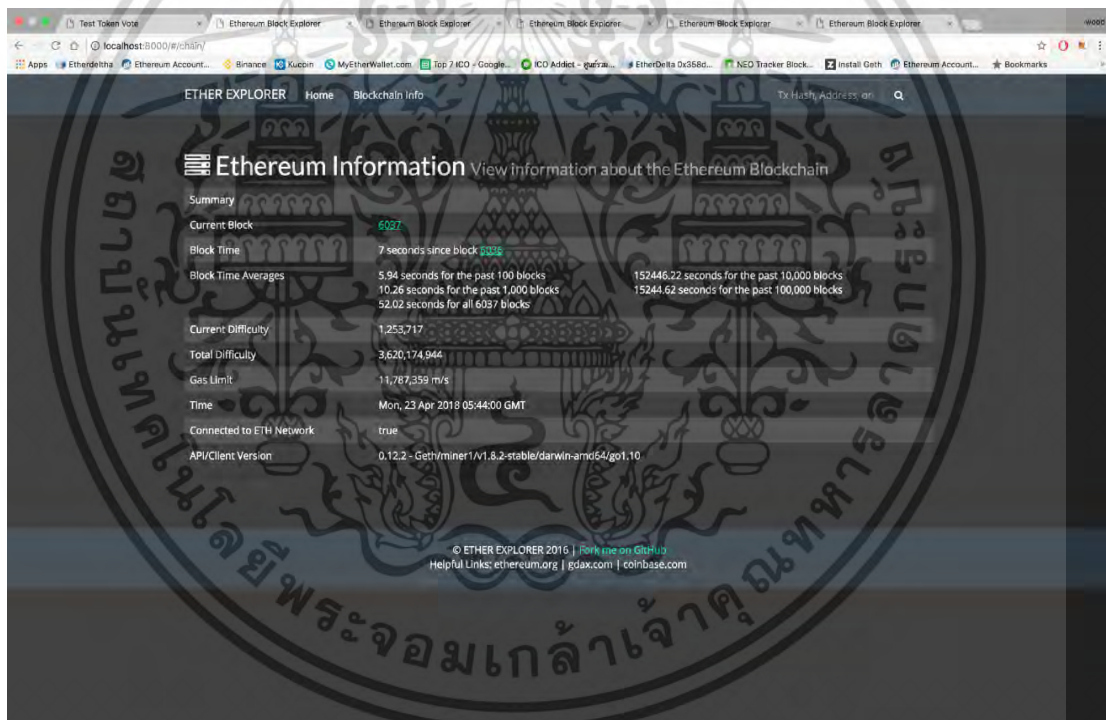


(รูปที่ 4.16 แสดงผลการค้นหาด้วย Block)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5.2 เมนู Blockchain Info ของ Check transaction

ในส่วนเมนู Blockchain Info จะแสดงข้อมูลอีเทอเรียมบนอีเทอเรียมบล็อกเชน โดยจะประกอบด้วย Current Block บล็อกล่าสุดที่ถูกคอนเฟิร์มทรานแซคชัน , Block Time จะแสดงเวลาของบล็อกที่ถูกคอนเฟิร์มทรานแซคชันล่าสุดเทียบกับบล็อกก่อนหน้า , Block Time Averages จะแสดงค่าเฉลี่ยของบล็อกที่มีการคอนเฟิร์มทรานแซคชัน เทียบต่อบล็อกล่าสุดจำนวน 100 บล็อก , 1000 บล็อก , 10000บล็อก , 100000บล็อก และ บล็อกทั้งหมดที่ถูกคอนเฟิร์มทรานแซคชันไปก่อนหน้านี้ตั้งแต่บล็อกที่ 1 , Current Difficulty แสดงค่าความยากในการขุดล่าสุดในการคอนเฟิร์มทรานแซคชัน , Total Difficulty แสดงค่าความยากทั้งหมดในการคอนเฟิร์มทรานแซคชัน , Gas Limit แสดงค่าแก๊สที่จำกัดไว้ , Time แสดงเวลาที่บล็อกล่าสุดคอนเฟิร์มทรานแซคชัน , Connected to ETH Network แสดงสถานะว่ามีการเชื่อมต่อกับอีเทอเรียมเน็ต , API/Client Version แสดงค่าซอฟต์แวร์ที่ใช้ในการเชื่อมต่อกับตัวเว็บแสดงผล ดังรูป 4.17



(รูปที่ 4.17 แสดงหน้า Blockchain Info ในส่วนของ Check transaction)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

# สรุปผลการดำเนินโครงการ

### 5.1 สรุปผลการทดลอง

โครงการระบบลงคะแนนเสียงแบบไม่มีศูนย์กลางบนระบบบล็อกเชน (Decentralized Voting Application Based On Blockchain System) ได้สร้างระบบบล็อกเชนส่วนตัว และเว็บแอปพลิเคชัน เพื่อเปลี่ยนแปลงรูปแบบการเลือกตั้งลงคะแนนเสียงเลือกตั้ง หรือการลงคะแนนเสียงในรูปแบบรายการทีวี โดยเทคโนโลยีบล็อกเชน เป็นเทคโนโลยีที่มีการพูดถึงอย่างมากในปัจจุบัน นั้นมีความปลอดภัย โปร่งใส ตรวจสอบได้ และสามารถเข้าถึงได้ง่าย ความสามารถเหล่านี้ เป็นสิ่งที่ควรจะนำไปใช้ในการเลือกตั้ง

การลงคะแนนเสียงนั้น จะผ่านหน้าเว็บแอปพลิเคชัน และบล็อกเชนส่วนตัวนั้น จะเป็นผู้ตรวจสอบคะแนนของการลงคะแนนเสียง นั้นว่ามีผู้ใดลงให้แก่ใคร ข้อมูลเหล่านี้จะถูกคำนวณประมวลผลเพื่อถอดรหัส หากมีใครพยายามเปลี่ยนแปลงข้อมูล บล็อกเชนจะทำการตรวจสอบ และแก้ไขให้เป็นเหมือนเดิมภายในเสี้ยววินาที หากได้นำไปใช้ในการเลือกตั้ง จะทำให้การเลือกตั้งนั้นมีประสิทธิภาพอย่างมาก เพราะทุกคนสามารถช่วยระบบตรวจสอบความถูกต้องของทรานแซคชันได้

บล็อกเชนส่วนตัวนั้นยังมีปัญหาด้านความล่าช้าหากเกิดการคำนวณประมวลผลข้อมูลจำนวนมากๆ เพราะยังเป็นแบบส่วนตัว และบล็อกเชนส่วนตัวนี้ยังสามารถใช้ได้แค่บุคคลที่ลงซอฟต์แวร์ของบล็อกเชนตัวนี้เท่านั้น แต่กระนั้นยังเป็นผลดีที่ทำให้เข้าใจระบบการทำงานอย่างถ่องแท้มากขึ้น และเพื่อเป็นทักษะในการพัฒนาต่อยอดลงบล็อกเชนเส้นหลักเพื่อใช้ในการใช้งานแบบเปิดกว้างต่อหลากหลายบุคคลในอนาคต

### 5.2 ปัญหาและอุปสรรคของการดำเนินโครงการ

#### 5.2.1 แหล่งที่มาอ้างอิงข้อมูลยังไม่เพียงพอ

- แหล่งข้อมูลหรือบทความทางวิชาการยังไม่ถูกแพร่หลายทุกฉบับในออนไลน์ ทำให้ในช่วงแรกนั้นหาแนวทางการพัฒนาโครงการค่อนข้างยาก ทำให้เสียเวลาในการทดลองชุดคำสั่งโปรแกรมหลายๆรูปแบบ กว่าที่จะเข้าใจและประสบผลสำเร็จ

#### 5.2.2 ราสเบอร์รี่พายไม่สามารถทำการคำนวณประมวลผลได้

- เนื่องจากความสามารถของซอฟต์แวร์ราสเบอร์รี่พายนั้นยังไม่รองรับการคำนวณประมวลผลของบล็อกเชน แต่แค่สามารถใช้เป็นโหนดเพื่อรับ ส่งและเรียกธุรกรรมได้ จึงเลือกใช้แค่คอมพิวเตอร์นำมาประมวลผลอย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษา ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.3 แนวทางการพัฒนาโครงการ

