วีดีกรี: ระบบตรวจสอบใบประมวลผลการศึกษาดิจิทัลด้วยเทคโนโลยีบล็อกเชน

VDegree: Verifiable Credential-Based Digital Transcript Verification System Using Blockchain Technology

นายธนกฤต บรรจงศิลป์

Thanagrit Banjongsilp

นายปรินทร ปาณฑิตยางกูร

Parintorn Panditiyangkun

นายประณิธาน วิทยารณยุทธ

Pranitan Wittayaronnayutt

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

มหาวิทยาลัยศรีนครินทรวิโรฒ ปีการศึกษา 2563

คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ชื่อหัวข้อโครงงาน	วีดีกรี: ระบบตรวจสอบใบประมวลผลการศึกษาดิจิทัลด้วยเทคโนโลยีบล็อกเชน VDegree: Verifiable Credential-based Digital Transcript Verification System Using Blockchain Technology		
นิสิต	นายธนกฤต	บรรจงศิลป์	60102010333
	นายปรินทร	ปาณฑิตยางกูร	60102010339
	นายประณิธาน	วิทยารณยุทธ	60102010567
ปริญญา	วิทยาศาสตร์บัณฑิต (วท.บ.)		
ภาควิชา	วิทยาการคอมพิวเตอร์		
อาจารย์ที่ปรึกษาโครงงาน	ผศ.ดร.จันตรี ผลประเสริฐ		

ลงชื่อ.....

(ผศ.ดร.จันตรี ผลประเสริฐ)

อาจารย์ที่ปรึกษาโครงงาน

บทคัดย่อ

ปัจจุบันปัญหาการปลอมแปลง Transcript ยังเกิดขึ้นทั่วโลกและสร้างความเสียหายเป็นวงกว้าง เนื่องจากขาดกระบวนการตรวจสอบที่เหมาะสม และเครื่องมือในการปลอมแปลง Transcript สามารถเข้าถึงได้ง่าย อีกทั้งกระบวนการขอ Transcript จากสถาบันการศึกษาหลายแห่งนั้นช้าและไม่มีประสิทธิภาพ ในงานวิจัยนี้ได้ทำการพัฒนาระบบการออก Digital transcript ในรูปแบบ proof-of-concept (PoC) ที่มีชื่อว่า VDegree ระบบนี้ช่วยป้องกันการปลอมแปลง Transcript ในรูปแบบ Digital โดยใช้เทคโนโลยี Blockchain ซึ่งจะเก็บ Transcript ที่ถูกเข้ารหัสไว้ที่อุปกรณ์ของนักศึกษา และระบบนี้จะช่วยให้นักศึกษาสามารถเข้าถึง จัดการ ตรวจสอบ และส่ง Transcript ได้ง่าย กระบวนการออก การขอ และการตรวจสอบ Transcript มีวัตถุประสงค์เพื่อใช้ในการจำลองเท่านั้น และในส่วนของ Hyperledger Fabric blockchain ได้ทำการพัฒนาและทำการทดสอบบน AWS-managed blockchain ทางคณะผู้จัดทำได้ทำการทดสอบประสิทธิภาพของระบบใน AWS-managed blockchain โดยใช้ bc.m5.xlarge instance type - 4 vCPU, RAM 16 GB และ CouchDB เป็น State database จากผลการทดลองแสดงให้เห็นว่า ระบบนี้สามารถประมวลผลได้ประมาณ 3 Transactions/วินาที และยังแสดงให้เห็นถึงประสิทธิภาพของการลดขั้นตอนการขอ Transcript และประสิทธิภาพของ Blockchain ได้เป็นอย่างดี

Abstract

At present, fake transcripts are considered one of the most damaging problems globally due to lack of appropriate inspection processes and easy-to-access tools to generate counterfeit certificates. In addition, the conventional process to request transcripts from several academic institutions is slow and inefficient. In this work, we investigate a proof-of-concept (PoC) of the online digital transcript service called VDegree. With VDegree, the system provides a tamper-proof transcript in digital format using blockchain technology. By storing student's encrypted academic transcript in the user's devices, this system helps students easily access, share, manage and verify their transcript. The process of issuing, requesting or verifying the transcript is proposed in the PoC and a Hyperledger Fabric blockchain is developed and tested in the AWS-managed blockchain. We test the performance of our proposed system in the AWS-managed blockchain using bc.m5.xlarge instance type - 4 vCPU, 16 GB of RAM and CouchDB as State Database. Simulation results show that our proposed system can process approximately 3 transactions per second over a range of number of transactions. The results show promising potential both in the reduction of process and the performance of blockchain for the proposed use case.

กิตติกรรมประกาศ

ขอขอบคุณ ผศ.ดร.จันตรี ผลประเสริฐ อาจารย์ที่ปรึกษาและอาจารย์ประจำภาควิชาวิทยาการคอมพิวเตอร์ ผู้คอยให้คำแนะนำ ข้อคิดเห็น และสนับสนุนการทำโครงงาน ซึ่งทำให้โครงงานฉบับนี้สามารถสำเร็จลุล่วงไปได้ด้วยดี

ขอขอบคุณนายจักรวาล องค์ทองคำ ผู้คอยให้คำปรึกษาและคำแนะนำที่เป็นประโยชน์ต่อการทำโครงงานและการนำเสนอโครงงาน

ขอขอบคุณภาควิชาวิทยาการคอมพิวเคอร์ คณะวิทยาศาสตร์ สำหรับเครื่องมือ อุปกรณ์ต่าง ๆ และสถานที่ที่ใช้ในการทำโครงงาน รวมไปถึงอาจารย์ประจำภาควิชาวิทยาการคอมพิวเตอร์ทุก ๆ ท่าน ที่คอยให้ความรู้ คำแนะนำ และสั่งสอนจนทำให้สามารถจัดทำโครงงานนี้ขึ้นมาได้อย่างมีประสิทธิภาพตามที่คาดการณ์ไว้

ขอขอบคุณมหาวิทยาลัยศรีนครินทรวิโรฒ ซึ่งเป็นสถานที่ที่ให้วิชาความรู้จนสามารถนำความรู้ที่ได้มาจัดทำโครงงานฉบับนี้ขึ้นได้ อีกทั้งยังให้สถานที่ที่ใช้ในการทำโครงงาน

สุดท้ายนี้ ขอขอบคุณทุก ๆ ท่านที่มีความเกี่ยวข้องกับโครงงานฉบับนี้ ไม่ว่าจะให้ความรู้ คำปรึกษา คำแนะนำ ข้อคิดเห็น หรือความช่วยเหลือในด้านต่าง ๆ ทำให้โครงงานฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ทั้งนี้ ผู้จัดทำหวังว่าโครงงาน วีดีกรี: ระบบตรวจสอบใบประมวลผลการศึกษาดิจิทัลด้วยเทคโนโลยีบล็อกเชน (VDegree: Verifiable Credential-Based Digital Transcript Verification System Using Blockchain Technology) จะเป็นประโยชน์ต่อผู้อ่านและผู้ที่ต้องการนำโครงงานนี้ไปต่อยอดในอนาคต

สารบัญ

หัวข้อ **หน้า**

สารบัญรูปภาพ

สารบัญตาราง

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงาน

ในปัจจุบันการใช้ Transcript ในการสมัครงานมีความน่าเชื่อถือค่อนข้างน้อย เนื่องจาก Transcript สามารถปลอมแปลงได้ง่าย และทางบริษัทแทบจะไม่สามารถตรวจสอบได้เลยว่าเอกสารนั้นเป็นของจริงหรือถูกปลอมแปลงมา ถึงแม้ว่า Transcript ของสถาบันการศึกษาจะมีการประทับตรายาง ลายมือชื่อของอาจารย์ การทำตัวนูน (Emboss) สัญลักษณ์ของสถาบันตรงลายมือชื่อ และเนื้อกระดาษที่มีลักษณะเฉพาะแล้วก็ตาม ตัวอย่างเช่น วิทยาลัย Catawba ถูกปลอมเอกสารทั้ง Transcript และวุฒิการศึกษา แต่ที่แย่กว่านั้น คือ Catawba เป็นวิทยาลัยทางการแพทย์และอาจส่งผลให้ผู้คนเป็นอันตรายถึงแก่ชีวิตได้ถ้ากรณีที่บุคคลที่ปลอมแปลงเอกสารเหล่านี้สามารถนำไปสมัครและเข้าทำงานได้จริง [1] และไม่ใช่แค่ในต่างประเทศเท่านั้น ในปี พ.ศ. 2560 ได้มีการตรวจพบว่ามีผู้ใช้วุฒิการศึกษาปลอมมากถึง 10 คน เพื่อนำมาสมัครเข้าสอนในมหาวิทยาลัยชื่อดังแห่งหนึ่ง โดยมีนักศึกษาที่จบปริญญาโทและปริญญาเอกไปแล้ว 2 - 3 รุ่น [2]

จากตัวอย่างที่กล่าวมา การปลอมวุฒิการศึกษาถือเป็นปัญหาระดับโลก เพราะไม่ได้เป็นปัญหาที่พบแค่ในประเทศไทยแต่พบได้ในหลายประเทศทั่วโลก หากผู้ที่มีวุฒิการศึกษาไม่ได้มีความรู้ความสามารถที่เหมาะสม อาจทำให้บุคคลเหล่านั้นไม่สามารถทำงานในตำแหน่งหน้าที่นั้น ๆ ได้อย่างเหมาะสม และอาจจะส่งผลต่อภาพรวมของการทำงานในองค์กรได้ รวมไปถึงกระบวนการในการขอ Transcript ต้องเสียค่าธรรมเนียม ทางผู้จัดทำจึงมีแนวคิดที่จะเปลี่ยนจาก Transcript แบบธรรมดาเป็น Digital Transcript ซึ่งจะต้องใช้สิ่งที่เรียกว่า Verifiable Credential (VC) เข้ามาช่วย

VC เป็นชื่อเรียกของ Credential ที่เหมือนกับเอกสารในรูปแบบของกระดาษ เพียงแต่ถูกนำเข้าสู่ระบบดิจิทัล และจะถูกลงลายมือชื่อไว้ด้วย Digital Signature (ลายมือชื่อดิจิทัล) ทำให้มีความน่าเชื่อถือและไม่สามารถถูกปลอมแปลงได้โดยง่ายเหมือนกับเอกสารกระดาษ และการที่จะทำให้ VC มีความน่าเชื่อถือต้องอาศัย Certificate Authority (CA) ซึ่งเป็นองค์กรที่น่าเชื่อถือ ทำหน้าที่เป็นบุคคลที่สามในการดำเนินการออกใบรับรองดิจิทัลให้กับผู้ที่ขอใช้บริการในการทำ Key Management ซึ่งจะนำมาใช้ในการการยืนยันว่าข้อมูลนี้เป็นข้อมูลของนักศึกษาในมหาวิทยาลัยจริง ๆ ไม่ใช่บุคคลอื่นปลอมแปลงหรือแอบอ้างมา

อย่างไรก็ดี เนื่องจาก CA มีค่าบริการในการออกใบรับรองให้กับแต่ละบุคคลสูงถึง 8,900 - 17,900 บาทต่อปี [3] ซึ่งเมื่อลองมาคำนวณดูแล้ว ในปี พ.ศ. 2561 มีจำนวนผู้สำเร็จการศึกษาในระดับปริญญาตรี ปริญญาโท และปริญญาเอกมากถึง 5,113 คน [4] ถ้าคำนวณจากราคาขั้นต่ำ คือ 8,900 บาท มหาวิทยาลัยจะต้องเสียค่าใช้จ่ายในการทำ CA ให้กับผู้สำเร็จการศึกษามากถึง 45,505,700 บาท และ CA ยังมีการทำงานแบบรวมอยู่ที่

ศูนย์กลางที่เดียว (Centralized) ซึ่งถ้าเกิดปัญหาหรือถูกโจมตีโดยผู้ไม่หวังดี ข้อมูลที่เก็บอยู่กับ CA ก็อาจมีปัญหาหรือถูกแก้ไขได้

ผู้จัดทำจึงนำเทคโนโลยี Blockchain มาประยุกต์ใช้ ซึ่งเป็นรูปแบบการเก็บข้อมูล (Database) รูปแบบหนึ่งของระบบที่ไม่มีศูนย์กลาง เชื่อถือได้ เนื่องจากข้อมูลภายใน Chain ไม่สามารถแก้ไขได้ โดยผู้จัดทำจะนำเทคโนโลยี Private Blockchain ซึ่งจะสามารถเข้าถึงข้อมูลได้เฉพาะแค่ผู้ที่ได้รับอนุญาตเท่านั้น มาประยุกต์ใช้ในกระบวนการบริหารจัดการกุญแจสำหรับการเข้ารหัสลับ หรือที่เรียกว่า Key Management และจัดเก็บรายการของกุญแจที่ถูกเพิกถอนกรณีที่มีความจำเป็นต้องเปลี่ยนกุญแจ ส่วนที่จัดเก็บสถานะของกุญแจนี้จะเรียกว่า Key Revocation List ซึ่งสถานะของกุญแจจะมีทั้งหมด 3 สถานะ คือ Good (สถานะถูกต้อง), Revoked (สถานะถูกเพิกถอน) และ Unknown (ไม่สามารถตรวจสอบสถานะได้)

จากที่กล่าวมาข้างต้นจะเห็นได้ว่าเทคโนโลยี Blockchain นี้จะช่วยเพิ่มความน่าเชื่อถือของข้อมูล ช่วยลดค่าใช้จ่ายจากการใช้ CA และช่วยแก้ปัญหาที่อาจเกิดจากการทำงานแบบ Centralized ของ CA ได้ ด้วยเหตุผลทั้งหมดที่กล่าวมานี้จึงทำให้ Transcript ในรูปแบบดิจิทัลนั้นเป็นเอกสารที่มีความน่าเชื่อถือสูง และยังเป็นการเพิ่มความปลอดภัยให้ข้อมูลส่วนบุคคลภายใน Transcript ไปอีกระดับหนึ่งด้วย

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 พัฒนา Proof-of-concept ระบบ online digital transcript ที่มีความน่าเชื่อถือด้วยเทคโนโลยี Blockchain

1.2.2 เพื่อทดสอบประสิทธิภาพในการส่งข้อมูลเมื่อมีการส่งข้อมูลเข้ามาเป็นจำนวนมาก และมีจำนวน Node ในระบบมากขึ้น

1.2.3 เพื่อศึกษาและเปรียบเทียบข้อดีและข้อเสียของ Digital Transcript และ Transcript ในรูปแบบเดิม

1.2.4 ศึกษาการพัฒนาระบบผ่าน Cloud Computing

1.2.5 ศึกษาการใช้งานระบบผ่าน Mobile Application

1.2.6 ทดสอบประสิทธิภาพของระบบในรูปของจำนวน Transactions/Second

1.3 ขอบเขตของโครงงาน

1.3.1 ทำระบบ Generate key pair สำหรับออก Key ให้กับผู้ใช้

1.3.2 พัฒนาระบบบน Cloud

1.3.3 พัฒนา Application เพื่อให้ผู้ใช้สามารถเข้าถึง Private Blockchain ได้

1.3.4 สามารถติดตั้ง Hyperledger Fabric และปรับแต่งให้เป็นรูปแบบที่ต้องการได้

1.3.5 สามารถเขียนโครงสร้างของ Verifiable Credential ในรูปแบบของ Transcript ได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ช่วยลดระยะเวลาในการขอ Transcript จากมหาวิทยาลัย

1.4.2 สร้างความน่าเชื่อถือให้กับ Transcript ของนักศึกษา

1.4.3 ช่วยลดค่าใช้จ่ายในการใช้เทคโนโลยี Certificate Authority

1.4.4 ช่วยลดการใช้งานทรัพยากรกระดาษ

1.4.5 ช่วยเพิ่มความน่าเชื่อถือให้กับระบบการออก Transcript ของมหาวิทยาลัย

1.4.6 ช่วยเพิ่มความน่าเชื่อถือและความปลอดภัยของการทำ Key Management

1.4.7 เพื่อลดอัตราความเสี่ยงที่จะเกิดการปลอมแปลง Transcript

1.4.8 เพื่อนำระบบการออก Transcript ของมหาวิทยาลัยเข้าสู่ระบบดิจิทัล

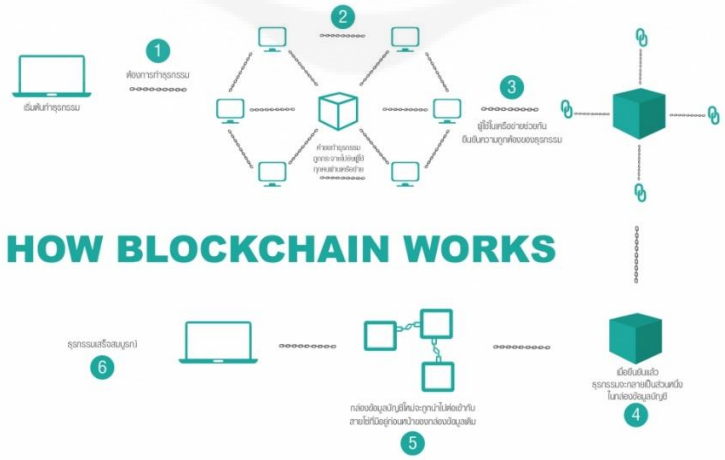
1.4.9 ข้อมูลถูกเปิดเผยให้กับบุคคลที่ได้รับอนุญาตเท่านั้น

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีเกี่ยวกับ Blockchain

Blockchain เป็นเทคโนโลยีในการจัดเก็บข้อมูลแบบไม่ต้องอาศัยคนกลาง (Third Party) ในการเก็บข้อมูล [5] โดยทั่วไปการเก็บข้อมูลดิจิทัลนั้นจะต้องมีแม่ข่ายหรือศูนย์ข้อมูลกลาง (Data Center) ในการจัดเก็บข้อมูล หากสังเกตดูจะพบว่าหลาย ๆ กิจการรอบตัวเราล้วนต้องอาศัยคนกลางทั้งสิ้น ไม่ว่าจะเป็นธุรกรรมทางการเงินที่ต้องอาศัยธนาคารเป็นตัวกลางให้ธุรกรรมทางการเงินนั้นเสร็จสมบูรณ์ เมื่อเป็นเช่นนี้แล้ว หากคนกลางหายไป รูปแบบกิจกรรมก็ย่อมเปลี่ยนไปด้วยเช่นกัน ซึ่ง Blockchain จะเข้ามาเปลี่ยนรูปแบบการทำงานจากแบบเดิมไปเป็นแบบใหม่โดยสิ้นเชิง โดยที่สมาชิกทุกคนมีสิทธิเท่าเทียมกันในการรับและส่งข้อมูล ระบบนี้มีกลไกในการเข้ารหัสของข้อมูลซึ่งการเข้ารหัสของข้อมูลนี้ก็เพื่อรักษาความปลอดภัยของข้อมูล หากข้อมูลมีการเปลี่ยนแปลง สมาชิกทุกคนในเครือข่ายต้องทำการตรวจสอบข้อมูลนั้นเพื่อยืนยันความถูกต้องของข้อมูล ดังนั้นเมื่อเกิดการเปลี่ยนแปลง สมาชิกทุกคนจะได้รับทราบ ยืนยัน และจัดเก็บข้อมูลต่อไป โดยในรูปภาพที่ 1 เป็นตัวอย่างการทำงานของ Blockchain โดยมีขั้นตอนดังนี้

1. ส่งคำขอเพื่อทำธุรกรรม
2. คำขอทำธุรกรรมถูกกระจายไปยังผู้ใช้ทุกคนผ่านเครือข่าย
3. ผู้ใช้ในเครือข่ายยืนยันความถูกต้องของธุรกรรม
4. เมื่อยืนยันแล้ว ธุรกรรมจะกลายเป็นส่วนหนึ่งในกล่องข้อมูลบัญชี
5. กล่องข้อมูลบัญชีใหม่จะถูกนำไปต่อเข้ากับสายโซ่ที่มีอยู่ก่อนหน้าของกล่องข้อมูลเดิม
6. ธุรกรรมเสร็จสมบูรณ์