#### 5.3.1 นำไปพัฒนาต่อบนบล็อกเชนหลัก

- เนื่องจากบล็อกเชนส่วนตัวยังไม่สามารถเปิดใช้ให้คนส่วนใหญ่ได้ หากได้มีการนำไปพัฒนาเพื่อใช้บนบล็อกเชนเส้นหลัก จะทำให้คนสามารถเข้ามาใช้งานได้อย่างง่ายขึ้น

#### 5.3.2 เพิ่มการใช้งานรูปแบบการลงคะแนนเสียง

- เพิ่มการลงคะแนนเสียงให้หลากหลายรูปแบบ ไม่ว่าจะเป็นแบบ 1 คน 1 โหวต หรือแบบแข่งขันกันตามกำลังทรัพย์ สามารถตั้งเวลาการลงคะแนนเสียงหมดอายุได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เอกสารอ้างอิง

- [1] Don Tapscott, Alex Tapscott. **Blockchain Revolution**. New York : An imprint of Penguin Random House. 2016
- [2] Chris Dannen. **Introducing Ethereum and Solidity**. Brooklyn, New York, USA : Apress Media. 2017
- [3] Read the Docs. **Ethereum Homestead Documentation**. 2017 Available: [www.ethdocs.org/](http://www.ethdocs.org/)
- [4] Read the Docs. **Solidity**. 2017. Available : <http://solidity.readthedocs.io/en/>
- [5] Said Eloudrhiri. **Set up the Private chain**. 10 March 2017. Available : <http://chainskills.com/2017/03/10/part-3-setup-the-private-chain-miners/>
- [6] Tomer Gabbai. **Web developer welcome to Ethereum**. Oct 24 2017. Available : <https://medium.com/wemark-stories/web-developer-welcome-to-ethereum-ff0355379025>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

Poster

Department of Computer Engineering  
(Information Engineering)  
CE 6070



## Decentralized Voting Application Based On Blockchain System (ระบบลงคะแนนเสียงแบบไม่มีศูนย์กลางบนระบบบล็อกเชน)

Griengsak Kumnodnae, Tirajad Sirimueanphong and Asst.Prof.Dr. Kitdakorn Klomkarn

---

**Abstract**

This thesis was created with the aim to develop a centralization change to de-centralization system. By adopting blockchain technology to create private blockchain and decentralize voting application. Whether voting Democratic. Or vote for contestants on the TV show must be verify and high security with the blockchain system. The private blockchain system created from Ethereum platform called "go-ethereum", used for generate transection and confirm transection caused by exchange cryptocurrency for voting candidate .

**Introduction**

Nowadays, social networks talking about bitcoin very much .It is a digital currency that will be used instead of money or to use speculation like stock market. But hardly anyone talking about the technology behind that has highly security ,reliability and verified . Many companies started to use blockchain technology to do various projects. It's not just a financial but blockchain can do such a variety such as fintech,internet of thing, education, music streaming, insurance etc. All of them are intended to change from Centralization to de-centralization system to prevent cheating, hacking, not transparent. And can verified. However, it is a new technology that was talking about the world, we have chosen to build de-centralized voting application.

**Methodology**



Deployed Voting Contract

**Results**





- The process of buying a token for voting.
- Voting procedure
- Steps to check the number of token purchased and voted with publickey.

Lookup Voter Info

Public Key: 0x00

Private Key: 0x00

Address: 0x00

Balance: 0 ETH

This website used for voting by blockchain operation data collection and management. Users must log in through the metamask. Metamask is make connection with blockchain and wallet for manage transactions that users purchase , voting and check the number of token in wallet.

**Conclusion**

This thesis has changed from simple voting from centralization to de-centralization system can vote on mainnet or testnet Ethereum Blockchain. Application can vote on the mainnet , testnet Ethereum Blockchain and private Ethereum blockchain. Financial transactions using google extension named Metamask to help voting easier. We simulated to vote contestant by voters can use as much as no limits. The system works expeditiously and without error, and in the future will be developed to form democratic voting is more convenient.

**References**

- [1]Chris Dannen. (2017),Introducing Ethereum and Solidity,
- [2]Solidity,[online] Available: <https://solidity.readthedocs.io>
- [3]Ethereum Homestead Documentation,[online] Available: [www.ethdocs.org/](http://www.ethdocs.org/)



kkkitdak@kmitl.ac.th



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข  
ตัวอย่าง การติดตั้ง Go - Ethereum

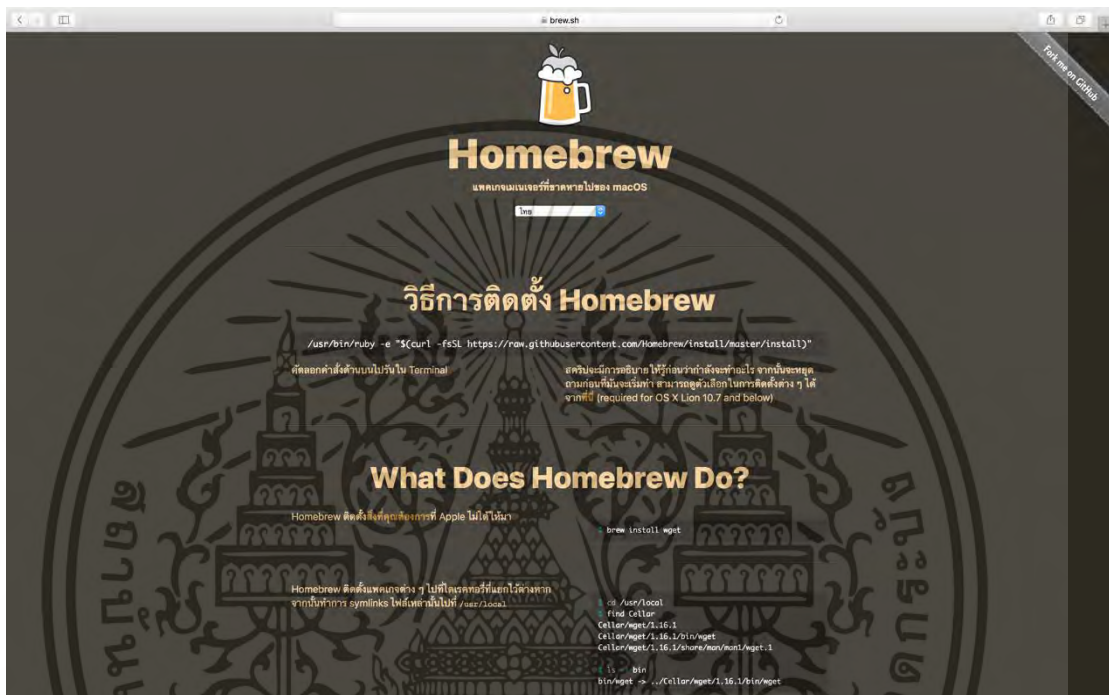
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ข

### ตัวอย่าง การติดตั้ง Go - Ethereum

#### การติดตั้งแพ็คเกจเมนเจอร์ Homebrew

1. เข้าหน้าเว็บไซต์ [https://brew.sh/index\\_th](https://brew.sh/index_th)



รูปที่ ข.1 แสดงหน้าเว็บไซต์ Homebrew

2. เปิดหน้าต่าง Terminal แล้วคัดลอกคำสั่งบนหน้าเว็บไซต์ เพื่อติดตั้งแพ็คเกจเมนเจอร์ Homebrew ดังนี้

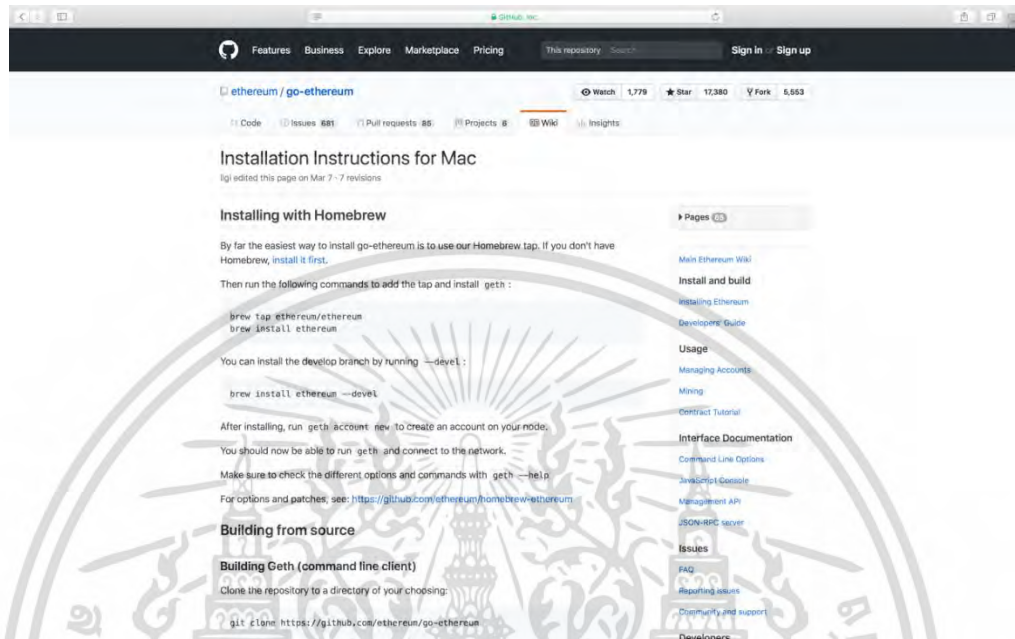
```
usr/bin/ruby -e "$(curl -fsSL
```

```
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อทำการลง Homebrew เรียบร้อย ต่อไปจะเป็นการลง Go - Ethereum โดยเข้าไปที่หน้าเว็บไซต์

<https://github.com/ethereum/go-ethereum/wiki/Installation-Instructions-for-Mac>



รูปที่ ข.2 แสดงหน้าเว็บไซต์ github ของ Go-ethereum

4. เปิดหน้า Terminal แล้วคัดลอกคำสั่งบนหน้าเว็บไซต์ เพื่อติดตั้ง Go - Ethereum ดังนี้
- ```
brew tap ethereum/ethereum
brew install ethereum
```

รูปที่ ข.3 แสดงหน้าTerminal ติดตั้ง Go-ethereum

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



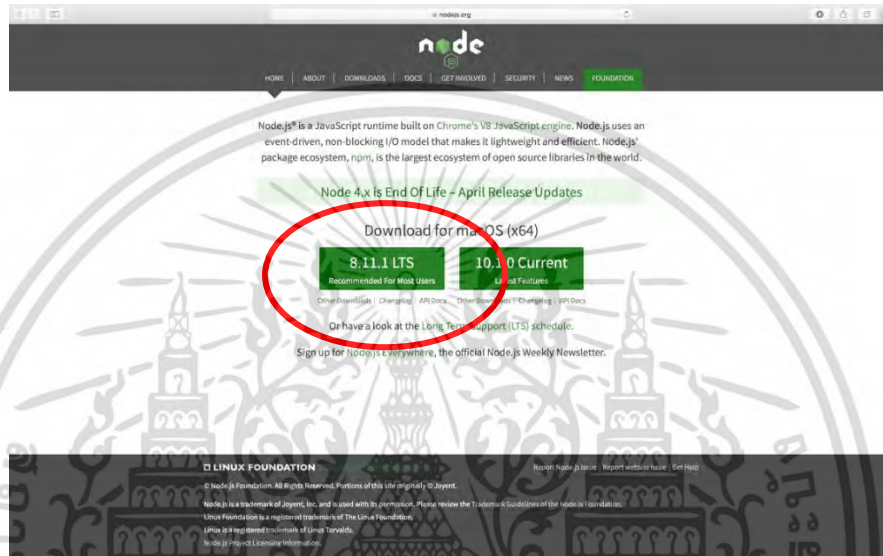
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ค

### ตัวอย่าง การติดตั้ง Node.js

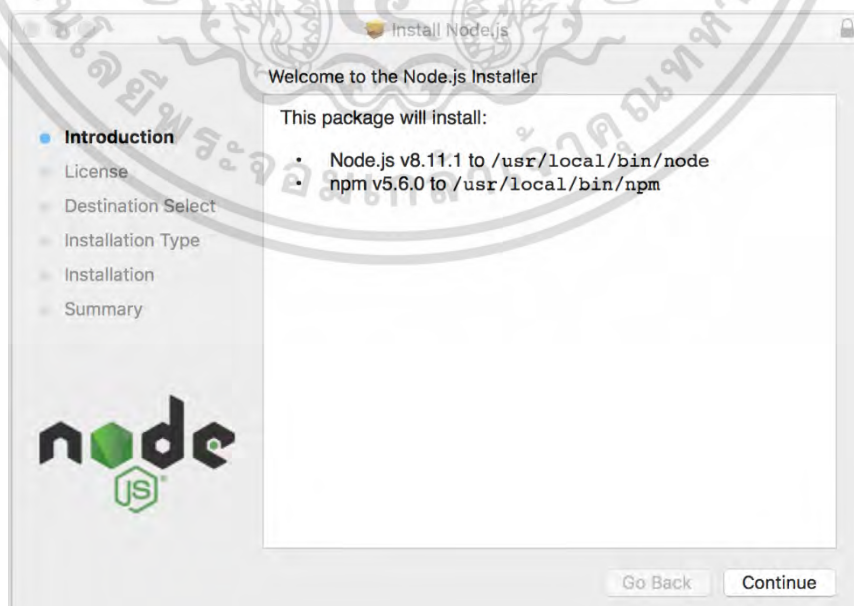
#### การติดตั้ง Node.js

1. เข้าหน้าเว็บไซต์ <https://nodejs.org/en/> และเลือกดาวน์โหลด ตามภาพ



รูปที่ ค.1 แสดงหน้าเว็บ Node.js

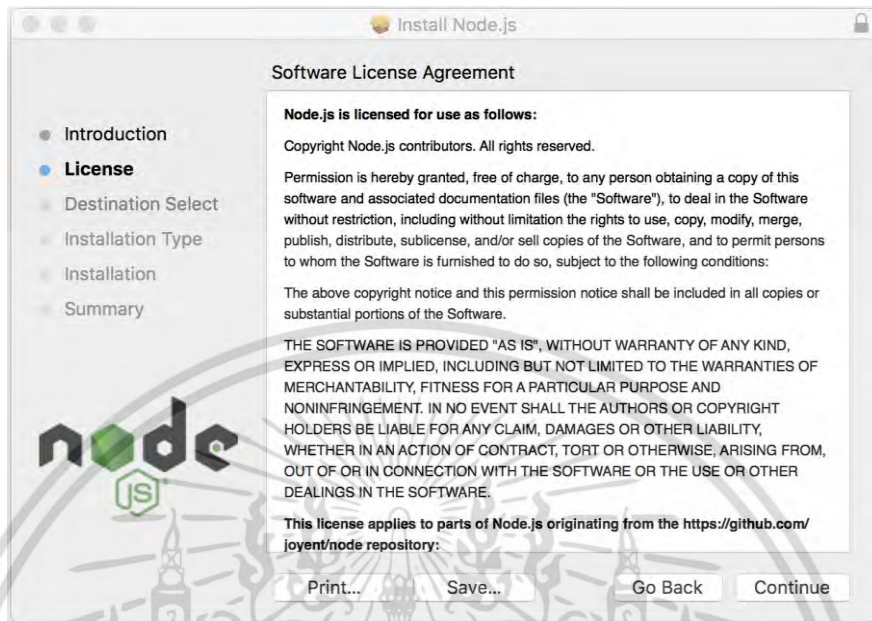
2. เปิดไฟล์ที่ดาวน์โหลด โดยในหน้าแรกจะเป็นการอธิบายถึงซอฟต์แวร์ที่จะติดตั้ง



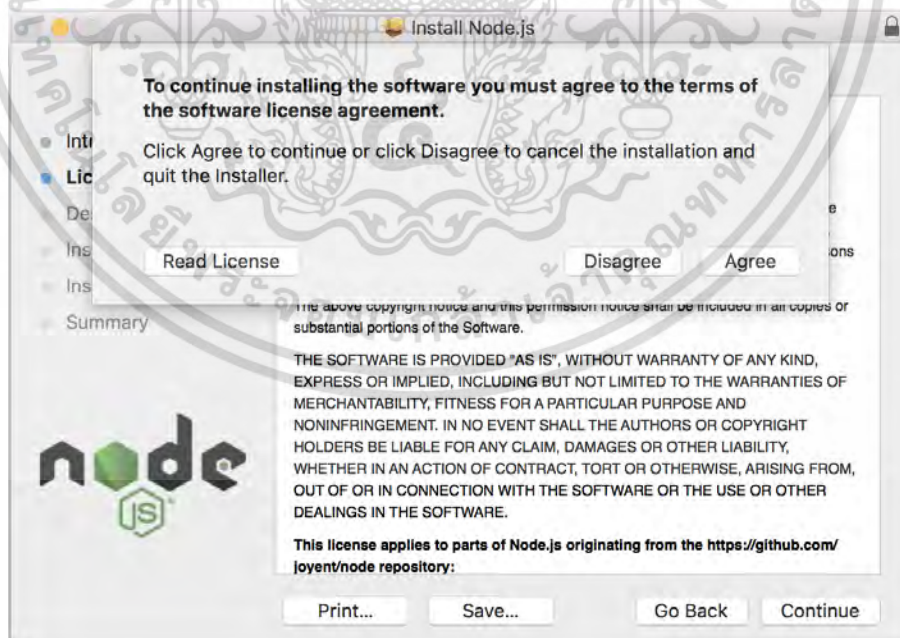
รูปที่ ค.2 หน้าต่างอธิบาย Node.js ที่ติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้เชิง นเพื่อการศึกษาเท่านั้น เมื่อผู้ผู้ใดเห็นนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การติดตั้งซอฟต์แวร์หน้าที่ 2 ต้องอ่านลิขสิทธิ์ และยอมรับข้อกำหนดการใช้งานของผู้พัฒนา