รูปภาพที่ 1 การทำงานของ Blockchain

2.1.1 องค์ประกอบของ Blockchain

1. Block

เป็นการเก็บ Data transaction โดยบรรจุในกล่อง (Block) และเมื่อปิดกล่องแล้ว จะ Hash ข้อมูล Data Transaction ภายในกล่องกำกับไว้ที่ Header Block ทำให้ข้อมูลไม่สามารถเปลี่ยนแปลงได้ และจะกระจายข้อมูลไปให้ทุกคนที่เกี่ยวข้องเก็บไว้

2. Chain

เป็นการนำกล่องใหม่ที่สร้างขึ้นมาต่อกับกล่องก่อนหน้า โดยจะมีการเก็บค่า Hash ของกล่องก่อนหน้าไว้ที่กล่องใหม่ ซึ่งทำให้เปลี่ยน Data transaction ที่เกิดขึ้นมาแล้วไม่ได้ เนื่องจากมี Hash กำกับไว้ที่กล่อง และถ้าต้องการจะแก้ไขข้อมูล ต้องตามแก้ทุกกล่องภายใน chain ดังนั้น Hash เปรียบเสมือนค่าที่นำมาใช้ในการยืนยันความถูกต้องของข้อมูลในแต่ละกล่อง

3. Consensus

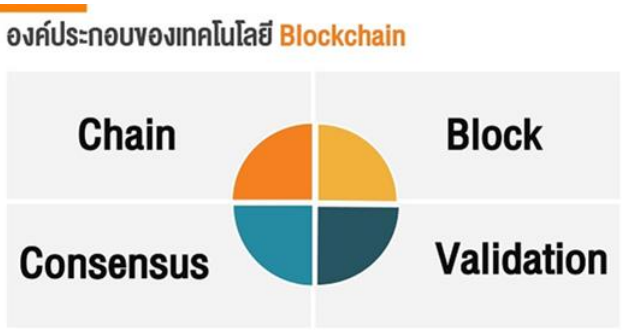
เป็นวิธีการตกลงของผู้ที่อยู่ในเครือข่าย Blockchain เดียวกัน โดยทำข้อตกลงในการใช้งานร่วมกัน เพื่อที่จะใช้ในการตรวจสอบความถูกต้อง

4. Validation

เป็นการตรวจสอบความถูกต้อง [6] โดยจะตรวจสอบทั้งระบบและทุก Node ในระบบ Blockchain เพื่อให้แน่ใจว่าจะไม่มีข้อผิดพลาดเกิดขึ้นไม่ว่าจะมาจากส่วนใดก็ตาม ซึ่งก็คือส่วนหนึ่งของ Consensus ที่เรียกว่า Proof of work ซึ่งโดยหลักการแล้วการทำ Validation นั้นมีจุดประสงค์อยู่ 2 ประการ คือ

1. วิธีการในการยอมรับ/ปฏิเสธ รายการใน Block นั้น ๆ
2. วิธีการตรวจสอบที่ทุกคนในระบบยอมรับร่วมกัน

โดยองค์ประกอบทั้งสี่นี้จะเป็นองค์ประกอบหลักของเทคโนโลยี Blockchain ตามที่เห็นในรูปภาพที่ 2



รูปภาพที่ 2 องค์ประกอบของ Blockchain

2.1.2 ประเภทของ Blockchain

1. Public Blockchain คือ Blockchain แบบสาธารณะที่ทุกคนสามารถเข้าใช้งานได้อย่างอิสระ เช่น Bitcoin, Ethereum

ข้อดี : การส่งข้อมูลไปยังปลายทางไม่จำเป็นต้องสร้างช่องทางการติดต่อสื่อสารกัน และในปัจจุบันนิยมทำ Web Service API เพื่อให้ Application ติดต่อสื่อสารกัน ผู้ส่งเพียงแค่ใส่ข้อมูลลงไปใน Blockchain แล้วจ่าหน้าซองข้อมูลถึงผู้รับ ผู้รับก็จะได้รับข้อมูลโดยทันที

ข้อเสีย : ข้อมูลที่ใส่เข้าไปบน Public Blockchain จะกลายเป็นข้อมูลที่โดนเปิดเผยแก่สาธารณชน

2. Private Blockchain คือ Blockchain ที่สร้างขึ้นเพื่อใช้ภายในองค์กร หรือบริษัทภายในเครือเท่านั้นที่มีสิทธิ์เข้าถึง โดย Blockchain ประเภทนี้จะมีการจำกัดการเข้าถึงข้อมูล ทำให้จะมีเพียงคนบางกลุ่มเท่านั้นที่สามารถใช้งานระบบหรือเข้าถึงข้อมูลได้

ข้อดี : ช่วยลดปัญหาด้านความเป็นส่วนตัวของข้อมูล และสามารถปรับกฎเกณฑ์ต่าง ๆ ของ Blockchain Network ที่ต้องการได้

ข้อเสีย : องค์กรต้องลงทุนในการสร้างระบบ Infrastructure ขึ้นมาให้รองรับการทำงานภายในองค์กร ซึ่งต้องพ่วงมากับการดูแลรักษาระบบ และเม็ดเงินที่องค์กรจะต้องลงทุนซึ่งมีมูลค่าสูง

3. Consortium Blockchain คือ Blockchain ที่รวมแนวคิดของ Public และ Private Blockchain เข้าด้วยกัน โดยนิยมใช้ในองค์กรต่าง ๆ ที่มีลักษณะธุรกิจเหมือนกันและต้องแลกเปลี่ยนข้อมูลกันอยู่แล้ว เช่น ธนาคารใช้ในการแลกเปลี่ยนข้อมูลการโอนเงินกันภายในสมาคมธนาคารด้วยกัน และธนาคารที่จะเข้ามาร่วมในวงได้ต้องได้รับการอนุญาตจากตัวแทนเสียก่อนถึงจะมีสิทธิ์เข้าถึงการใช้งานร่วมกันได้

ข้อดี : ธนาคารไม่ต้องกลัวว่าข้อมูลสำคัญขององค์กรและลูกค้าจะกลายเป็นข้อมูล Public และในส่วนของการลงทุนด้าน Infrastructure ก็จะลดลงไม่เหมือนการสร้าง Private Blockchain ขึ้นมาใช้เองภายในองค์กรซึ่งต้องใช้งบประมาณสูง

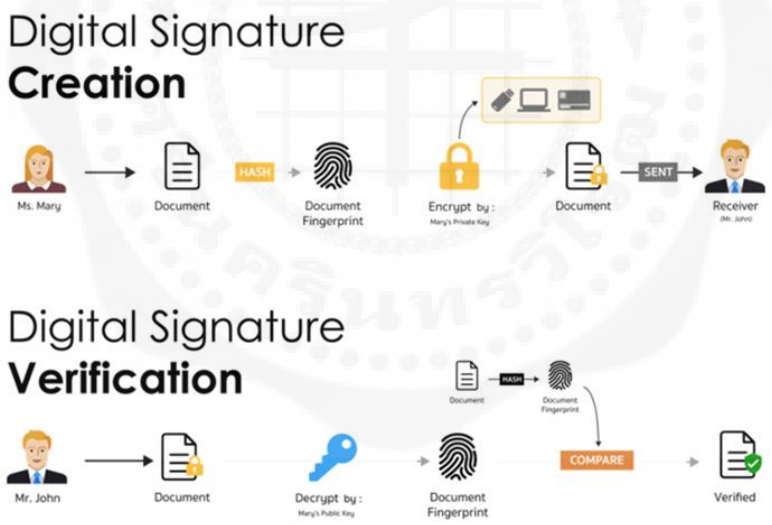
ข้อเสีย : มีความไม่คล่องตัวในการปรับปรุงเปลี่ยนแปลงเงื่อนไขการใช้งานต่าง ๆ เพราะอาจจะต้องผ่านมติเห็นชอบภายในสมาคมก่อน

2.2 ทฤษฎีเกี่ยวกับ Digital Signature

Digital Signature เป็นลายมือชื่อที่อยู่ในรูปแบบของอิเล็กทรอนิกส์ที่มีคุณสมบัติด้านความปลอดภัย เพื่อให้มีความน่าเชื่อถือมากยิ่งขึ้น [7] ประกอบด้วย

1. Signer Authentication คือ ความสามารถในการพิสูจน์ว่าใครเป็นคนลงลายมือชื่อบนเอกสาร ตัวลายมือชื่อจะสามารถใช้ในการเชื่อมโยงไปยังบุคคลที่ลงลายมือชื่อบนเอกสารได้
2. Data Integrity คือ ความสามารถในการตรวจสอบ หรือพิสูจน์ว่ามีการแก้ไขเอกสารหลังจากที่ได้มีการลงลายมือชื่อไปแล้วหรือไม่
3. Non-repudiation คือ การไม่สามารถปฏิเสธความรับผิดชอบได้ เนื่องจากลายมือชื่อที่สร้างขึ้นมีเอกลักษณ์ สามารถพิสูจน์ในชั้นศาลได้ว่าใครเป็นผู้ลงลายมือชื่อบนเอกสาร

จากรูปภาพที่ 3 แสดงให้เห็นกระบวนการสร้างและยืนยันลายมือชื่อดิจิทัล โดยมีกระบวนการดังนี้ ผู้ส่งข้อมูลจะทำการ Hash ข้อมูลหลังจากนั้นทำการเข้ารหัสข้อมูลแล้วส่งให้กับผู้รับ เมื่อผู้รับได้รับข้อมูลแล้วก็จะทำการถอดรหัสข้อมูล ถ้าค่า Hash ของข้อมูลและกุญแจตรงกันถือว่าข้อมูลถูกต้อง

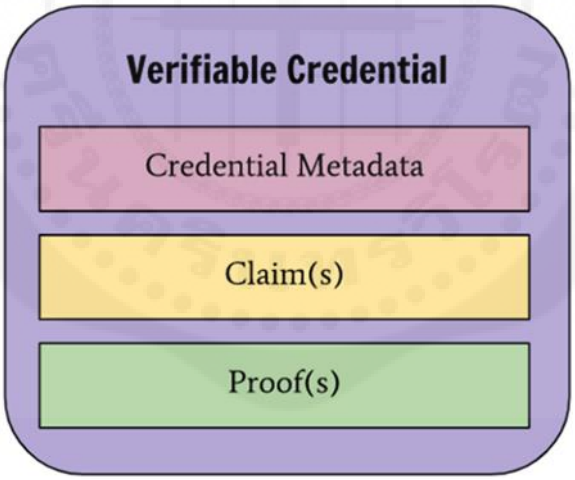


รูปภาพที่ 3 กระบวนการสร้างและยืนยันลายมือดิจิทัล

2.3 ทฤษฎีเกี่ยวกับ Verifiable Credential

Verifiable Credential (VC) หรือสารรับรองที่ตรวจสอบได้ คือ ชุดของข้อกล่าวอ้างอย่างน้อยหนึ่งรายการที่ออกโดยผู้ออกสารเดียวกัน [8] โดยสารรับรองที่ตรวจสอบได้จะมีคุณสมบัติที่สามารถตรวจพบการปลอมแปลงและตรวจสอบผู้ที่เขียนข้อมูลได้ด้วยกระบวนการเข้ารหัส สารรับรองที่ตรวจสอบได้อาจประกอบด้วยตัวระบุ (identifier) และคำอธิบายข้อมูล (metadata) เช่น ผู้ออกสาร วันที่ออกสาร และวันที่สารสิ้นอายุ ซึ่งคำอธิบายข้อมูลอาจมีการลงลายมือชื่อโดยผู้ออกสารก็ได้

การสร้าง VC จะต้องมีการกำหนดรูปแบบของเอกสาร (Verifiable Credential Schema) ว่าจะให้มีข้อมูลอะไรบ้าง ซึ่งสามารถกำหนดรูปแบบได้ตามที่ต้องการ ทำให้ VC สามารถรองรับรูปแบบเอกสารได้แทบทุกชนิด ภายในจะประกอบไปด้วยคุณสมบัติต่าง ๆ (Property) เช่น ชื่อ นามสกุล ที่อยู่ หรือข้อมูลอื่น ๆ ข้อเรียกร้อง (Claim) คือ ลักษณะหรือข้อความเกี่ยวกับเจ้าของข้อมูล และข้อมูลพิสูจน์ (Proof) คือ คุณสมบัติที่ใช้แสดงวิธีการพิสูจน์เพื่อให้สารรับรองที่ตรวจสอบได้หรือสารสำแดงที่ตรวจสอบได้มีคุณสมบัติที่สามารถตรวจพบการปลอมแปลงและตรวจสอบผู้เขียนข้อมูลได้ด้วยกระบวนการเข้ารหัสลับ ซึ่งโครงสร้างที่กล่าวมาทั้งหมดจะถูกเก็บไว้ในรูปแบบของ VC ดังรูปภาพที่ 4



รูปภาพที่ 4 โครงสร้างของ Verifiable Credential

รูปภาพที่ 5 ข้อมูลตัวอย่างของ Verifiable Credential

จากรูปภาพที่ 5 แสดงตัวอย่าง VC ที่มีข้อมูลต่างๆ เช่น ID ของ Credential, ชนิดของ Credential, ผู้ออก Credential, วันที่ออก Credential, ID และชื่อมหาวิทยาลัย เป็นต้น

2.3.1 การสร้างเอกสาร (Credential Issuance)

การใช้ Transcript ในการสมัครงาน ผู้ถือเอกสาร (Holder) จำเป็นต้องขอเอกสารจากทางมหาวิทยาลัย ซึ่งในที่นี้คือผู้ออกเอกสาร (Issuer) ก่อนหน้าที่จะมีเอกสารประเภท Transcript ได้ จะต้องลงทะเบียน Identifier เพื่อระบุตัวตนบนระบบซึ่งอาจจะเป็น Public Key ที่ถูกสร้างด้วยวิธีการของระบบนั้น ๆ รวมถึงสร้างรูปแบบของเอกสาร (ทำโดย Issuer) ทั้งสองอย่างนี้จะต้องส่งขึ้นไปเก็บบน Verifiable Data Registry หรือตัวกลางในการเก็บข้อมูลที่ทุกคนภายในระบบควรรู้ เช่น Identifier และ Verifiable Credential Schema (รูปแบบเอกสาร) หลังจากที่มี Identifier และ Verifiable Credential Schema แล้ว Issuer ก็จะสามารถออกเอกสารให้กับ Holder ได้ทันที โดย Holder อาจจะเข้าไปดูชนิดของเอกสารที่จะขอจาก Issuer เอง จากนั้นก็จะยื่นเรื่องขอเอกสาร Transcript ที่ลงลายมือชื่อดิจิทัลของมหาวิทยาลัย โดย VC จะเก็บไว้ที่ Holder เพียงผู้เดียว

2.4 ทฤษฎีเกี่ยวกับ Key Management

Key management คือ กระบวนการบริหารจัดการกุญแจสำหรับการเข้ารหัสลับ [9] ซึ่งจะเกี่ยวข้องกับการสร้าง ปกป้อง จัดเก็บ เปลี่ยนแปลง และแก้ไขกุญแจ โดยจุดประสงค์หลักของการบริหารจัดการกุญแจมี 3 อย่าง คือ

1. Secure key stores

การจัดเก็บกุญแจจำเป็นต้องมีการป้องกันและเก็บรักษาเป็นอย่างดี [10] ทั้งขณะที่จัดเก็บ ขณะส่ง และขณะสำรองข้อมูลเก็บเอาไว้ แต่ถ้าการจัดเก็บกุญแจนั้นจัดเก็บไว้อย่างไม่เหมาะสม จะทำให้ข้อมูลที่ถูกเข้ารหัสไว้นั้นไม่ปลอดภัย

2. Access to key stores

การเข้าถึงพื้นที่จัดเก็บกุญแจจำเป็นต้องใช้กุญแจส่วนตัวของแต่ละบุคคล และจะต้องมีนโยบายที่ใช้ในการควบคุมการเข้าถึงพื้นที่เก็บกุญแจ

3. Key backup and recoverability

การสูญหายของกุญแจนั้นสามารถเกิดขึ้นได้อย่างแน่นอน และนั่นหมายความว่าจะสูญเสียข้อมูลที่ถูกกุญแจนั้นเข้ารหัสไว้ด้วย ซึ่งถ้าข้อมูลที่สูญเสียไปนั้นเป็นข้อมูลสำคัญ อาจส่งผลให้เกิดความเสียหายต่อบุคคล ธุรกิจ หรือองค์กรได้ ดังนั้นจำเป็นต้องมีระบบที่สามารถสำรองและกู้คืนข้อมูลได้อย่างปลอดภัย

2.5 ทฤษฎีเกี่ยวกับ Certificate Authority

Certificate Authority (CA) คือ ผู้ให้บริการออกใบรับรองอิเล็กทรอนิกส์ ว่าเว็บไซต์มีความปลอดภัยและน่าเชื่อถือ [11] ผู้ใช้งานอินเทอร์เน็ตสามารถเข้าเยี่ยมชมเว็บไซต์ได้โดยไม่ต้องกังวลเรื่องภัยคุกคามมากนัก เนื่องจากมีการกรองข้อมูลต่าง ๆ บนเว็บไซต์มาในระดับหนึ่งแล้ว

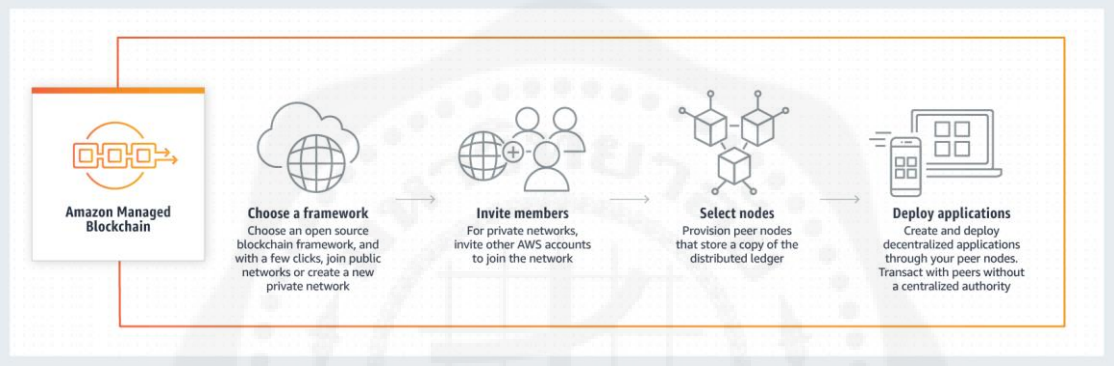
2.6 ทฤษฎีเกี่ยวกับ Cloud Computing

Cloud Computing คือ ระบบคอมพิวเตอร์ที่เกิดขึ้นเพื่อรองรับการทำงานของผู้ใช้งานในทุกๆด้าน ทั้งด้านระบบเครือข่าย ด้านการจัดเก็บข้อมูล ด้านการติดตั้งฐานข้อมูล หรือการใช้งานซอฟต์แวร์เฉพาะด้านในธุรกิจต่างๆ เป็นต้น โดยที่ผู้ใช้บริการไม่จำเป็นต้องติดตั้งระบบทั้งฮาร์ดแวร์และซอฟต์แวร์ไว้ที่สำนักงานของตนให้มันยุ่งยาก แต่ผู้ใช้บริการสามารถใช้งานง่ายๆ ด้วยการเชื่อมต่อกับระบบ Cloud Computing ผ่านระบบอินเทอร์เน็ต โดย Cloud Computing แบ่งออกเป็น 5 ประเภท ได้แก่