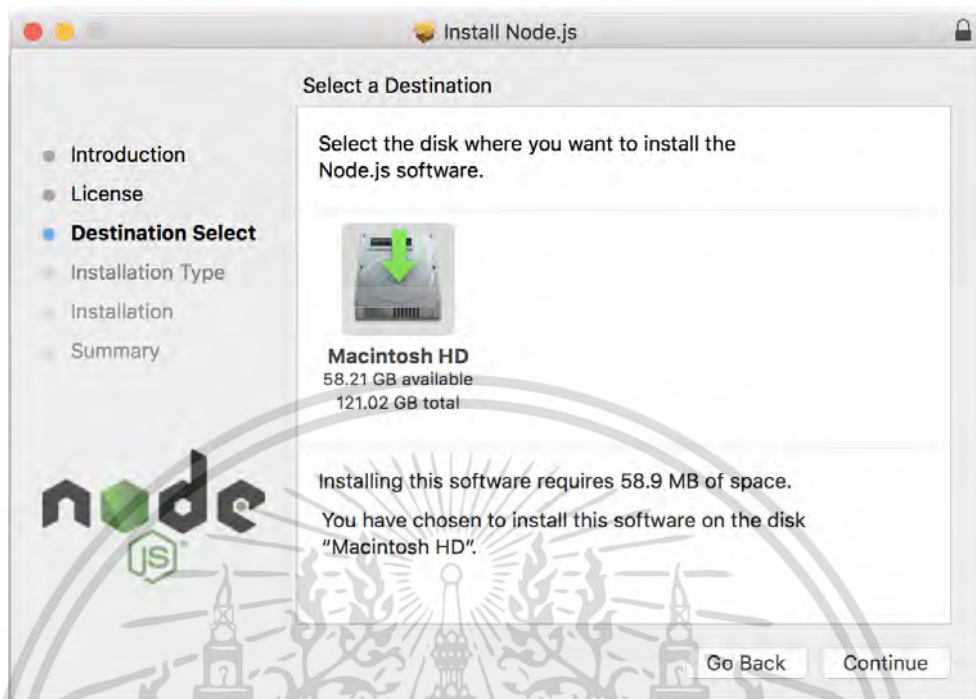
รูปที่ ค.3 หน้าต่างเงื่อนไข และการยอมรับการใช้งานของ Node.js  
 ในรูปแรก นั้นทำการกด Continue และ ทำการ Agree ตามภาพที่ 2



รูปที่ ค.4 กดยอมรับเงื่อนไขของ Node.js

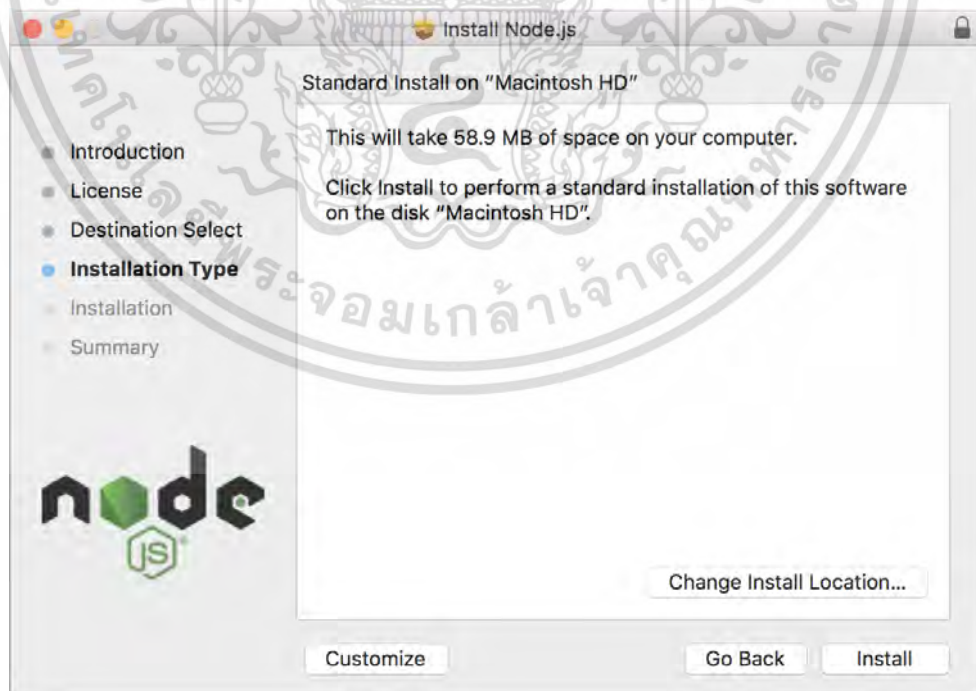
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ทำการเลือกไดรฟ์ที่ต้องการจัดเก็บข้อมูล แล้วกดปุ่ม Continue



รูปที่ ค.5 เลือกไดรฟ์ในการเก็บข้อมูล

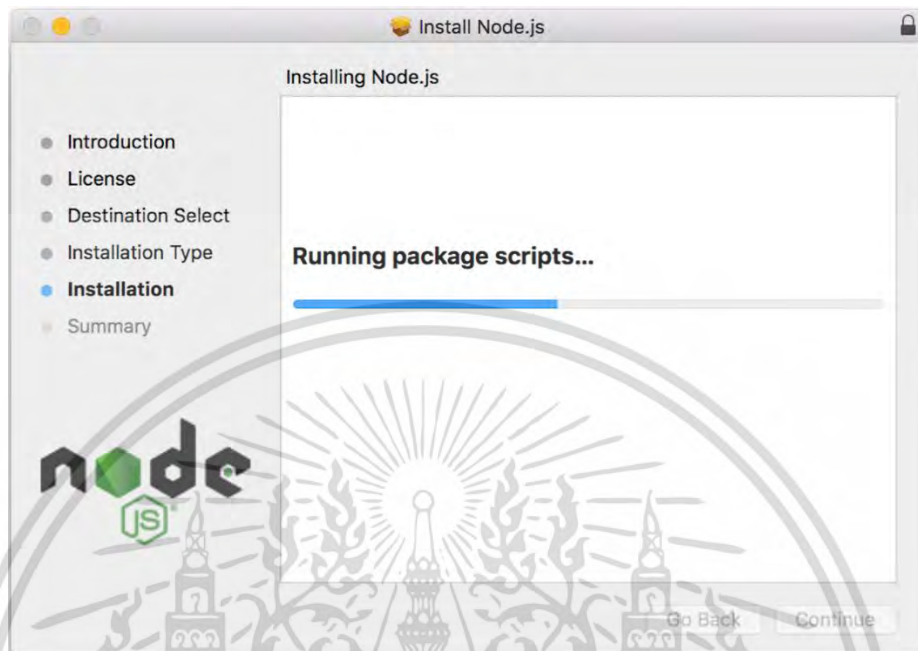
5. ทำการยืนยันพื้นที่ ที่ซอฟต์แวร์ต้องการใช้ในการติดตั้ง และกด Install



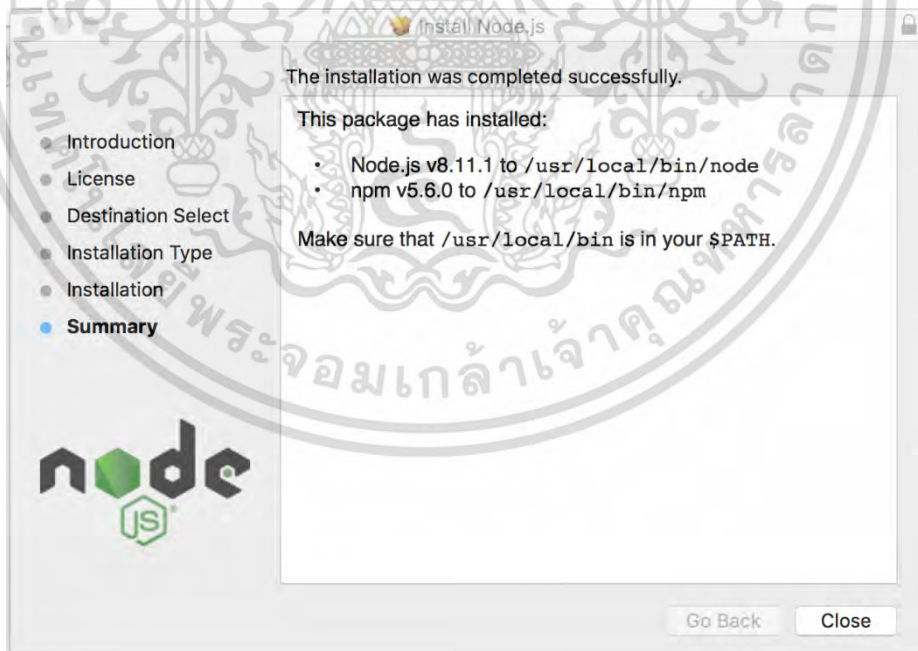
รูปที่ ค.6 ยืนยันการเลือกไดรฟ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. รอกการดำเนินการของการติดตั้งซอฟต์แวร์ และเมื่อเสร็จสมบูรณ์ดังรูปที่ 2 กด Close เป็นการเสร็จสมบูรณ์



ภาพที่ ค.7 รอกการติดตั้ง Node.js



ภาพที่ ค.8 แสดงการติดตั้ง Node.js เสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ง  
ตัวอย่าง การติดตั้ง Truffle Framework

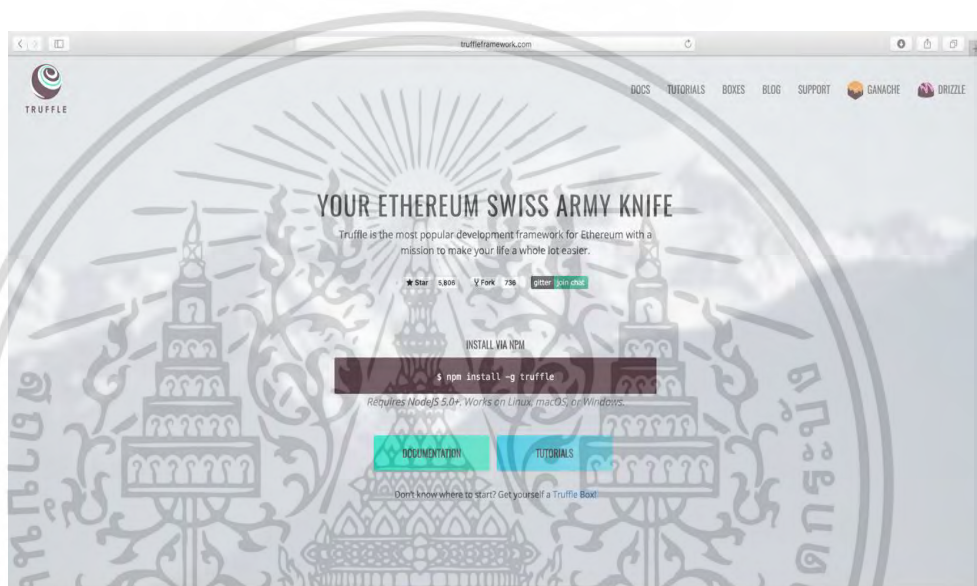
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ง

### ตัวอย่าง ตัวอย่าง การติดตั้ง Truffle Framework

#### การติดตั้ง Truffle

1. เข้าเว็บไซต์ <http://truffleframework.com> และคัดลอกคำสั่ง



ภาพที่ 1 แสดงหน้าเว็บไซต์ Truffle Framework

2. เปิด Terminal และวางคำสั่ง เพื่อติดตั้ง

```
tonmaiwood — npm TERM_PROGRAM=Apple_Terminal SHELL=/b...
Last login: Wed May 9 20:01:56 on ttys000
[Griengsaks-MacBook-Pro:~ tonmaiwood$ npm install -g truffle
( ) :: fetchMetadata: sill resolveWithNewModule has-flag@1.0.0
```

ภาพที่ 2 แสดงคำสั่งและการติดตั้ง Truffle Framwork

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



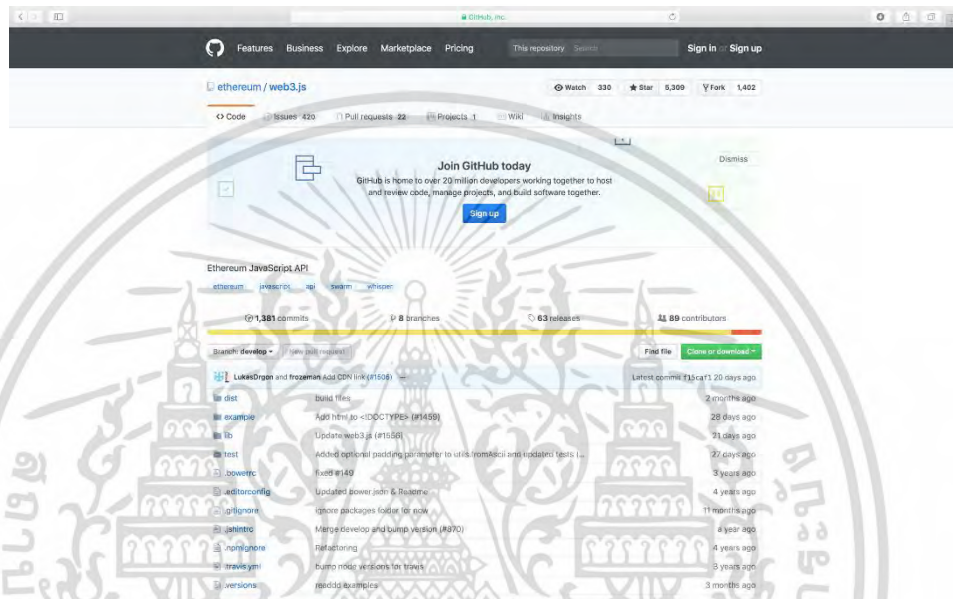
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก จ

### ตัวอย่าง การติดตั้ง Web3.JS

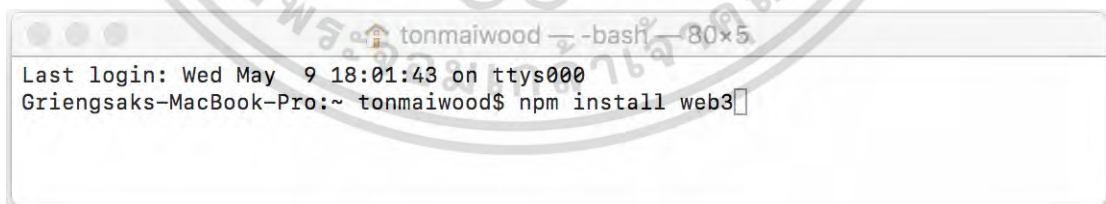
#### การติดตั้ง Web3.JS

1. เข้าเว็บไซต์ <https://github.com/ethereum/web3.js/> เพื่อเช็คเวอร์ชันล่าสุดของ Web3.JS



รูปที่ จ.1 เว็บไซต์ตัวอย่างไฟล์ Web3.JS

2. เปิด Terminal ใช้คำสั่ง `npm install web3` เพื่อติดตั้ง Web3.JS



รูปที่ จ.2 แสดงคำสั่ง `npm install web3`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. Terminalเมื่อติดตั้ง Web3.JS เสร็จสมบูรณ์