1. Infrastructure as a Service (IaaS) เป็นโครงสร้างพื้นฐานเหมือนกับระบบคอมพิวเตอร์ สามารถเข้าถึงได้ผ่านเครือข่าย ช่วยรองรับความต้องการใช้งานในด้านการประมวลผล และด้านพื้นที่การจัดเก็บข้อมูล
2. Platform as a Service (PaaS) เป็นบริการด้านการใช้งานเกี่ยวกับแพลตฟอร์มซอฟต์แวร์ต่างๆ เช่น แอปพลิเคชัน เว็บไซต์ ระบบประมวลผลกลางสำหรับองค์กรขนาดใหญ่ และอื่นๆ โดยจะเป็นการทำงานที่มีการควบคุมในเรื่องการรักษาความปลอดภัยสูง
3. Software as a Service (SaaS) เป็นบริการด้านแอพพลิเคชัน โดยจะคิดค่าบริการตามจำนวนของผู้ใช้งาน หรือตามปริมาณการใช้งานของผู้ใช้งาน อย่าง Google Apps หรือ Google Mail เป็นต้น
4. Data as a Service (DaaS) เป็นแหล่งเก็บข้อมูลดิบหรือข้อมูลเพื่อใช้เชื่อมโยงนำมาวิเคราะห์ มีการให้บริการข้อมูลข่าวสาร
5. Business Process as a Service (BPaaS) เป็นการให้บริการเพื่อเอื้อประโยชน์สำหรับผู้ที่ทำธุรกิจ ที่มีความต้องการปรับปรุงกระบวนการทางธุรกิจ และวัดผลลัพธ์ทางธุรกิจของตน

2.7 AWS Managed Blockchain

AWS Manage Blockchain ช่วยให้การสร้างและปรับใช้เครือข่าย Blockchain นั้นทำง่าย รวดเร็ว และปลอดภัย โดยใช้กรอบงานโอเพนซอร์สที่ได้รับความนิยม เทมเพลตเหล่านี้ช่วยให้สามารถมุ่งเน้นไปที่การสร้างแอปพลิเคชัน Blockchain แทนที่จะใช้เวลาและพลังงานไปกับการตั้งค่าเครือข่าย Blockchain ด้วยตนเอง และยังไม่มีค่าใช้จ่ายเพิ่มเติมสำหรับการใช้งานเทมเพลต AWS Blockchain โดยจ่ายเฉพาะทรัพยากรที่จำเป็นในการใช้งานเครือข่าย Blockchain เท่านั้น ในส่วนของเทมเพลตที่มีให้เลือกใช้นั้นมี 2 เทมเพลต คือ เทมเพลตสำหรับ Ethereum และเทมเพลตสำหรับ Hyperledger Fabric



รูปภาพที่ 6 กระบวนการทำงานของ AWS Managed Blockchain

จากรูปภาพที่ 6 แสดงการทำงานของ AWS Managed Blockchain โดยมีขั้นตอนดังนี้

1. เลือก Blockchain framework แบบ Open source หลังจากนั้นเข้าร่วมกับ Public network หรือสร้าง Private network ขึ้นมาเอง
2. เชิญผู้อื่นเข้าร่วม Network ในกรณีที่เป็น Private network
3. จัดเตรียม Peer node สำหรับจัดเก็บสำเนา Distributed ledger
4. Deploy applications

2.8 ทฤษฎีเกี่ยวกับ Ethereum

Ethereum เป็นแพลตฟอร์ม Open source software ที่ใช้เทคโนโลยี Blockchain ซึ่งช่วยให้นักพัฒนาสามารถ Smart Contract และแอปพลิเคชัน Distributed ledger ได้ โดย Ethereum มีสกุลเงินดิจิตอลเป็นของตนเอง (ETH) ซึ่งเป็นอันดับ 2 รองจาก Bitcoin

รูปภาพที่ 7 ตราสัญลักษณ์ของ Ethereum

2.9 ทฤษฎีเกี่ยวกับ Hyperledger Fabric

Hyperledger Fabric ถูกออกแบบมาให้เป็น Distributed Ledger โดยใช้แนวคิดของ Channel ซึ่งเป็นช่องทางการส่งข้อมูล ทุกคนที่อยู่ใน Channel เดียวกันจะเห็นข้อมูลและทำงานกับข้อมูลได้เหมือน ๆ กัน แต่คนที่อยู่นอก Channel ถึงแม้จะเข้าถึง Blockchain ได้แต่ก็ไม่สามารถถอดรหัสข้อมูลได้ นอกจากนี้ Hyperledger Fabric ยังสามารถเปลี่ยนแปลง Consensus และระบบ Login ของแต่ละองค์กรได้อย่างง่ายดาย

รูปภาพที่ 8 ตราสัญลักษณ์ของ Hyperledger Fabric

2.10 งานวิจัยที่เกี่ยวข้อง

2.10.1 งานวิจัย เรื่อง “Performance Analysis of Private Blockchain Platforms in Varying Workloads”

บทความนี้นำเสนอการวัดประสิทธิภาพ Private blockchain ของผู้พัฒนาทั้งสองเจ้าอย่าง Ethereum และ Hyperledger Fabric [12] โดยจะทำการส่ง transaction เข้าไปเป็นจำนวนมากเพื่อเปรียบเทียบ จากผลการประเมินแสดงให้เห็นว่า Hyperledger Fabric สามารถรับปริมาณของ Transaction ได้เยอะอีกทั้งยังมีความเร็วในการส่งที่สูงกว่าทางฝั่งของ Ethereum โดยเฉพาะตอนที่มีการส่งข้อมูลเข้าไปถึง 10,000 Transaction นอกจากนี้ ความเร็วเฉลี่ยของการส่งข้อมูลเมื่อเทียบจำนวนของ Transaction แล้วจะเห็นว่า Hyperledger Fabric จะยิ่งเหนือกว่าเมื่อมีการส่งข้อมูลเข้าไปเป็นจำนวนมาก

2.10.2 งานวิจัย เรื่อง “KeyChain: Bloackchain-based Key Distribution”

วิธีป้องกันความเป็นส่วนตัวและความปลอดภัยในการส่งข้อความแบบ End-to-End Encryption (E2EE) เป็นเรื่องที่สำคัญมาก [13] ส่วนที่สำคัญที่สุดส่วนหนึ่งคือปัญหาการกระจาย Public key ให้ปลอดภัย เราเลยนำเสนอ KeyChain ที่เราใช้ Blockchain ในการเก็บ Public-key-to-ID ที่สามารถตรวจสอบย้อนหลังได้และมีความโปร่งใส โดยจะใช้ Delegated Proof of Stake (DPoS) เป็นกลไกแบบเอกฉันท์ (Consensus Mechanism) แทนการใช้ Proof of Work (PoW) เพื่อที่จะลดการคำนวณลง ดังนั้น ผู้ที่จะมีสิทธิ์เขียนข้อมูลลง Blockchain ก็คือคนที่จะถูกโหวตโดยผู้ใช้ที่มีเหรียญ โดยเราจะมีวิธีทำให้ปลอดภัยโดยการใช้ Multi-signed Certificate เพื่อที่จะตรวจสอบสถานะว่าคนนั้นมีสิทธิ์เขียนข้อมูลจริง ๆ แต่ถ้าหากผู้โจมตีเกิดมีเสียงโหวตตั้งแต่ 51% ขึ้นไป ผู้โจมตีก็จะสามารถเปลี่ยนแปลงแก้ไข Public-key-to-ID ได้แต่ก็ไม่สามารถทำอะไรได้นาน เพราะในรอบการโหวตต่อ ๆ ไป ผู้ใช้ก็จะไม่ยอมโหวตให้กับผู้ที่เปลี่ยนแปลงข้อมูลใน Blockchain เป็นผู้เขียนอีก

2.10.3 งานวิจัย เรื่อง “Full-text Search for Verifiable Credential Metadata on Distributed Ledgers” [14]

ในบทความนี้จะเสนอเกี่ยวกับการค้นหา Verifiable Credential Metadata บน Distributed Ledgers โดยจะใช้เป็น Self-Sovereign Identity model เพื่อที่จะเก็บ Metadata ในสถานการณ์นี้ ผู้เขียนจะสร้างฟีเจอร์ที่เรียกว่า Full-text search เพื่อทำการค้นหาข้อมูลบน Blockchain ซึ่งข้อมูลนั้นจะประกอบไปด้วย Metadata ของ Verifiable Credential ทางผู้เขียนบทความได้ลองสร้างและวัดประสิทธิภาพของการเรียกดูข้อมูลแล้วจากรูปที่ 3 พบว่าประสิทธิภาพการทำงานค่อนข้างได้ผลลัพธ์ที่น่าพอใจ และคิดว่าเป็นระบบที่ทั่วโลกควรต้องมีในอนาคต

2.10.4 บทความ เรื่อง “ตรวจปริญญาปลอมโดยใช้ Blockchain” [15]

บริษัท ดิจิทัล เวนเจอร์ส ได้นำ Blockchain เข้ามาตรวจสอบเอกสารทางการศึกษาด้วยเทคโนโลยี B.VER (Blockchain Solution for Academic Document Verification) โดยขั้นตอนการทำงานจะเป็นการอัปโหลดไฟล์ต้นฉบับที่มีการเข้ารหัสขึ้นสู่แพลตฟอร์มโดยมหาวิทยาลัย และผู้ตรวจสอบสามารถตรวจสอบโดยอัปโหลดเอกสารขึ้นไปเพื่อเทียบกับเอกสารต้นฉบับ ซึ่งจะเปิดให้บริการตรวจสอบได้ตั้งแต่มกราคม 2562 เป็นต้นไป แต่ถึงอย่างไรก็ตาม ระบบนี้ยังไม่มีการยืนยันการใช้งานได้จริงจนปัจจุบัน

บทที่ 3
วิธีการดำเนินโครงงาน

3.1 วิธีการดำเนินงาน

3.1.1 เริ่มต้นและวางแผนโครงงาน

3.1.1.1 กำหนดฟังก์ชันการทำงานของระบบที่ต้องการเพิ่มเติม

3.1.1.2 วางแผนการแบ่งภาระงานและตรวจสอบขั้นตอนการทำงาน

3.1.1.3 กำหนดระยะเวลาการทำงานที่ชัดเจน

3.1.2 ดำเนินการทำงานวิจัย

3.1.2.1 เพิ่มฟังก์ชันการทำงานของระบบ

3.1.2.2 นำระบบไปพัฒนาบน Server

3.1.2.3 พัฒนา Application

3.1.3 ทดสอบการทำงานของระบบ

3.1.3.1 ทดสอบการทำงานของระบบ

3.1.3.2 หาข้อผิดพลาดและแก้ไขปรับปรุง

3.1.4 สรุปผลและเผยแพร่งานวิจัย

3.1.4.1 สรุปผลงานวิจัย

3.2 ตารางแผนการดำเนินงาน

ตารางที่ 1 แผนการดำเนินงาน

ขั้นตอนการดำเนินการ	ปีการศึกษา 2563			
	ม.ค. 64	ก.พ. 64	มี.ค. 64	เม.ย. 64
เริ่มต้นและวางแผนโครงงาน				
1. กำหนดฟังก์ชันการทำงานของระบบที่ต้องการเพิ่มเติม				
2. วางแผนการแบ่งภาระงานและตรวจสอบขั้นตอนการทำงาน				
3. กำหนดระยะเวลาการทำงานที่ชัดเจน				
ดำเนินการทำงานวิจัย				
4. เพิ่มฟังก์ชันการทำงานของระบบ				
5. นำระบบไปพัฒนาบน Server				
6. พัฒนา Application				
ทดสอบการทำงานของระบบ				
7. ทดสอบการทำงานของระบบ				
8. หาข้อผิดพลาดและแก้ไขปรับปรุง				
สรุปผลและเผยแพร่งานวิจัย				
9. สรุปผลงานวิจัย				

3.3 อุปกรณ์และเครื่องมือที่ใช้

3.3.1 ฮาร์ดแวร์

3.3.1.1 Blockchain services environment

- Instance type: bc.m5.xlarge
- CPU: 4 Cores
- Ram 16 GB
- CouchDB

3.3.1.2 EC2 environment

- Operating System: Amazon Linux 2 AMI 64 bit (x86)
- Instance type: c5.large
- CPU: 2 Core
- Ram 4 GB
- Storage type: Elastic Block Store
- Volume type: General purpose SSD (gp2) 8 GB
- Network Performance: Up to 10 Gbps

3.3.2 ซอฟต์แวร์

3.3.2.1 Virtual Box

3.3.2.2 Linux Ubuntu

3.3.2.3 Visual Studio Code

3.3.2.4 Hyperledger Fabric

3.3.2.5 Flutter

3.3.2.6 Android Studio

3.3.2.7 Amazon Web Services (AWS)

3.3.2.8 Mobile application emulator

3.3.3 ภาษาที่ใช้

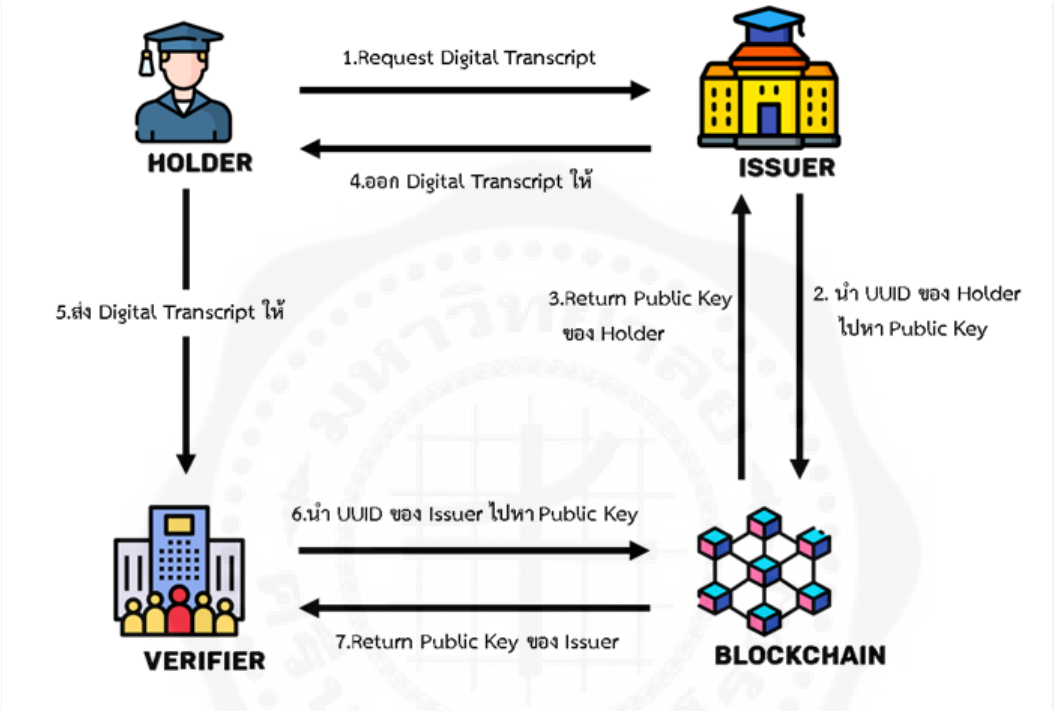
3.3.3.1 Node.js

3.3.3.2 Dart

3.4 การออกแบบและพัฒนาระบบ

1. กำหนดปัญหาที่ต้องการแก้ไขเพื่อนำไปวางแผนออกแบบระบบ
2. วางแผนและออกแบบการทำงานของระบบ

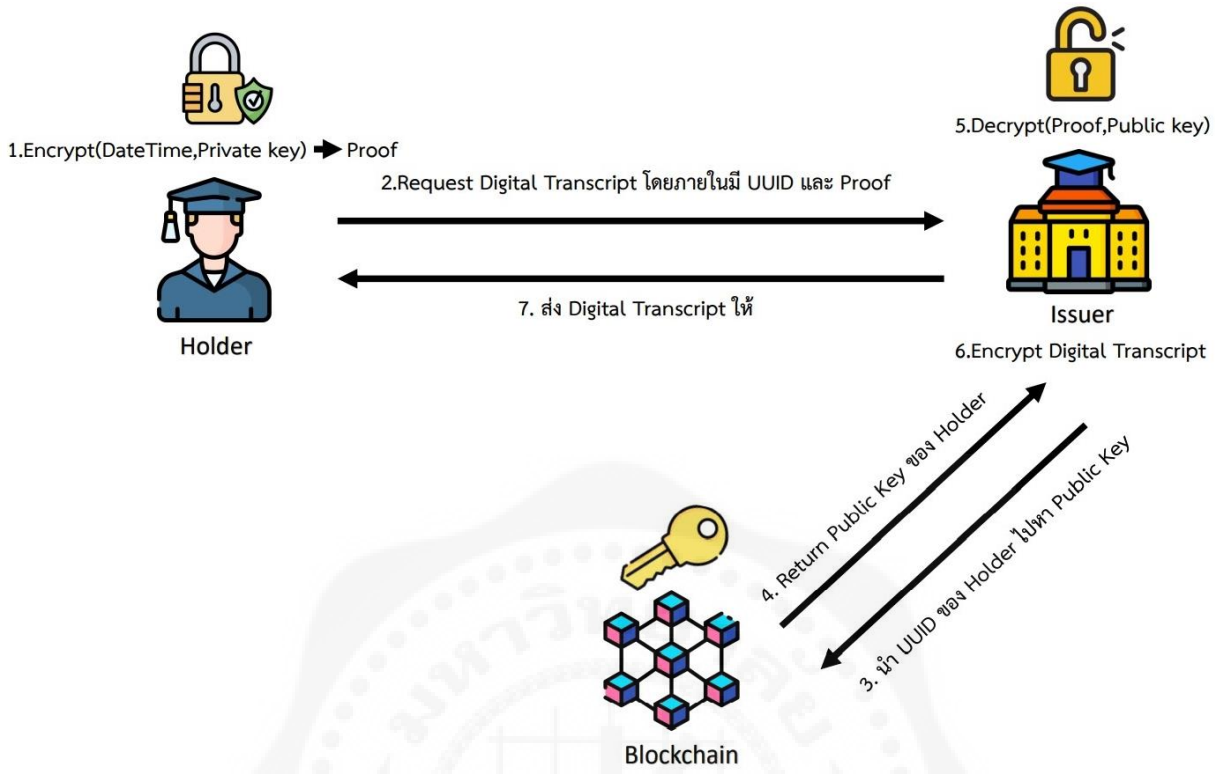
ขั้นตอนการทำงานของระบบ Digital Transcript



รูปภาพที่ 9 การทำงานของระบบ

จากรูปภาพที่ 9 แสดงการทำงานของระบบ โดยมีขั้นตอนการทำงาน ดังนี้

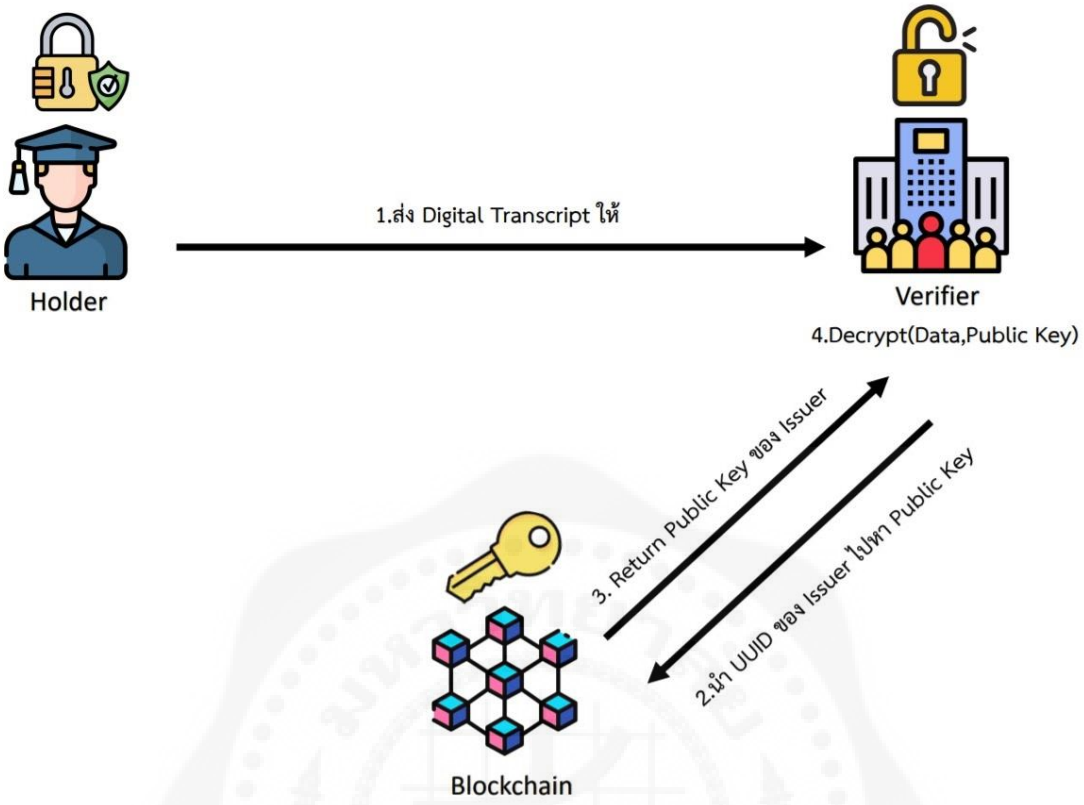
1. Holder ส่งคำร้องขอ Digital Transcript ไปยัง Issuer
2. Issuer นำ UUID ของ Holder ไปหา Public Key บน Blockchain
3. Blockchain ส่ง Public Key ของ Holder มาให้ Issuer เพื่อทำการตรวจสอบความถูกต้องของข้อมูล
4. Issuer ทำการออก Digital Transcript ให้กับ Holder
5. Holder ส่ง Digital Transcript ไปให้กับ Verifier
6. Verifier นำ UUID ของ Issuer ไปหา Public Key บน Blockchain
7. Blockchain ส่ง Public Key ของ Issuer มาให้ Verifier เพื่อทำการตรวจสอบความถูกต้องของข้อมูล