```

nodetest — -bash — 80x26
CXX(target) Release/obj.target/bufferutil/src/bufferutil.o
SOLINK_MODULE(target) Release/bufferutil.node
CXX(target) Release/obj.target/validation/src/validation.o
SOLINK_MODULE(target) Release/validation.node
npm WARN saveError ENOENT: no such file or directory, open '/Users/tonmaiwood/Desktop/nodetest/package.json'
npm NOTICE created a lockfile as package-lock.json. You should commit this file.
npm WARN enoent ENOENT: no such file or directory, open '/Users/tonmaiwood/Desktop/nodetest/package.json'
npm WARN nodetest No description
npm WARN nodetest No repository field.
npm WARN nodetest No README data
npm WARN nodetest No license field.

+ web3@1.0.0-beta.34
added 305 packages in 56.789s

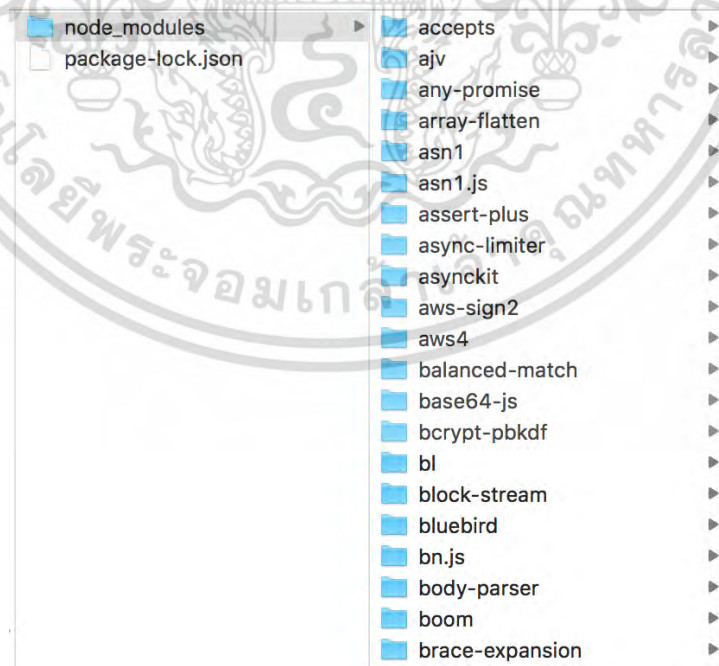
Update available 5.6.0 → 6.0.0
Run npm i -g npm to update