รูปภาพที่ 10 กระบวนการติดต่อกันระหว่าง Holder และ Issuer

จากรูปภาพที่ 10 แสดงกระบวนการติดต่อกันระหว่าง Holder และ Issuer โดยมีกระบวนการ ดังนี้

1. เข้ารหัสข้อมูล DateTime ด้วย Private Key ของ Holder เพื่อนำไปเป็น Proof ในการส่งคำร้องขอ Digital Transcript
2. ส่งคำร้องขอ Digital Transcript ไปยัง Issuer โดยภายในมี UUID และ Proof
3. Issuer นำ UUID ของ Holder ไปหา Public Key บน Blockchain
4. Blockchain ส่ง Public Key ของ Holder มาให้ Issuer เพื่อทำการตรวจสอบความถูกต้องของข้อมูล
5. Issuer ตรวจสอบความถูกต้องของข้อมูลการร้องขอ Digital Transcript จาก Holder ด้วย Public Key เพื่อทำการยืนยันตัวตนของ Holder
6. Issuer เข้ารหัส Digital Transcript เพื่อส่งให้ Holder
7. Issuer ส่ง Digital Transcript ให้ Holder

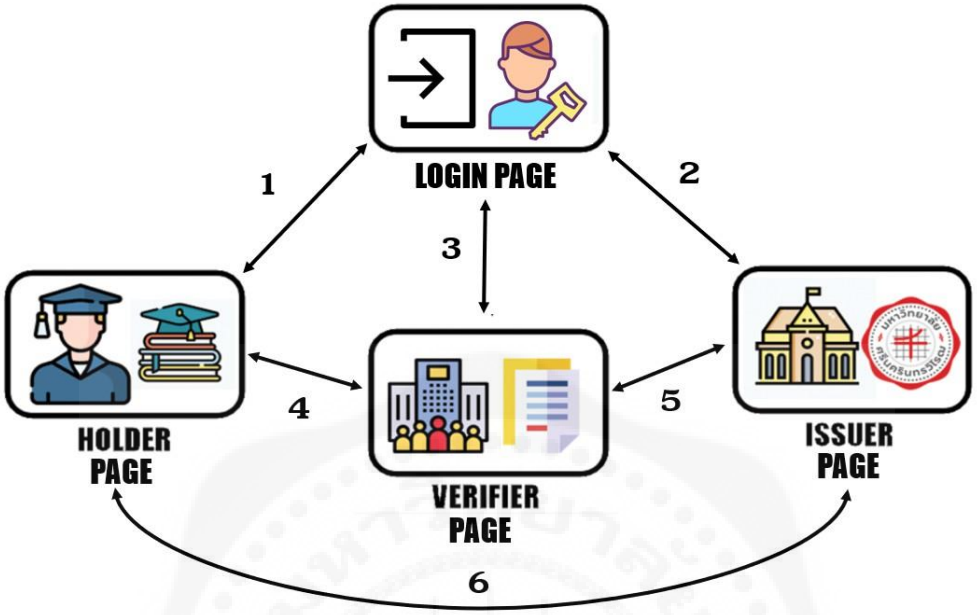


รูปภาพที่ 11 กระบวนการติดต่อกันระหว่าง Holder และ Verifier

จากรูปภาพที่ 11 แสดงกระบวนการติดต่อกันระหว่าง Holder และ Verifier โดยมีกระบวนการ ดังนี้

1. Holder ส่ง Digital Transcript ให้กับ Verifier
2. Verifier นำ UUID ของ Issuer ไปหา Public Key บน Blockchain
3. Blockchain ส่ง Public Key ของ Issuer มาให้ Verifier เพื่อทำการตรวจสอบความถูกต้องของข้อมูล
4. Verifier ตรวจสอบความถูกต้องของ Digital Transcript จาก Holder ด้วย Public Key ของ Issuer

ขั้นตอนการทำงานของ Application



รูปภาพที่ 12 ขั้นตอนการทำงานของ Application

จากรูปภาพที่ 12 แสดงขั้นตอนการทำงานของ Application โดยมีการติดต่อกัน ดังนี้

หมายเลข 1 แสดงการติดต่อกันระหว่างหน้า Login และหน้า Holder โดยจากหน้า Login สามารถไปที่หน้า Holder ได้ และจากหน้า Holder สามารถกลับไปที่หน้า Login ได้

หมายเลข 2 แสดงการติดต่อกันระหว่างหน้า Login และหน้า Issuer โดยจากหน้า Login สามารถไปที่หน้า Issuer ได้ และจากหน้า Issuer สามารถกลับไปที่หน้า Login ได้

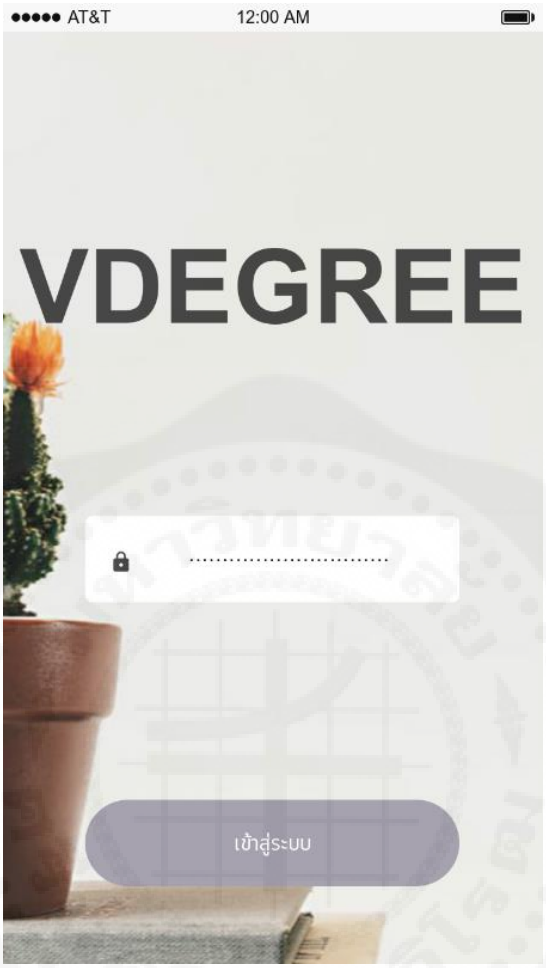
หมายเลข 3 แสดงการติดต่อกันระหว่างหน้า Login และหน้า Verifier โดยจากหน้า Login สามารถไปที่หน้า Verifier ได้ และจากหน้า Verifier สามารถกลับไปที่หน้า Login ได้

หมายเลข 4 แสดงการติดต่อกันระหว่างหน้า Holder และหน้า Verifier โดยจากหน้า Holder สามารถไปที่หน้า Verifier ได้ และจากหน้า Verifier สามารถไปที่หน้า Holder ได้

หมายเลข 5 แสดงการติดต่อกันระหว่างหน้า Verifier และหน้า Issuer โดยจากหน้า Verifier สามารถไปที่หน้า Issuer ได้ และจากหน้า Verifier สามารถไปที่หน้า Issuer ได้

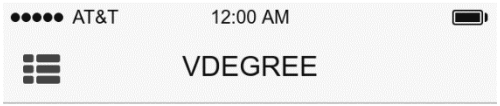
หมายเลข 6 แสดงการติดต่อกันระหว่างหน้า Holder และหน้า Issuer โดยจากหน้า Holder สามารถไปที่หน้า Issuer ได้ และจากหน้า Issuer สามารถไปที่หน้า Holder ได้

ออกแบบแอปพลิเคชันสำหรับให้ผู้ใช้งานสามารถติดต่อกับระบบได้



รูปภาพที่ 13 หน้าแรกของแอปพลิเคชัน

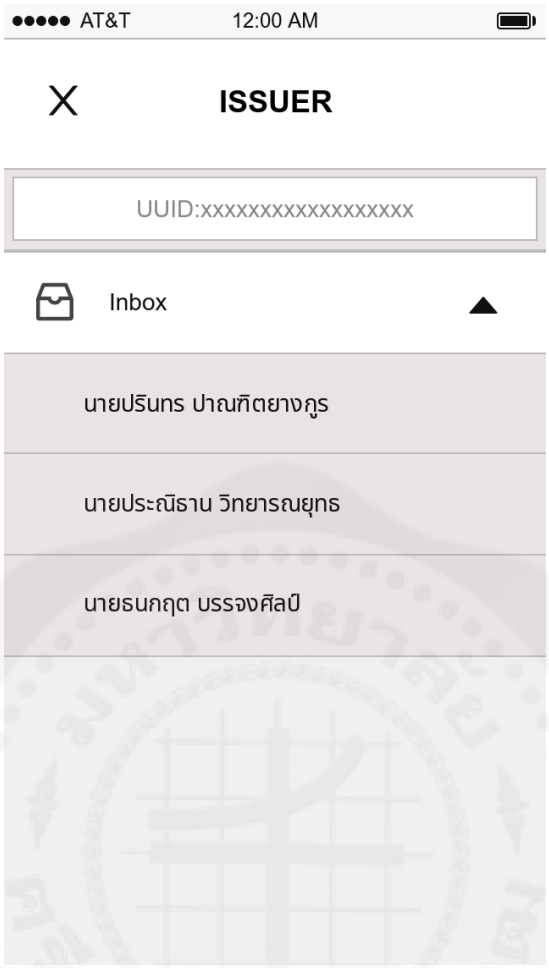
จากรูปภาพที่ 13 แสดงหน้าแรกของแอปพลิเคชันเพื่อเข้าใช้งานระบบ โดยจะต้องกรอกรหัสผ่านเพื่อเข้าใช้งานระบบ เมื่อกดปุ่มเข้าสู่ระบบแล้วจะเข้าสู่หน้า Holder, Issuer และ Verifier ขึ้นอยู่กับบทบาทของผู้ใช้



Empty

รูปภาพที่ 14 หน้าแอปพลิเคชันสำหรับ Issuer

จากรูปภาพที่ 14 แสดงหน้าแอปพลิเคชันสำหรับให้ผู้ใช้ที่เป็น Issuer เข้าใช้งานระบบ โดยด้านซ้ายมือเป็นปุ่มเมนู เมื่อกดแล้วจะแสดงแถบเมนูขึ้นมา ส่วนบริเวณพื้นที่ตรงกลางเป็นพื้นที่สำหรับแสดงข้อมูล โดยเป็นกรณีที่ยังไม่มีผู้ส่งคำขอ Transcript เข้ามา



รูปภาพที่ 15 แถบเมนูสำหรับ Issuer

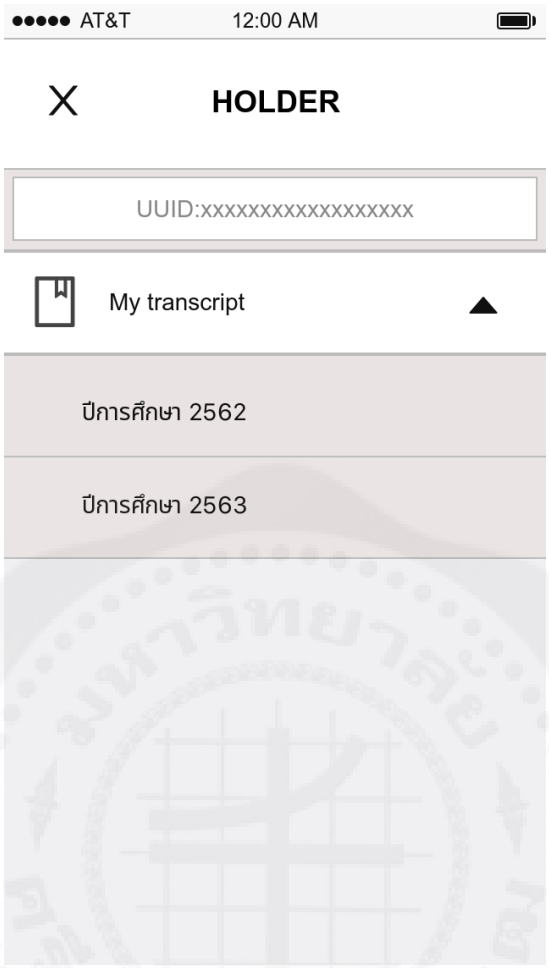
จากรูปภาพที่ 15 แสดงแถบเมนูสำหรับผู้ใช้ที่เป็น Issuer โดยมีเมนู Inbox เมื่อกดแล้วจะมีรายการของผู้ที่ส่งคำขอ Transcript เข้ามา และสามารถกดที่รายการเพื่อดูรายละเอียดของคำขอได้

รูปภาพที่ 16 หน้าแอปพลิเคชันสำหรับ Issuer เมื่อผู้ส่งคำขอเข้ามา

จากรูปภาพที่ 16 แสดงหน้าแอปพลิเคชันสำหรับผู้ใช้ที่เป็น Issuer โดยเป็นกรณีที่ Issuer กดที่รายการคำขอ จะแสดง UUID และรายชื่อของผู้ที่ส่งคำขอเข้ามา ด้านล่างมีช่องสำหรับกรอกรหัสผ่านและมีปุ่มให้เลือกว่าอนุมัติหรือไม่อนุมัติ

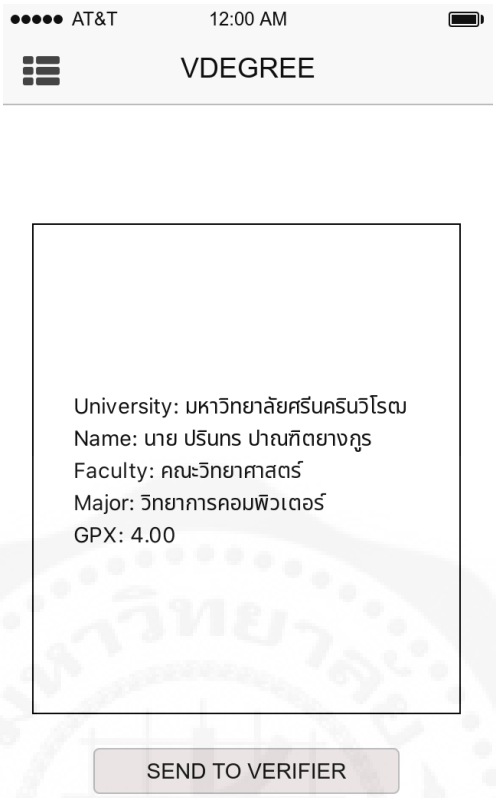
รูปภาพที่ 17 หน้าแอปพลิเคชันสำหรับ Holder

จากรูปภาพที่ 17 แสดงหน้าแอปพลิเคชันสำหรับให้ผู้ใช้ที่เป็น Holder เข้าใช้งานระบบ โดยด้านซ้ายมือเป็นปุ่มเมนู เมื่อกดแล้วจะแสดงแถบเมนูขึ้นมา ส่วนพื้นที่ตรงกลางเป็นพื้นที่สำหรับแสดงข้อมูล มีช่องสำหรับกรอกรหัสและมีปุ่มสำหรับส่งคำขอ Transcript



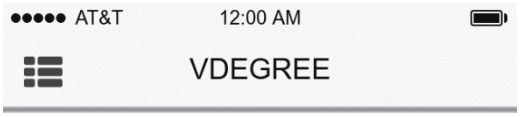
รูปภาพที่ 18 แถบเมนูสำหรับ Holder

จากรูปภาพที่ 18 แสดงแถบเมนูสำหรับผู้ใช้งานที่เป็น Holder โดยมีเมนู My transcript เมื่อกดแล้วจะมีรายการ Transcript ที่มีอยู่ขึ้นมา และสามารถกดที่รายการเพื่อดูรายละเอียดของ Transcript ได้



รูปภาพที่ 19 หน้าแอปพลิเคชันสำหรับแสดงรายละเอียด Transcript ของ Holder

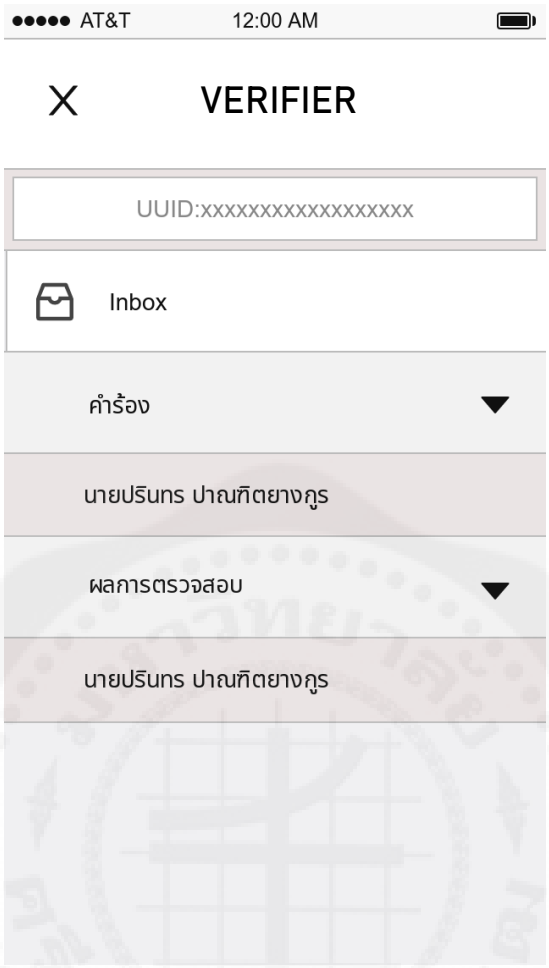
จากรูปภาพที่ 19 แสดงหน้าแอปพลิเคชันสำหรับผู้ใช้ที่เป็น Holder ในกรณีที่ Holder กดที่รายการ Transcript จะแสดงข้อมูลของ Transcript และมีปุ่มสำหรับส่ง Transcript ให้ Verifier



EMPTY

รูปภาพที่ 20 หน้าแอปพลิเคชันสำหรับ Verifier

จากรูปภาพที่ 20 แสดงหน้าแอปพลิเคชันสำหรับให้ผู้ใช้ที่เป็น Verifier เข้าใช้งานระบบ โดยด้านซ้ายมือเป็นปุ่มเมนู เมื่อกดแล้วจะแสดงแถบเมนูขึ้นมา ส่วนบริเวณพื้นที่ตรงกลางเป็นพื้นที่สำหรับแสดงข้อมูล โดยเป็นกรณีที่ยังไม่มีผู้ส่ง Transcript เข้ามา



รูปภาพที่ 21 แถบเมนูสำหรับ Verifier

จากรูปภาพที่ 21 แสดงแถบเมนูสำหรับผู้ใช้งานที่เป็น Verifier โดยมีเมนู Inbox เมื่อกดแล้วจะมีรายการ Transcript ที่ Holder ส่งเข้ามา และสามารถกดที่รายการเพื่อดูรายละเอียดของ Transcript ได้

รูปภาพที่ 22 หน้าแอปพลิเคชันของ Verifier กรณีมีผู้ส่ง Transcript เข้ามา

จากรูปภาพที่ 22 แสดงหน้าแอปพลิเคชันสำหรับผู้ใช้ที่เป็น Verifier ในกรณีกดที่รายการ Transcript ที่ส่งเข้ามา โดยจะแสดง UUID ของผู้ที่ส่งเข้ามา และมีปุ่มสำหรับตรวจสอบ Transcript

รูปภาพที่ 23 หน้าแอปพลิเคชันสำหรับแสดงรายละเอียด Transcript ของผู้ที่ส่งเข้ามา

จากรูปภาพที่ 23 แสดงหน้าแอปพลิเคชันสำหรับผู้ใช้ที่เป็น Verifier ในกรณีที่กดตรวจสอบ Transcript แล้ว จะแสดงข้อมูล Transcript ของผู้ที่ส่งเข้ามา

3.5 ปัญหาและอุปสรรค

1. หลังจากใช้ Blockchain service ของ AWS พบว่า Version ของ Hyperledger Fabric ต่ำกว่าระบบจำลองที่สร้างขึ้นก่อนหน้า จึงต้องทำการปรับเปลี่ยน Smart contract รวมถึงเรียนรู้คำสั่งของ Hyperledger Fabric ใหม่
2. AWS มีการจำกัดประสิทธิภาพของระบบ Network ทำให้มีอุปสรรคในการทดสอบประสิทธิภาพของระบบ
3. Flutter มีปัญหาในการเรียกใช้ API จึงต้องทำการตั้งค่าเพิ่มเติมในฝั่ง Server

บทที่ 4
ผลการดำเนินโครงงาน

4.1 การสร้าง Smart contract

```
const shim = require('fabric-shim');
const util = require('util');

var Chaincode = class {

    // Initialize the chaincode
    async Init(stub) {
      console.info('========= key_management02 Init =========');
      let ret = stub.getFunctionAndParameters();
      console.info(ret);
      let args = ret.params;
      if (args.length != 2) {
        return shim.error('Incorrect number of arguments. Expecting 2');
      }

      let uuid = args[0];
      let pu_key = args[1];

      try{
        await stub.putState(uuid, Buffer.from(pu_key));
        return shim.success();
      }catch (err){
        return shim.error(err);
      }
    }

```

รูปภาพที่ 24 smart contract ฉบับแก้ไข

จากรูปภาพที่ 24 Smart contract ตัวเก่าที่ใช้ ไม่สามารถใช้กับระบบของ Hyperledger Fabric บน AWS ได้ เพราะ มี Version ที่ต่ำกว่า จึงต้องแก้ไขเพื่อให้สามารถนำไปใช้กับระบบได้ ฟังก์ชัน Init คือฟังก์ชันที่จะใส่ข้อมูลแรกเข้าไปใน Blockchain

```
27      async Invoke(stub) {
28        let ret = stub.getFunctionAndParameters();
29        console.info(ret);
30        let method = this[ret.fcn];
31        if (!method) {
32          console.log('no method of name:' + ret.fcn + ' found');
33          return shim.success();
34        }
35        try {
36          let payload = await method(stub, ret.params);
37          return shim.success(payload);
38        } catch (err) {
39          console.log(err);
40          return shim.error(err);
41        }
42      }
43
```

รูปภาพที่ 25 ฟังก์ชัน Invoke

จากรูปภาพที่ 25 ฟังก์ชัน Invoke คือ ฟังก์ชันที่ใช้ทดสอบว่ามีฟังก์ชันดังกล่าวที่ใส่เข้าไปหรือไม่ ถ้าไม่มีจะแจ้งเตือนว่าฟังก์ชันชื่อนี้ไม่มีในระบบ แต่ถ้ามีก็จะให้ทำการใส่ข้อมูลไปที่ฟังก์ชันนั้น

```
44      async invoke(stub, args) {
45        if (args.length != 2) {
46          throw new Error('Incorrect number of arguments. Expecting 2');
47        }
48
49        let uuid = args[0];
50        let pu_key = args[1];
51
52
53        // Write the states back to the ledger
54        await stub.putState(uuid, Buffer.from(pu_key));
55
56      }
57
58      // Deletes an entity from state
59      async delete(stub, args) {
60        if (args.length != 1) {
61          throw new Error('Incorrect number of arguments. Expecting 1');
62        }
63
64        let uuid = args[0];
65
66        // Delete the key from the state in ledger
67        await stub.deleteState(uuid);
68      }
69
```

รูปภาพที่ 26 ฟังก์ชัน invoke

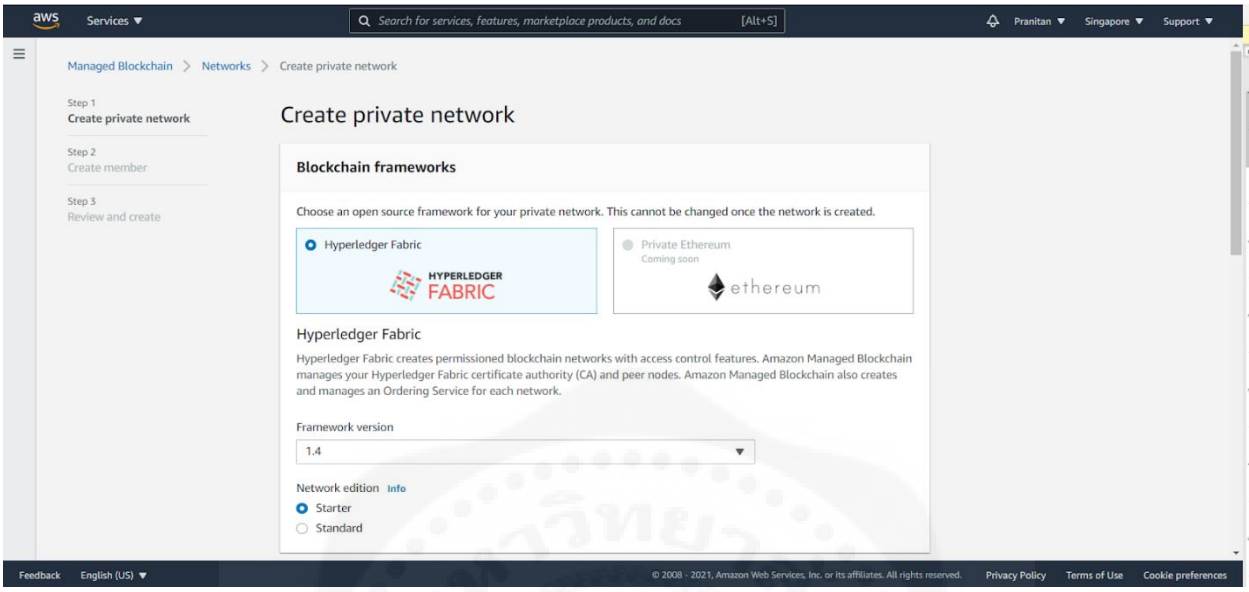
จากรูปภาพที่ 26 ฟังก์ชัน invoke ทำหน้าที่รับข้อมูลเข้ามาเป็น String 2 ค่า คือ UUID และ Public Key เพื่อที่จะเขียนลงบน Blockchain ฟังก์ชัน delete ทำการลบข้อมูลโดยทำการอิงจาก UUID

```
70      // query callback representing the query of a chaincode
71      async query(stub, args) {
72        if (args.length != 1) {
73          throw new Error('Incorrect number of arguments. Expecting name of the person to query')
74        }
75
76        let jsonResp = {};
77        let uuid = args[0];
78
79        // Get the state from the ledger
80        let Databytes = await stub.getState(uuid);
81        if (!Databytes) {
82          jsonResp.error = 'Failed to get state for ' + A;
83          throw new Error(JSON.stringify(jsonResp));
84        }
85
86        jsonResp.uuid = uuid;
87        jsonResp.publickey = Databytes.toString();
88        console.info('Query Response:');
89        console.info(jsonResp);
90        return Databytes;
91      }
92    };
93
94    shim.start(new Chaincode());
95
```

รูปภาพที่ 27 ฟังก์ชัน query

จากรูปภาพที่ 27 ฟังก์ชัน query เป็นฟังก์ชันที่จะนำ UUID ไปค้นหาข้อมูลบน Blockchain จากนั้นก็จะคืนค่าข้อมูลที่ได้ไปให้กับผู้เรียกใช้

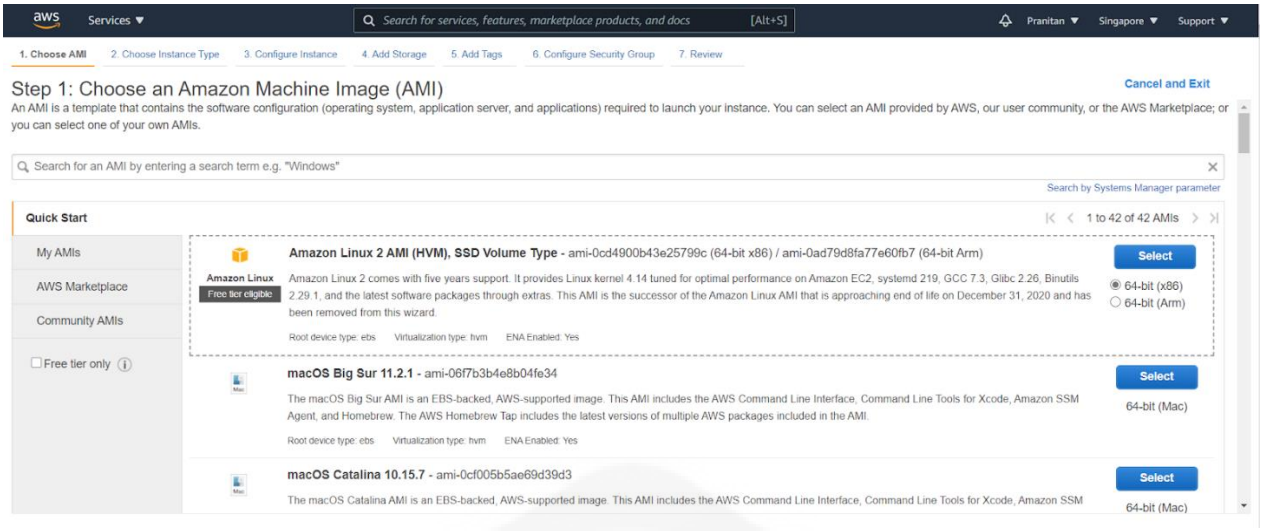
4.2 การติดตั้ง Blockchain และการ Deploy smart contract



รูปภาพที่ 28 การสร้าง Blockchain network ผ่านเว็บไซต์ AWS

การติดตั้ง Network มีการตั้งค่าดังนี้

- เลือก Version ของ Hyperledger Fabric
- กำหนด Network name เพื่อสร้างชื่อของเน็ตเวิร์คนี้
- กำหนด Voting policy ว่าจะให้มีการอนุมัติทำบางอย่างเมื่อมีการโหวตกี่เปอร์เซ็นต์ขึ้นไป
- Create Member สร้างสมาชิกในบล็อกเชน
- Hyperledger Fabric CA ตั้งค่า username และ password ของ Admin ใน CA เพื่อที่จะร้องขอ Certificate เพื่อยืนยันตัวตนในการเขียนข้อมูลหรืออ่านจาก Blockchain



รูปภาพที่ 29 การสร้าง Blockchain network ผ่านเว็บไซต์ AWS (ต่อ)

จากรูปภาพที่ 29 แสดงจากนั้นทำการติดตั้ง EC2 โดยเลือกระบบปฏิบัติการ Linux x86 เพื่อทำการตั้งค่าให้ EC2 ตัวนี้เป็น Admin ของ Blockchain จากนั้นจะสามารถ deploy smart contract ลงบน Blockchain ได้

```
docker exec cli configtxgen \
-outputCreateChannelTx /opt/home/mychannel.pb \
-profile OneOrgChannel -channelID mychannel \
--configPath /opt/home/

docker exec cli peer channel create -c mychannel \
-f /opt/home/mychannel.pb -o $ORDERER \
--cafile /opt/home/managedblockchain-tls-chain.pem --tls

docker exec cli peer channel join -b mychannel.block \
-o $ORDERER --cafile /opt/home/managedblockchain-tls-chain.pem --tls

docker exec -it cli bash
peer chaincode install -n mycc -v v0 -l node -p  ../../../key_management02/
exit

docker exec cli peer chaincode instantiate \
-o $ORDERER -C mychannel -n mycc -v v0 \
-c '{"Args":["init","UUIDTest","PublickeyTest"]}' \
--cafile /opt/home/managedblockchain-tls-chain.pem --tls

docker exec cli peer chaincode list --instantiated \
-o $ORDERER -C mychannel \
--cafile /opt/home/managedblockchain-tls-chain.pem --tls
```

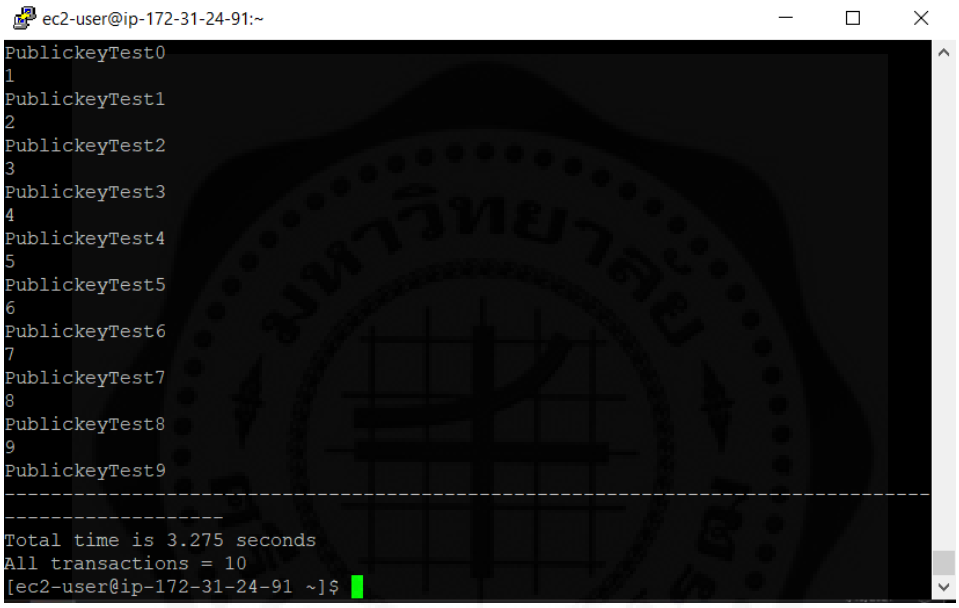
รูปภาพที่ 30 การสร้าง Blockchain network ผ่านเว็บไซต์ AWS (ต่อ)

จากรูปภาพที่ 30 แสดงจากนั้นจะใช้โค้ดนี้เพื่อสร้าง Channel และทำการ Deploy smart contract ที่เขียนไว้ลงไปใน Blockchain จากนั้นจะทำการเขียน Block แรกลงไปใน Blockchain เพื่อเป็นการเริ่มระบบอย่างสมบูรณ์

4.3 การทดสอบประสิทธิภาพของระบบ

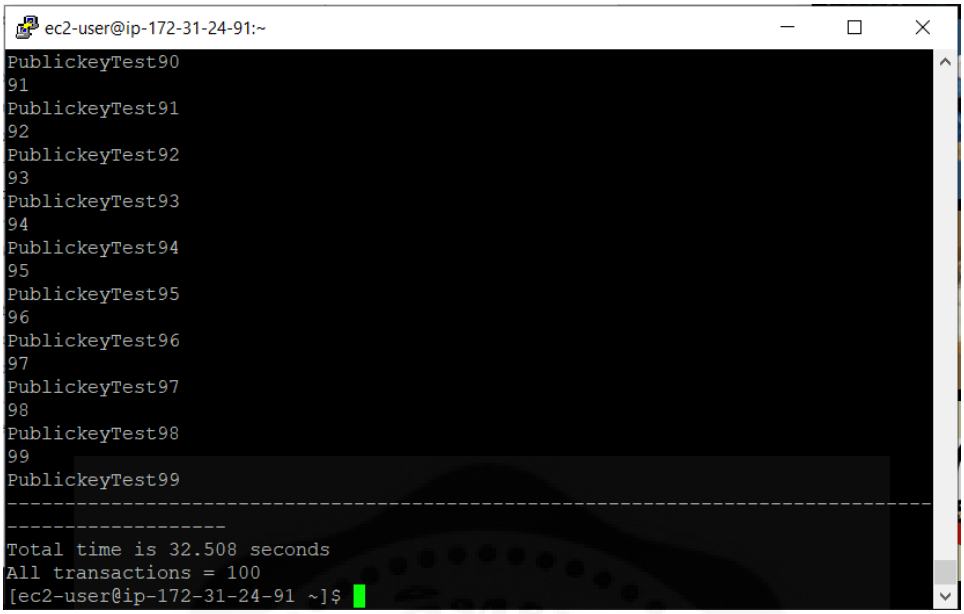
การทดสอบประสิทธิภาพของระบบ แบ่งออกเป็น 2 ส่วน คือ ทดสอบด้านการอ่านข้อมูลและทดสอบด้านการเขียนข้อมูล โดยแต่ละด้านจะแยกย่อยว่าอ่านข้อมูลจากตอนที่มีข้อมูลกี่ Node และเขียนข้อมูลลงไปในจำนวน Node ที่แตกต่างกัน

ทดสอบประสิทธิภาพของระบบด้านการอ่านข้อมูล



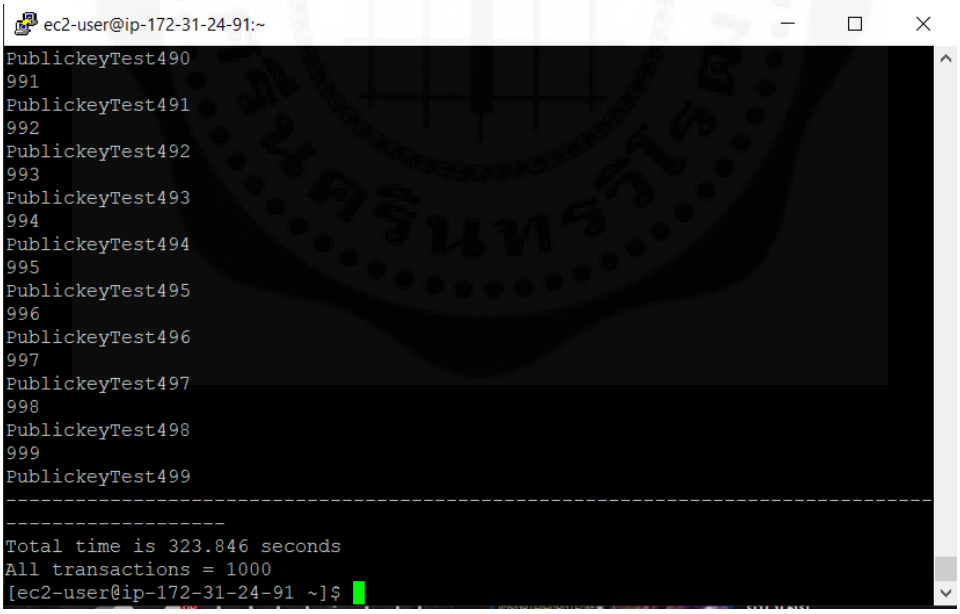
รูปภาพที่ 31 การทดสอบอ่านข้อมูลจากระบบ และทำการร้องขอ 10 รายการ

จากรูปภาพที่ 31 แสดงการทดสอบร้องขอข้อมูลจากระบบ 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 3.257 วินาที