Griengsaks-MacBook-Pro:nodetest tonmaiwood$

```

รูปที่ จ.3 แสดงการติดตั้ง Web3.JS เสร็จสมบูรณ์

### 4. แสดงไฟล์ในโฟลเดอร์ที่ได้จากการติดตั้ง Web3.JS



รูปที่ จ.4 แสดงโฟลเดอร์ ที่ติดตั้ง Web3.JS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ฉ  
ตัวอย่าง การติดตั้ง Sublime Text

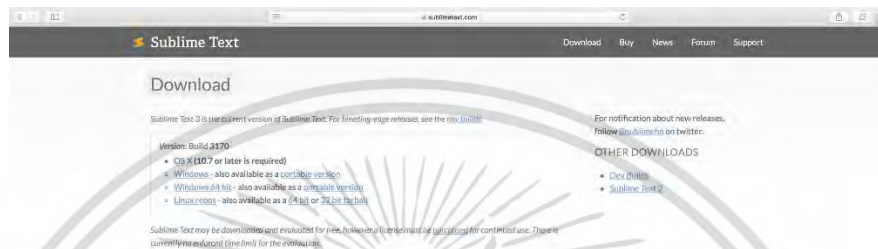
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ฉ

### ตัวอย่าง การติดตั้ง Sublime Text

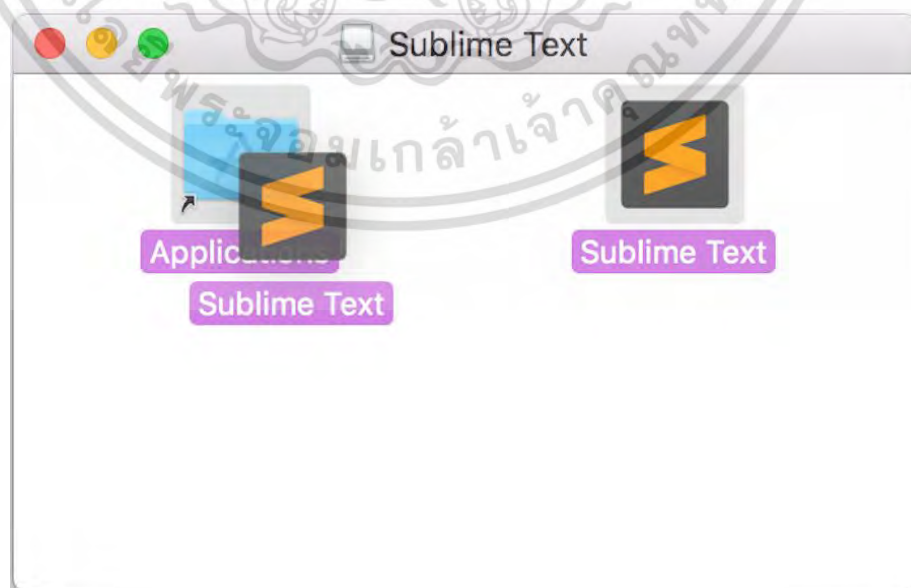
#### การติดตั้ง Sublime Text

1. เข้าสู่หน้าเว็บไซต์ <https://www.sublimetext.com/3> เลือกระบบปฏิบัติการ



รูปที่ ฉ.1 เว็บไซต์ของโปรแกรม Sublime Text

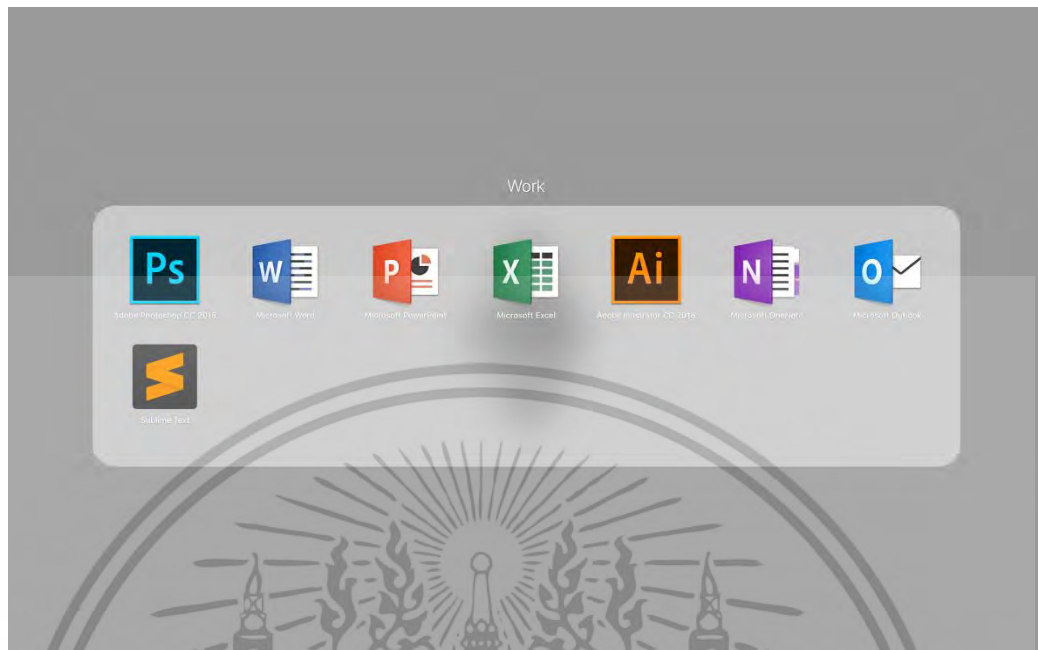
2. ลาก Sublime Text.dmg ลงในโฟลเดอร์ Application



รูปที่ ฉ.2 แสดงการลาก Sublime Text ลงในโฟลเดอร์ Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. หน้าแสดงการติดตั้ง Sublime Text เสร็จสมบูรณ์



รูปที่ ๑.3 แสดงการติดตั้ง Sublime Text เสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



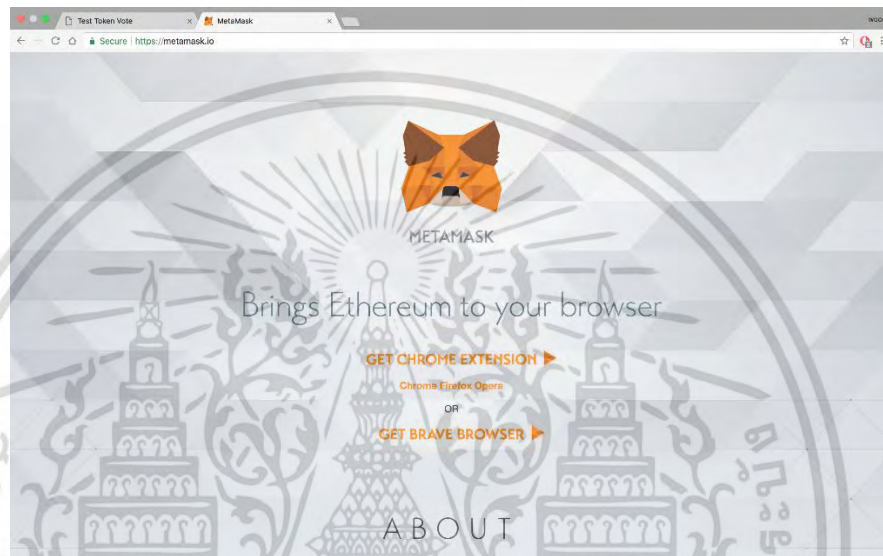
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ข

### ตัวอย่าง การติดตั้งส่วนขยายเบราว์เซอร์ Metamask

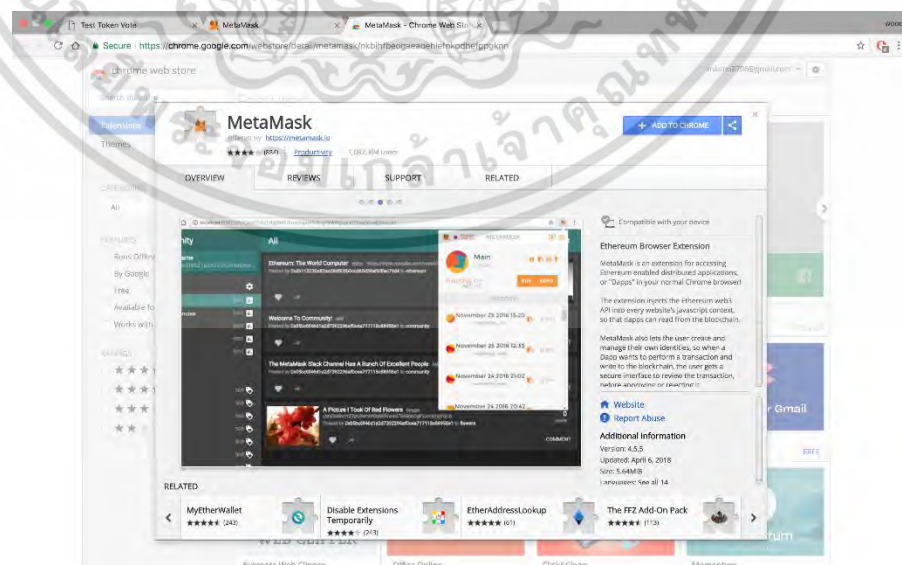
#### การติดตั้งส่วนขยายเบราว์เซอร์ Metamask

1. เข้าเว็บไซต์ <https://metamask.io> และ กดที่ GET CHROME EXTENSION



รูปที่ ข.1 เว็บไซต์ของส่วนขยายเบราว์เซอร์ Metamask

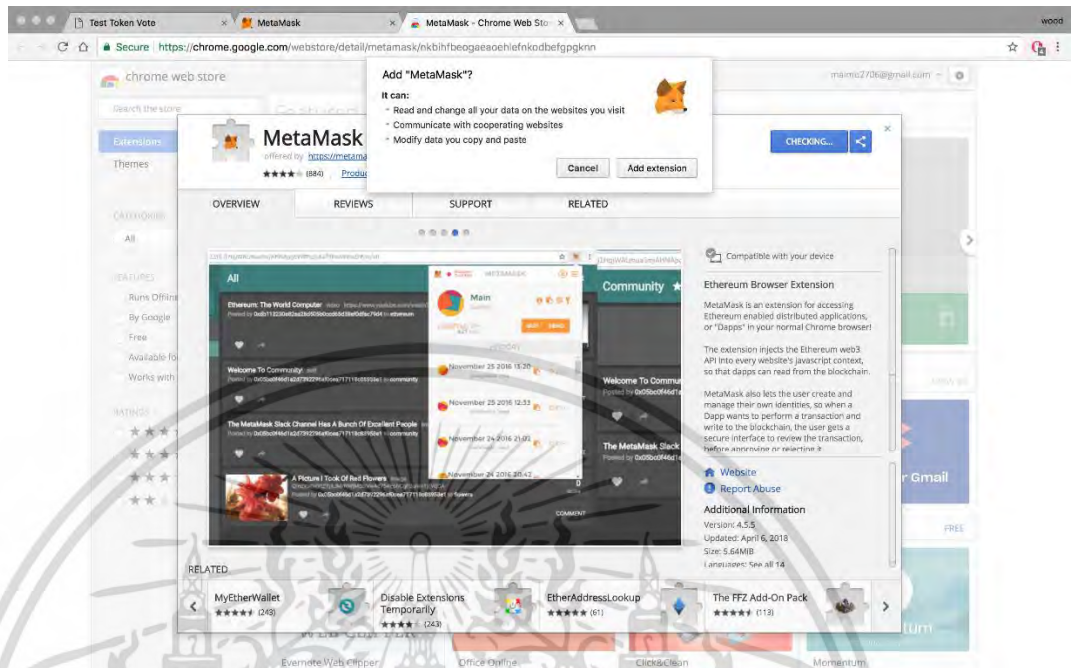
2. กดที่ ADD TO CHROME เพื่อติดตั้ง Metamask



รูปที่ ข.2 หน้าติดตั้งของส่วนขยายเบราว์เซอร์ Metamask

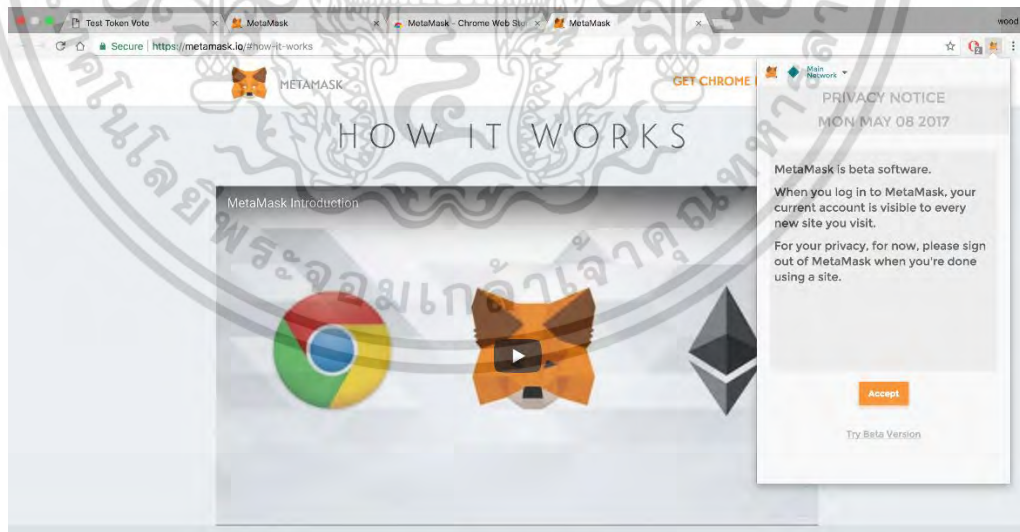
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออยู่ภายใต้เงื่อนไขใบอนุญาตการใช้งาน  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. กดที่ Add extension เพื่อยืนยันการติดตั้งอีกครั้ง