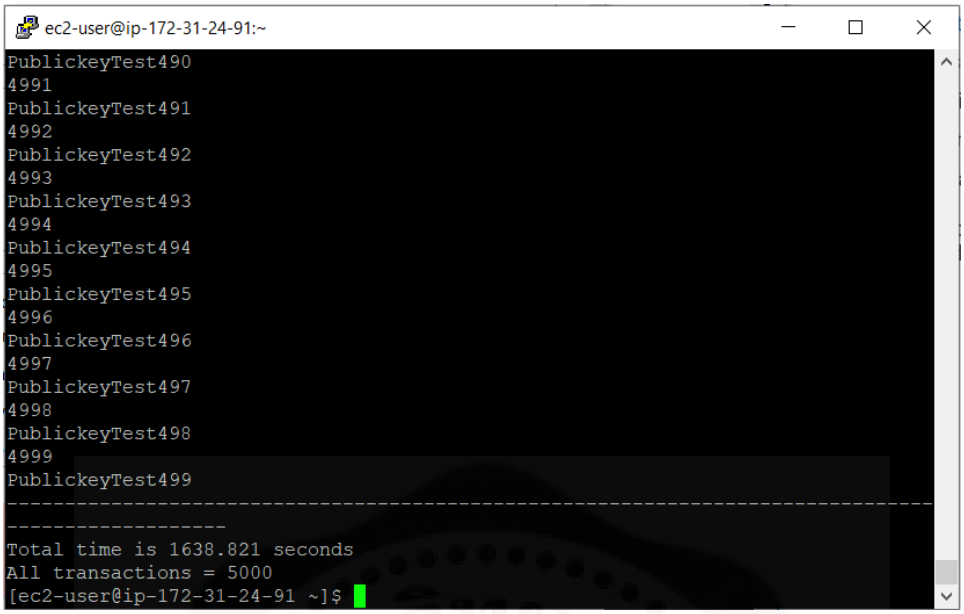
รูปภาพที่ 32 การทดสอบอ่านข้อมูลจากระบบ และทำการร้องขอ 100 รายการ

จากรูปภาพที่ 32 แสดงการทดสอบร้องขอข้อมูลจากระบบ 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 32.529 วินาที



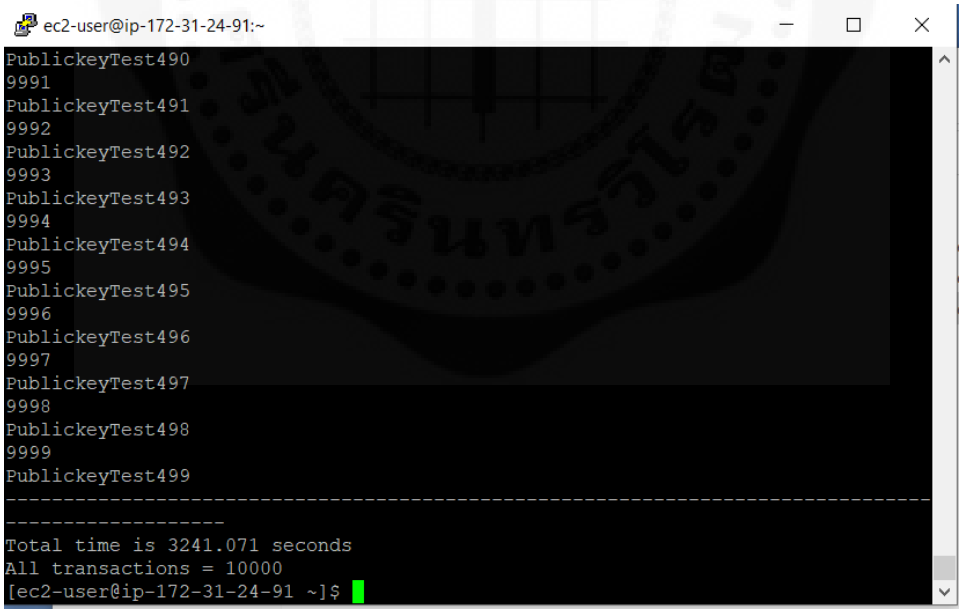
รูปภาพที่ 33 การทดสอบอ่านข้อมูลจากระบบ และทำการร้องขอ 1000 รายการ

จากรูปภาพที่ 33 แสดงการทดสอบร้องขอข้อมูลจากระบบ 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 324.324 วินาที



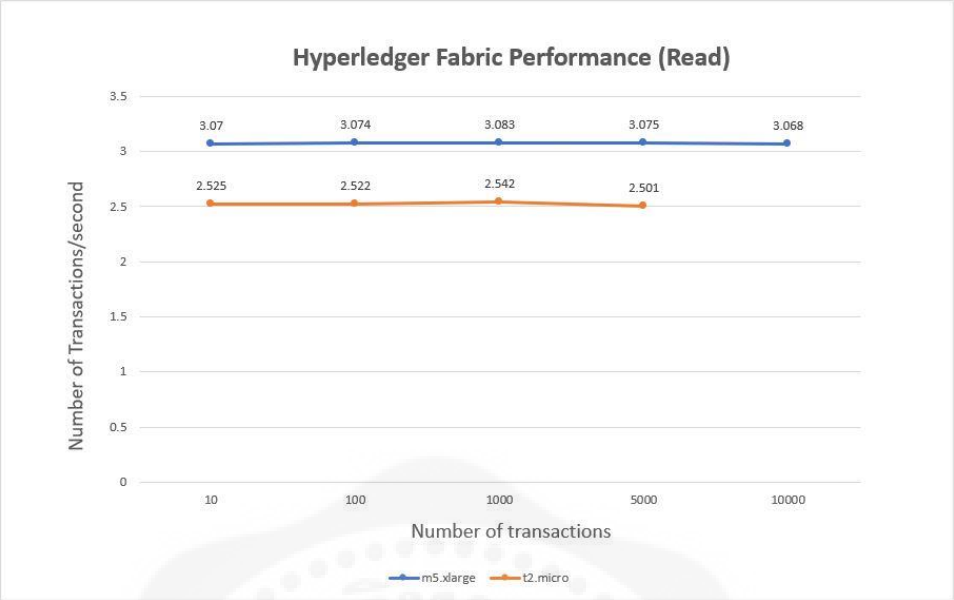
รูปภาพที่ 34 การทดสอบอ่านข้อมูลจากระบบ และทำการร้องขอ 5000 รายการ

จากรูปภาพที่ 34 แสดงการทดสอบร้องขอข้อมูลจากระบบ 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 1626.197 วินาที



รูปภาพที่ 35 การทดสอบอ่านข้อมูลจากระบบ และทำการร้องขอ 10000 รายการ

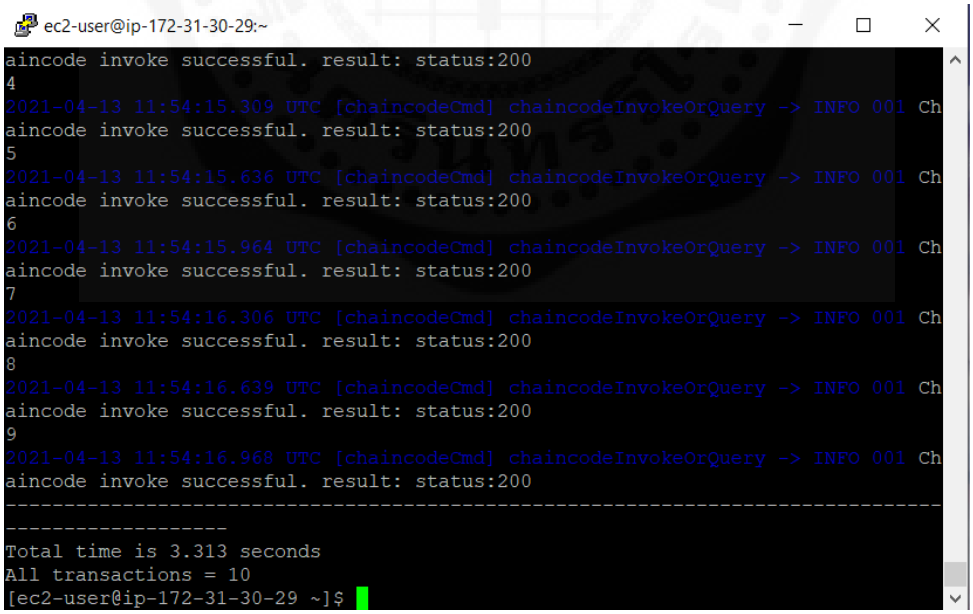
จากรูปภาพที่ 35 แสดงการทดสอบร้องขอข้อมูลจากระบบ 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 3259.138 วินาที



รูปภาพที่ 36 กราฟเส้นแสดงประสิทธิภาพด้านการอ่านข้อมูลจากระบบ

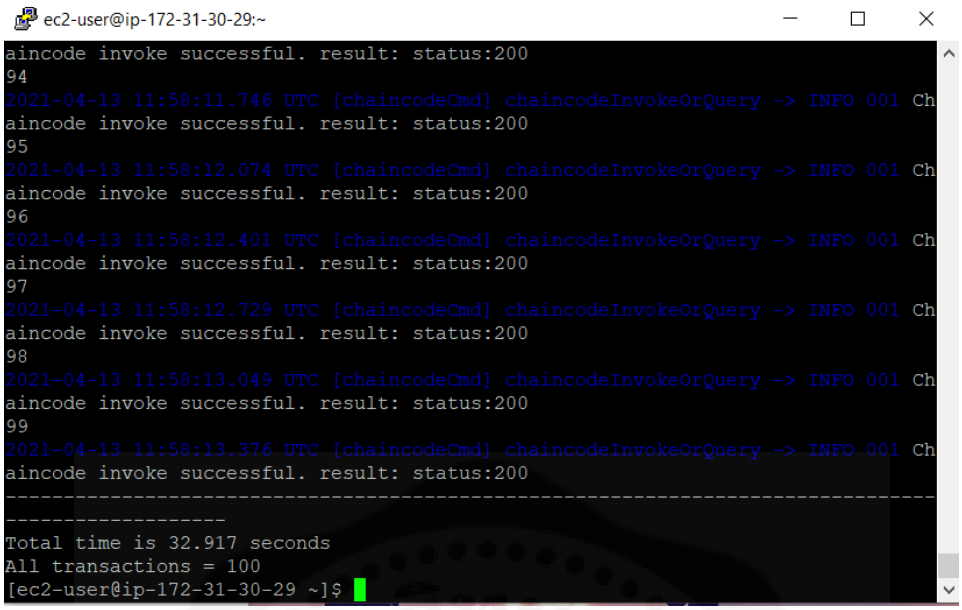
จากรูปภาพที่ 36 แสดงกราฟเปรียบเทียบประสิทธิภาพการอ่านข้อมูลจากระบบระหว่าง m5.xlarge และ t2.micro

ทดสอบประสิทธิภาพของระบบด้านการเขียนข้อมูล



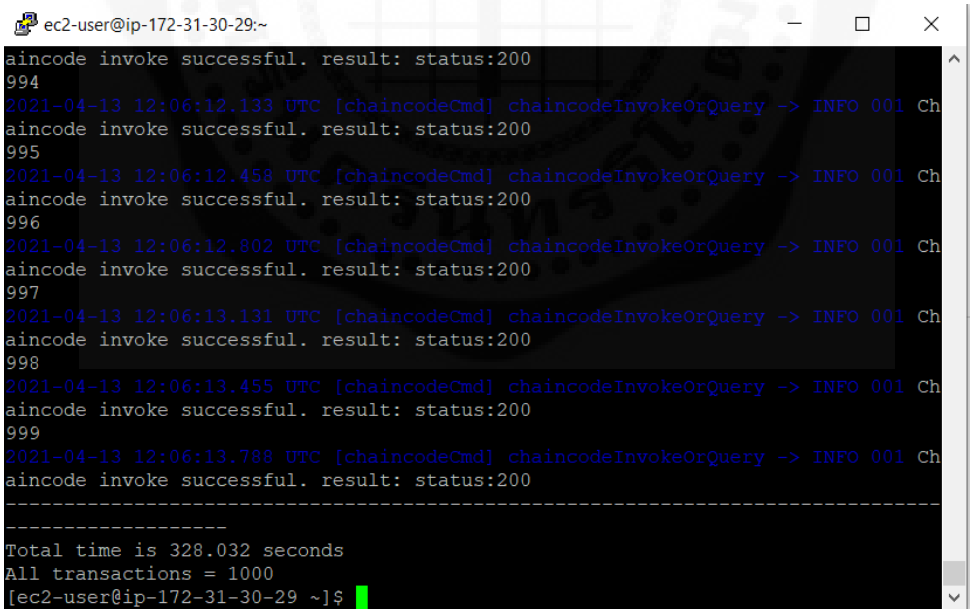
รูปภาพที่ 37 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 1 Node และเขียนข้อมูล 10 รายการ

จากรูปภาพที่ 37 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 3.307 วินาที



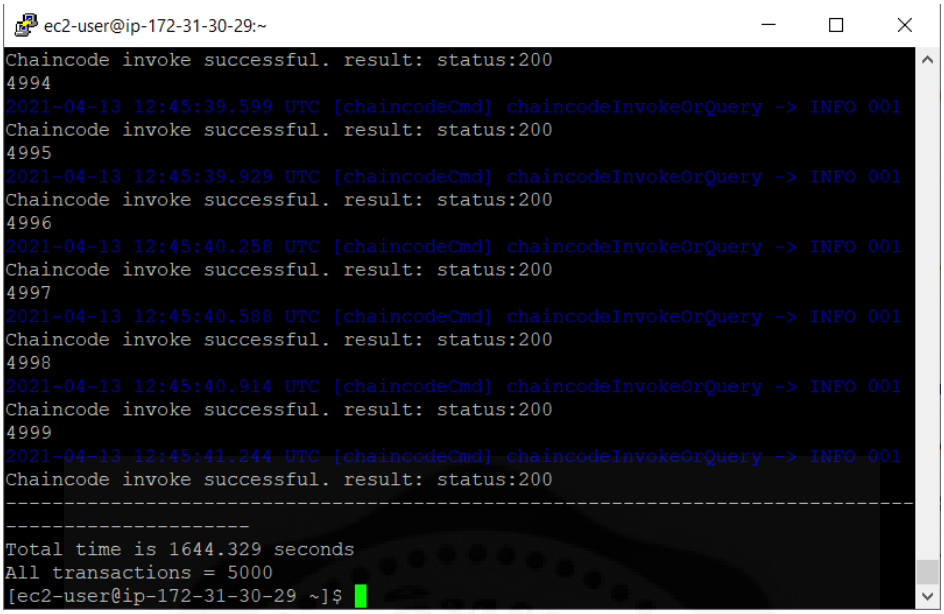
รูปภาพที่ 38 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 1 Node และเขียนข้อมูล 100 รายการ

จากรูปภาพที่ 38 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 32.817 วินาที



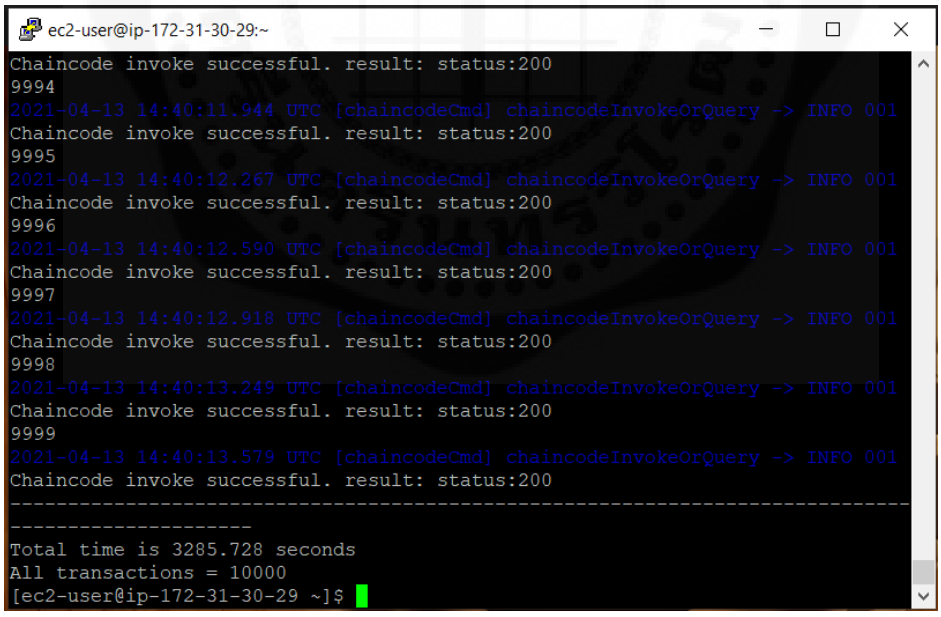
รูปภาพที่ 39 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 1 Node และเขียนข้อมูล 1000 รายการ

จากรูปภาพที่ 39 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 329.844 วินาที



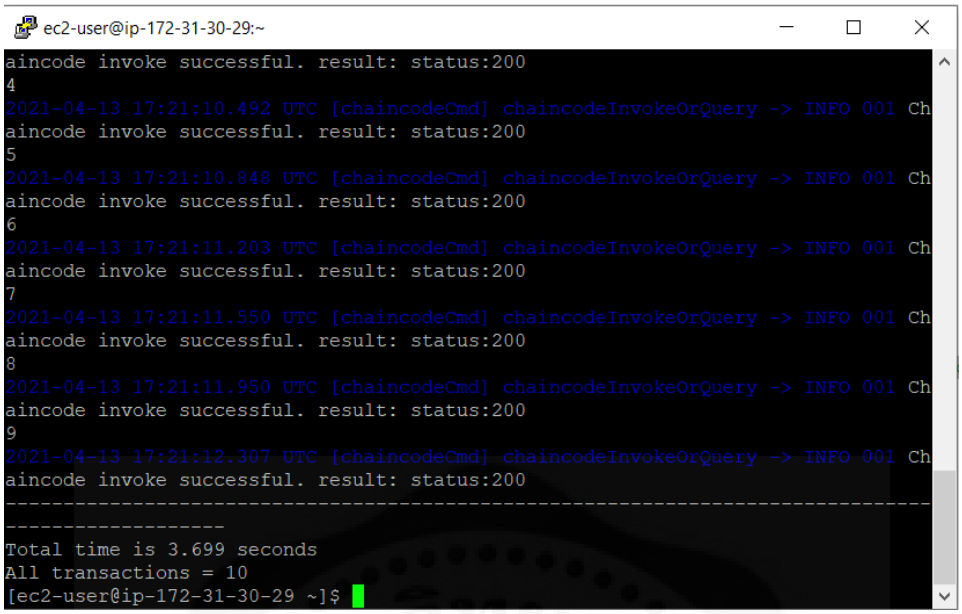
รูปภาพที่ 40 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 1 Node และเขียนข้อมูล 5000 รายการ

จากรูปภาพที่ 40 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 1639.040 วินาที



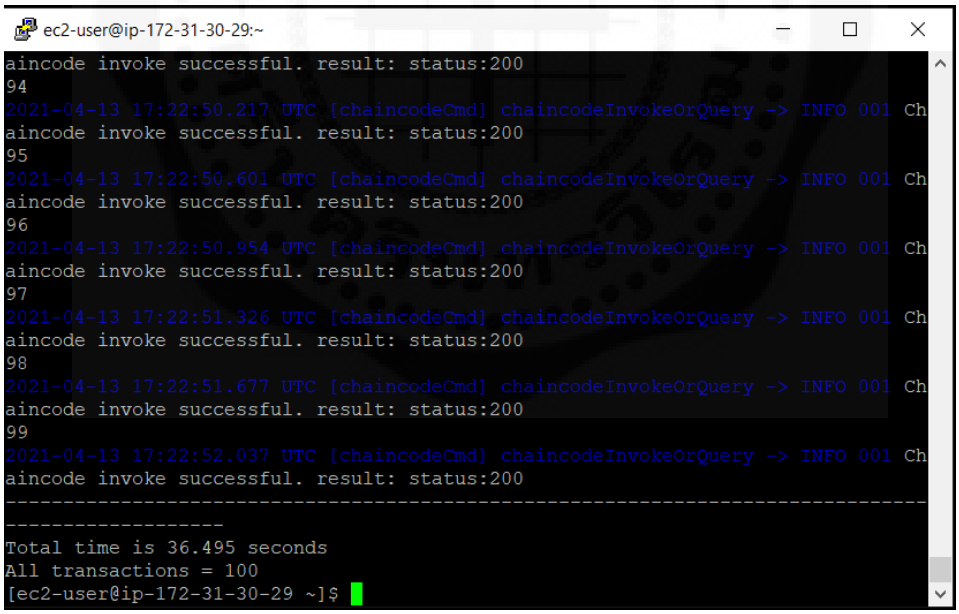
รูปภาพที่ 41 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 1 Node และเขียนข้อมูล 10000 รายการ