รูปที่ ข.3 หน้ายืนยันการติดตั้งของส่วนขยายเบราว์เซอร์ Metamask

### 4. หน้าแสดงการติดตั้ง Metamask เสร็จสมบูรณ์

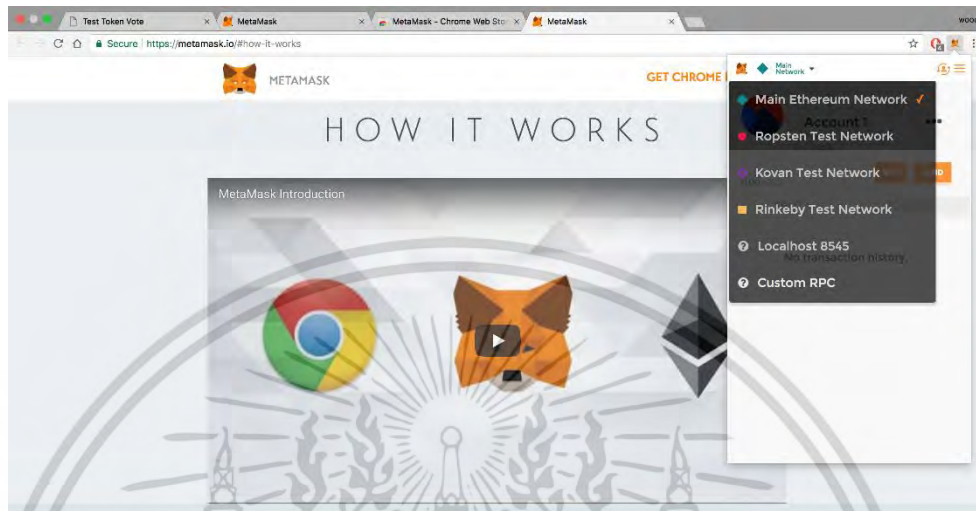


รูปที่ ข.4 หน้าแสดงการติดตั้ง Metamask เสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

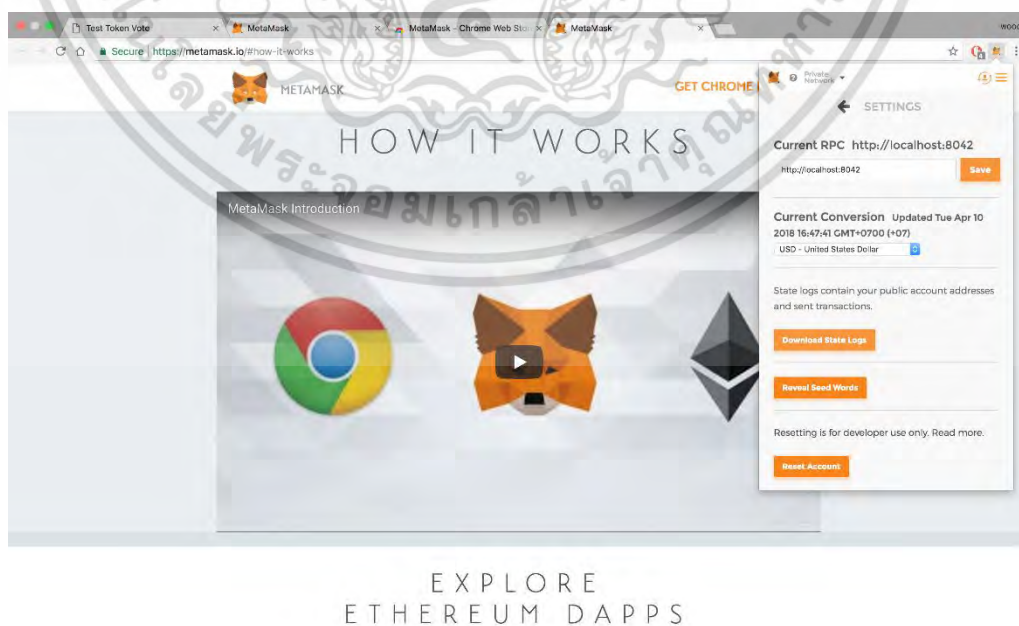
## การเชื่อมต่อ Metamask กับ บล็อกเชนส่วนตัว

1. หน้าแสดงการเชื่อมต่อ Metamask กับบล็อกเชนส่วนตัว เลือก Custom RPC



รูปที่ ข.5 หน้าแสดงการ Metamask เข้ากับบล็อกเชนส่วนตัว

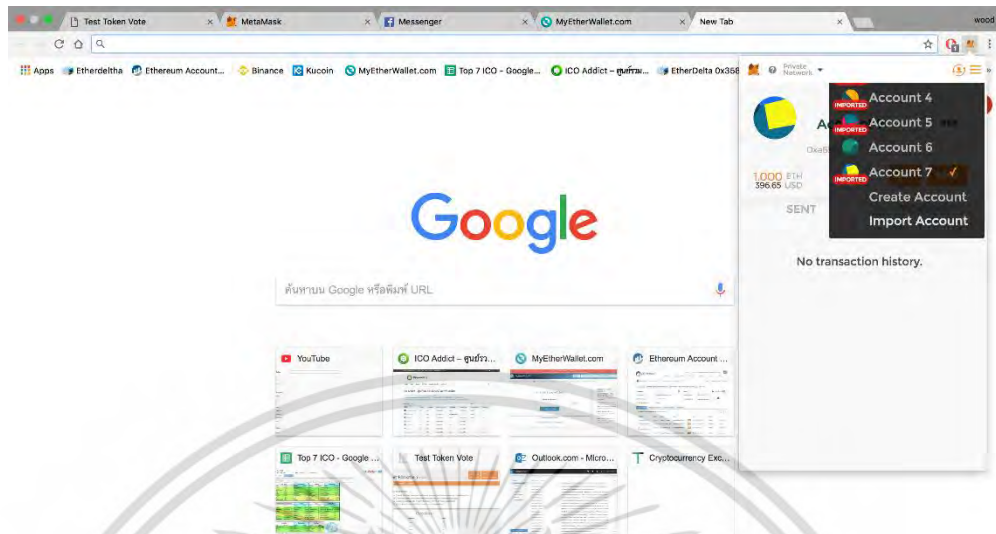
2. กรอก <http://localhost:8042> ให้ตรงกับพอร์ตที่ตั้งค่าไว้ในบล็อกเชนส่วนตัว และกด save



รูปที่ ข.6 หน้าแสดงการแก้ไขพอร์ตให้ตรงกับบล็อกเชนส่วนตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. แสดงการเชื่อมต่อ Metamask กับบล็อกเชนส่วนตัวเสร็จสมบูรณ์



รูปที่ ข.7 หน้าแสดงการเชื่อมต่อ Metamask กับบล็อกเชนส่วนตัวเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ประวัติคณะผู้จัดทำโครงการคนที่ 1

ชื่อผู้จัดทำ นายเกรียงศักดิ์ กำหนดแน  
ที่อยู่ 495 ถ.โกสุมรวมใจ แขวงดอนเมือง เขตดอนเมือง  
กรุงเทพฯ 10210  
โทร 092-602-0130  
Email: tu\_tutintom@hotmail.com

ประวัติการศึกษา ปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมสารสนเทศ  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2560

ตำแหน่ง/  
สถานที่ทำงาน -



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติคณะผู้จัดทำโครงการคนที่ 2

ชื่อผู้จัดทำ นายถิรเจตน์ สิริเหมือนพงศ์  
ที่อยู่ บ้านเลขที่ 52 หัวหมาก31 หัวหมาก บางกะปิ  
กรุงเทพ 10240  
โทร 085-116-8858 , 02-300-0824  
Email : Tirajad\_J@hotmail.com

ประวัติการศึกษา ปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมสารสนเทศ  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2560

ตำแหน่ง/  
สถานที่ทำงาน -



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้