จากรูปภาพที่ 41 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 3286.377 วินาที



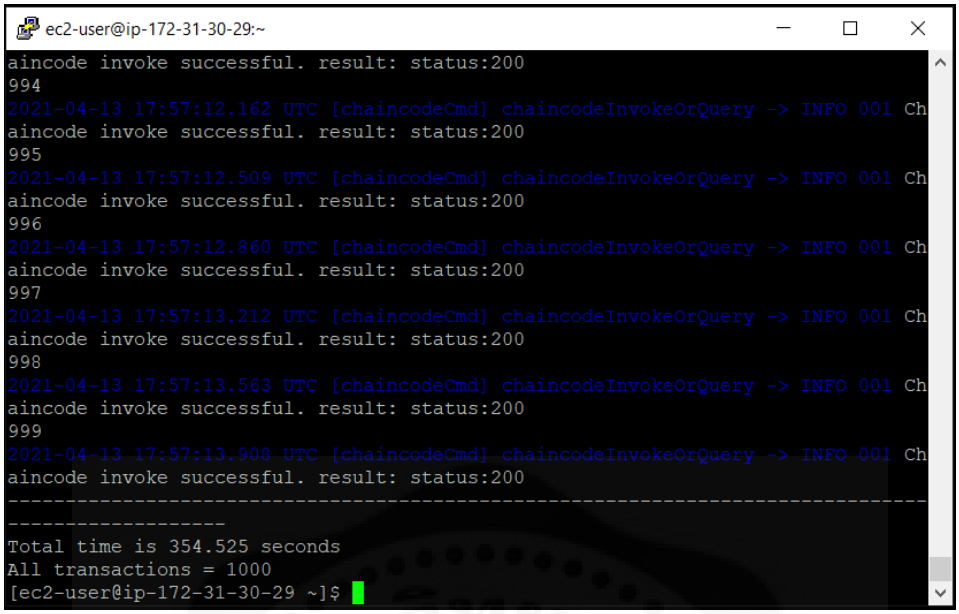
รูปภาพที่ 42 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 2 Node และเขียนข้อมูล 10 รายการ

จากรูปภาพที่ 42 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 3.647 วินาที



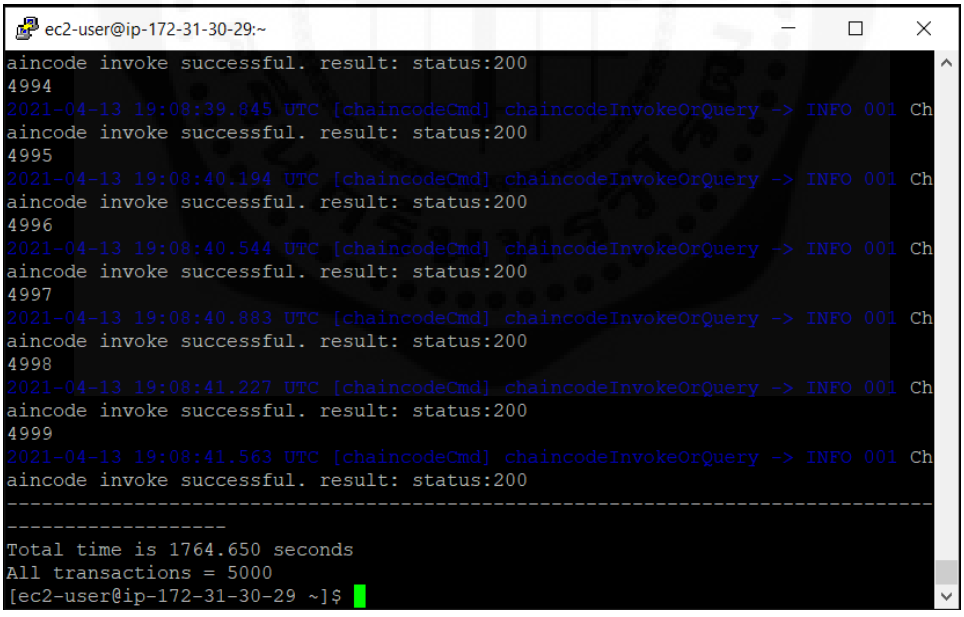
รูปภาพที่ 43 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 2 Node และเขียนข้อมูล 100 รายการ

จากรูปภาพที่ 43 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 35.944 วินาที



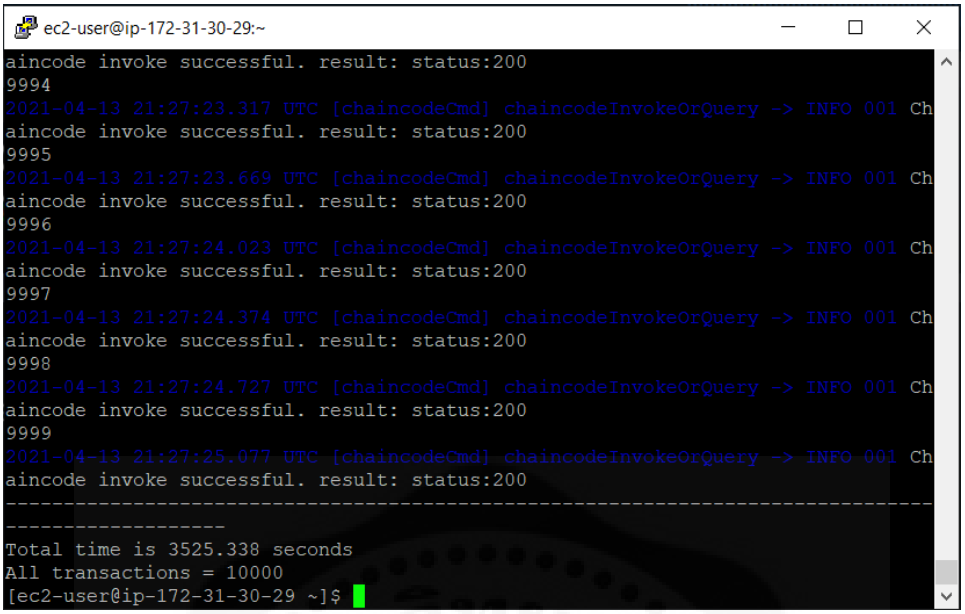
รูปภาพที่ 44 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 2 Node และเขียนข้อมูล 1000 รายการ

จากรูปภาพที่ 44 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 355.981 วินาที



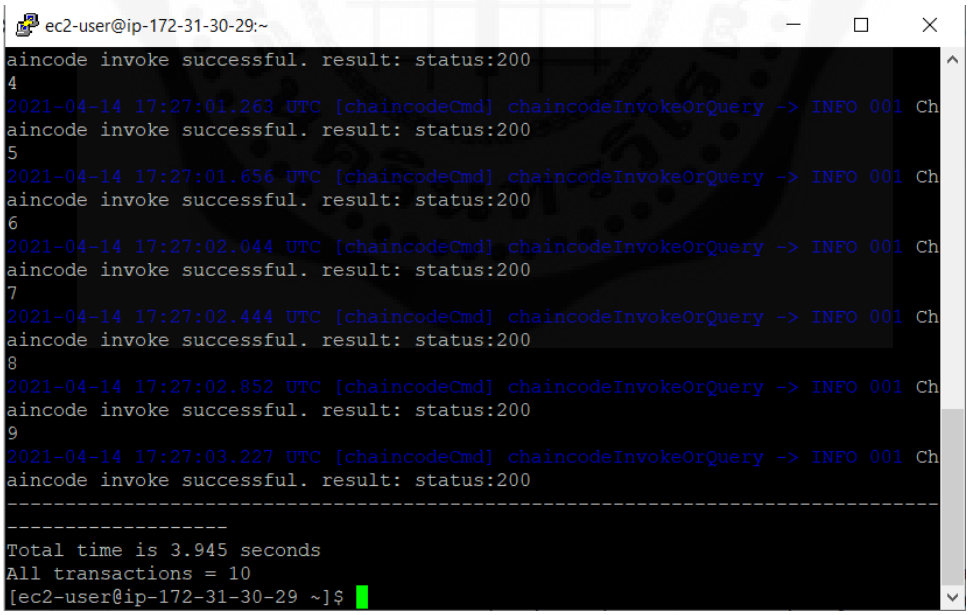
รูปภาพที่ 45 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 2 Node และเขียนข้อมูล 5000 รายการ

จากรูปภาพที่ 45 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 1761.631 วินาที



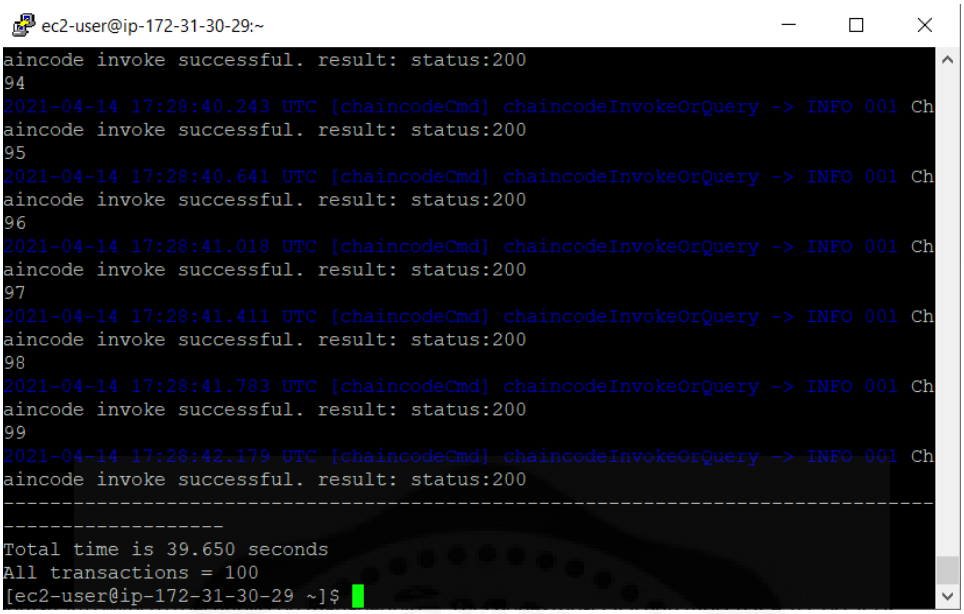
รูปภาพที่ 46 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 2 Node และเขียนข้อมูล 10000 รายการ

จากรูปภาพที่ 46 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 3539.397 วินาที



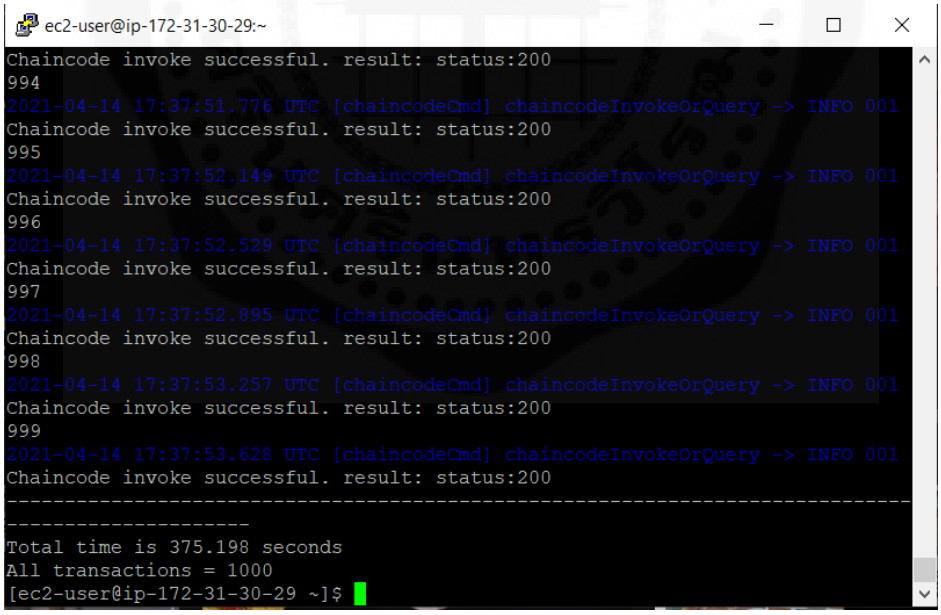
รูปภาพที่ 47 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 3 Node และเขียนข้อมูล 10 รายการ

จากรูปภาพที่ 47 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 3.862 วินาที



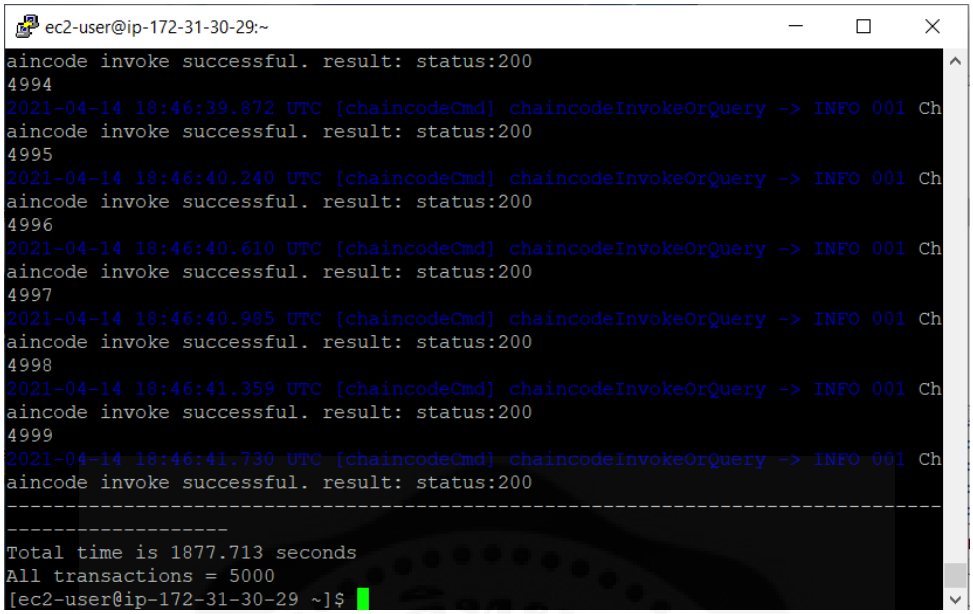
รูปภาพที่ 48 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 3 Node และเขียนข้อมูล 100 รายการ

จากรูปภาพที่ 48 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 38.913 วินาที



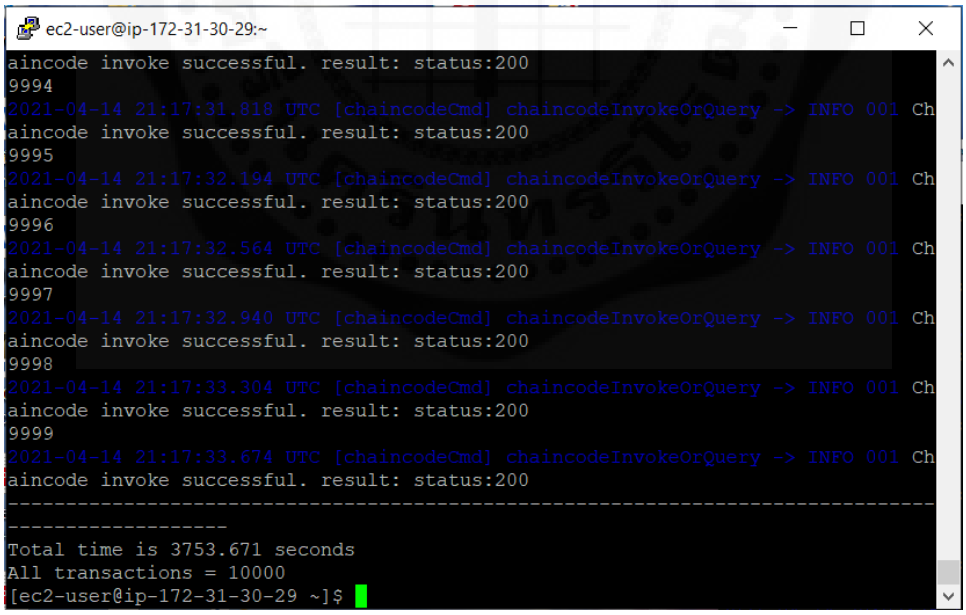
รูปภาพที่ 49 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 3 Node และเขียนข้อมูล 1000 รายการ

จากรูปภาพที่ 49 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 374.909 วินาที



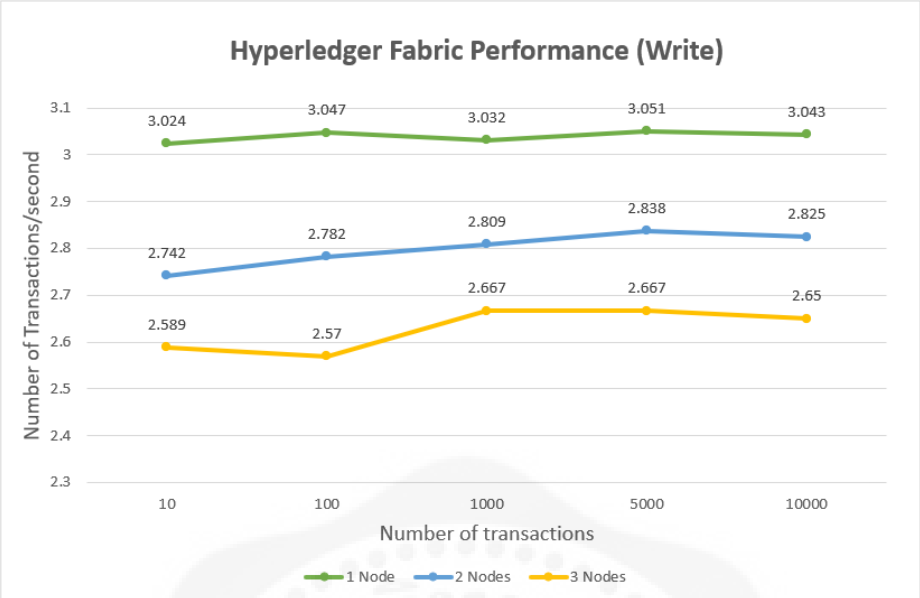
รูปภาพที่ 50 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 3 Node และเขียนข้อมูล 5000 รายการ

จากรูปภาพที่ 50 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 1874.523 วินาที



รูปภาพที่ 51 การทดสอบเขียนข้อมูลเข้าไประบบ กรณีมี 3 Node และเขียนข้อมูล 10000 รายการ

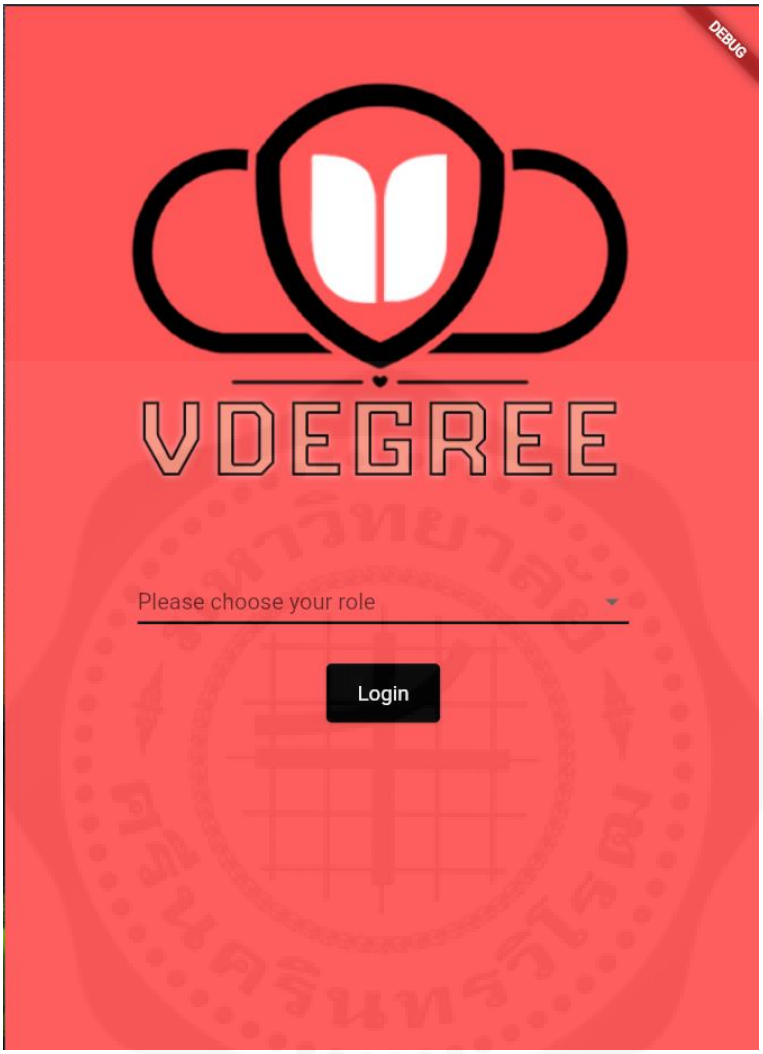
จากรูปภาพที่ 51 แสดงการทดสอบร้องขอให้ระบบเขียนข้อมูล 3 ครั้ง เฉลี่ยใช้เวลาทั้งสิ้น 3773.310 วินาที



รูปภาพที่ 52 กราฟเส้นแสดงประสิทธิภาพด้านการเขียนข้อมูลของ 1 Node, 2 Nodes และ 3 Nodes

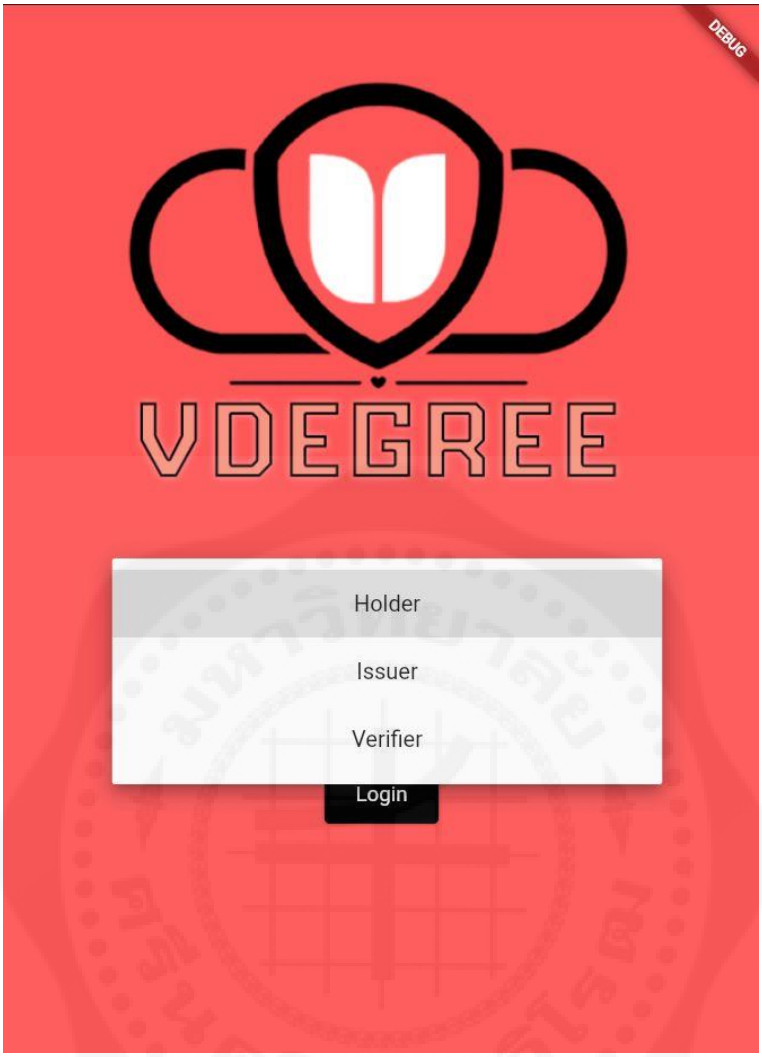
จากรูปภาพที่ 52 แสดงกราฟเปรียบเทียบประสิทธิภาพการเขียนข้อมูลเข้าไปในระบบระหว่าง 1 Node, 2 Nodes และ 3 Nodes

4.4 Application สำหรับติดต่อใช้งานระบบ



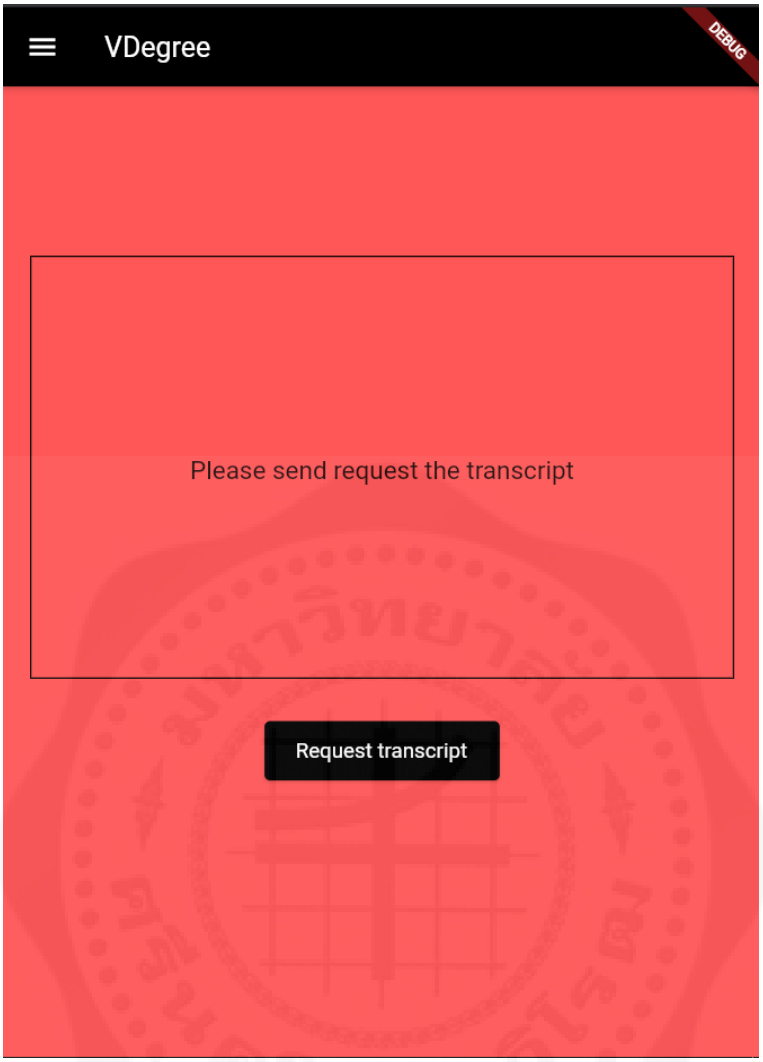
รูปภาพที่ 53 หน้า Login เพื่อเข้าสู่ระบบ

จากรูปภาพที่ 53 แสดงหน้า Login เพื่อเข้าสู่ระบบสำหรับผู้ใช้ โดยมี Dropdown เพื่อเลือก Role ของผู้ใช้



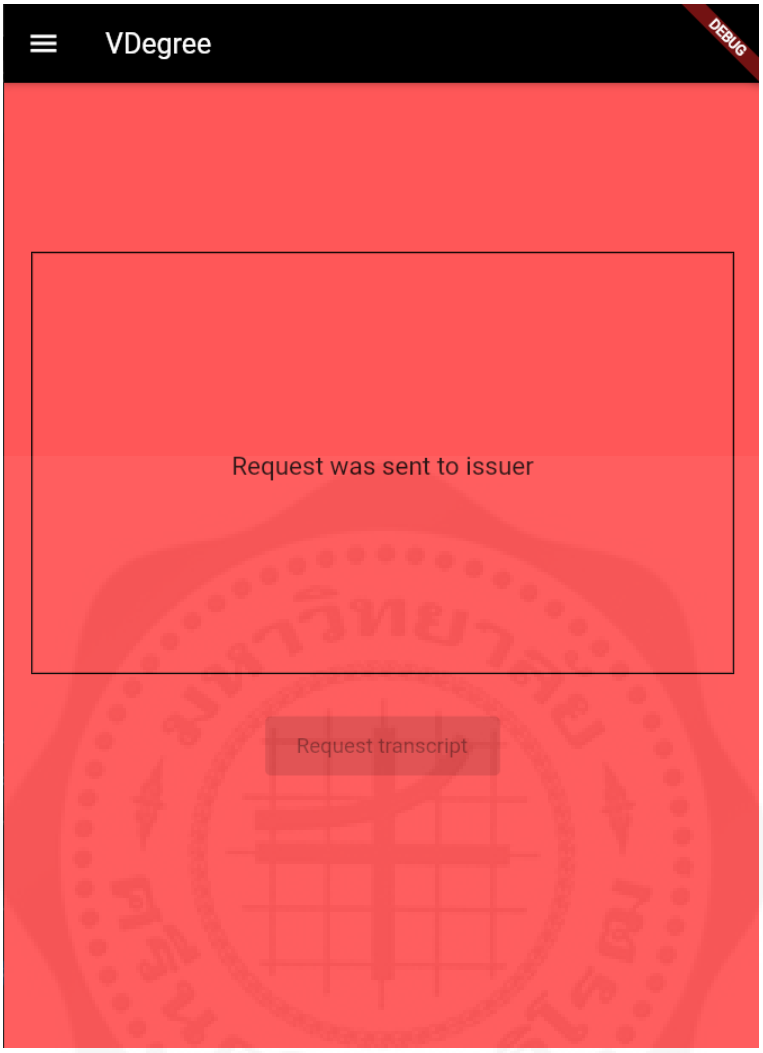
รูปภาพที่ 54 ข้อมูลใน Dropdown ของหน้า Login

จากรูปภาพที่ 54 แสดง Dropdown สำหรับเลือก Role ของผู้ใช้เพื่อเข้าสู่ระบบในหน้า Login โดย Role ที่มีให้เลือก คือ Holder, Issuer และ Verifier



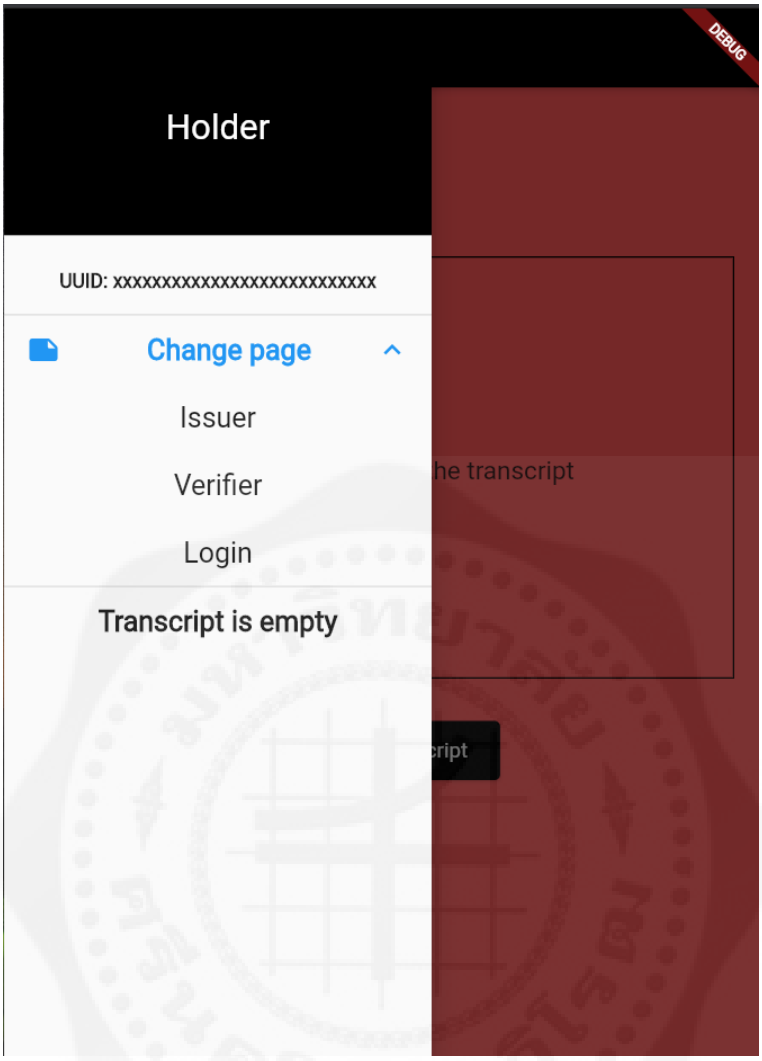
รูปภาพที่ 55 หน้าแรกของ Holder

จากรูปภาพที่ 55 แสดงหน้าแรกของ Holder โดยมีช่องสำหรับแสดงข้อมูล และมีปุ่มสำหรับกดส่งคำร้องขอ Transcript ไปหา Issuer



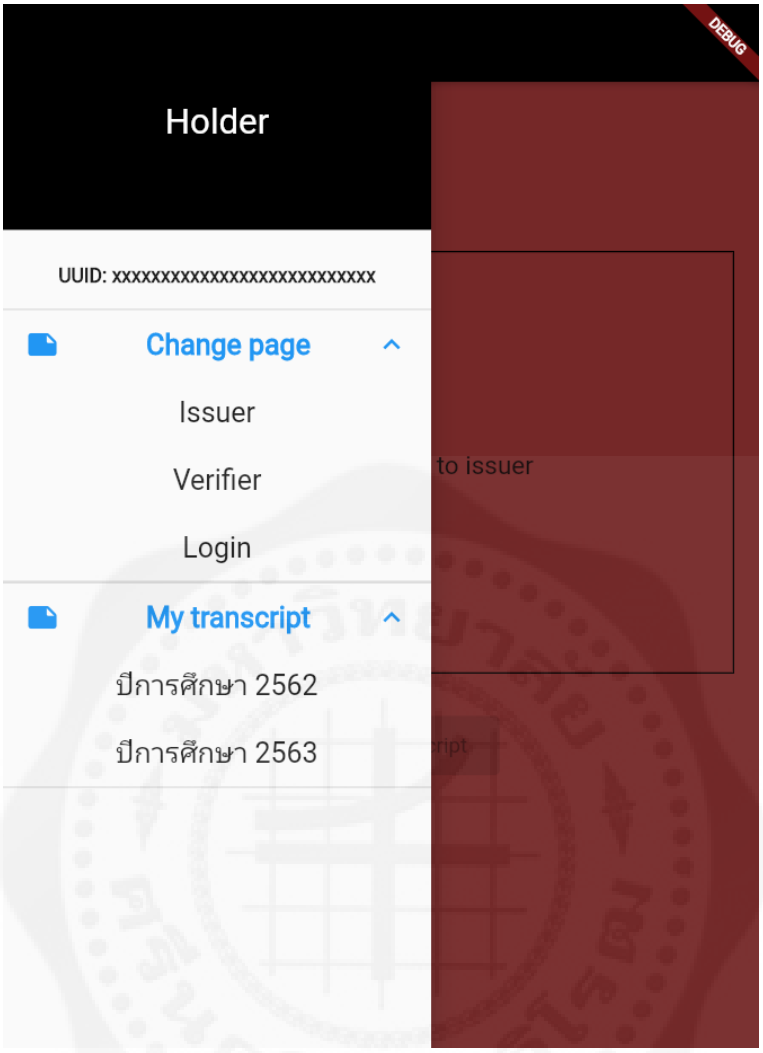
รูปภาพที่ 56 หน้าแรกของ Holder (ต่อ)

จากรูปภาพที่ 56 แสดงหลังจากส่งคำร้องขอ Transcript แล้ว จะมีข้อความขึ้นที่ช่องแสดงข้อมูลว่า Request was sent และปุ่มจะไม่สามารถกดได้



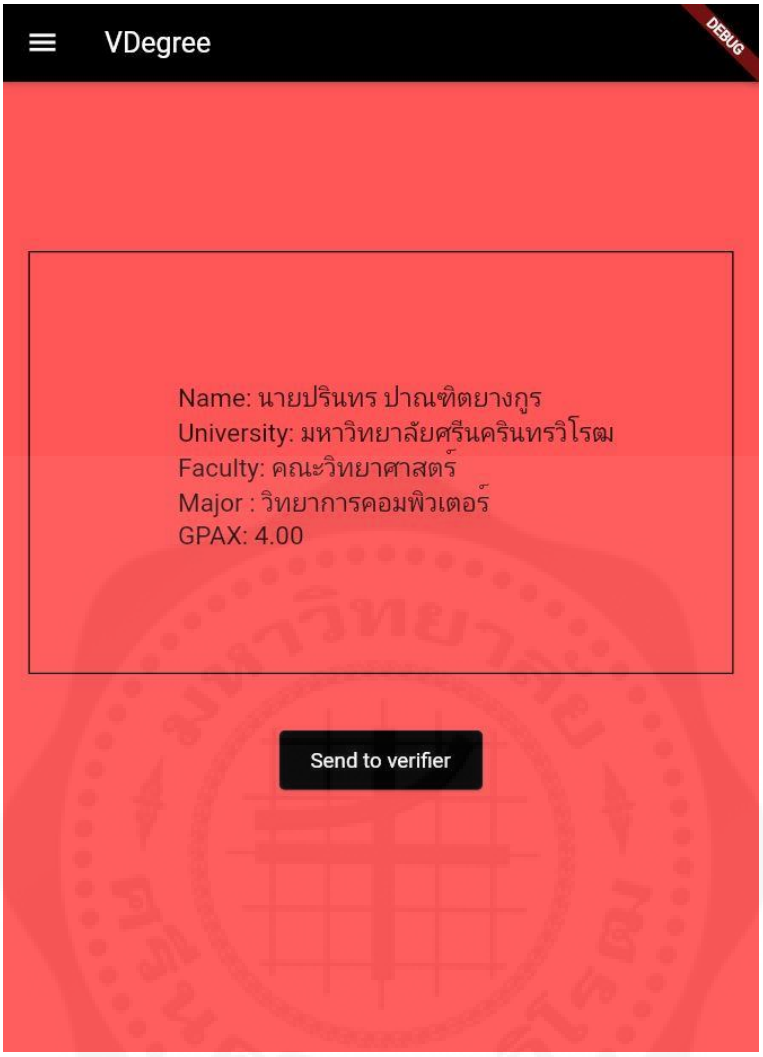
รูปภาพที่ 57 แถบเมนูของ Holder ในกรณีที่ไม่มีข้อมูล

จากรูปภาพที่ 57 แสดงแถบเมนูของ Holder โดยส่วนบนจะแสดง Role ของผู้ใช้ ส่วนต่อมาเป็น UUID ของผู้ใช้ และในส่วนสุดท้าย คือ Dropdown โดยมีทั้งหมด 2 Dropdown คือ Dropdown สำหรับเปลี่ยนไปยังหน้าอื่น ซึ่งสามารถไปที่หน้า Issuer, Verifier และ Login ได้ และ Dropdown ที่แสดงรายการ Transcript ที่ผู้ใช้มีอยู่ ซึ่งในกรณีนี้เป็นกรณีที่ยังไม่มี Transcript อยู่



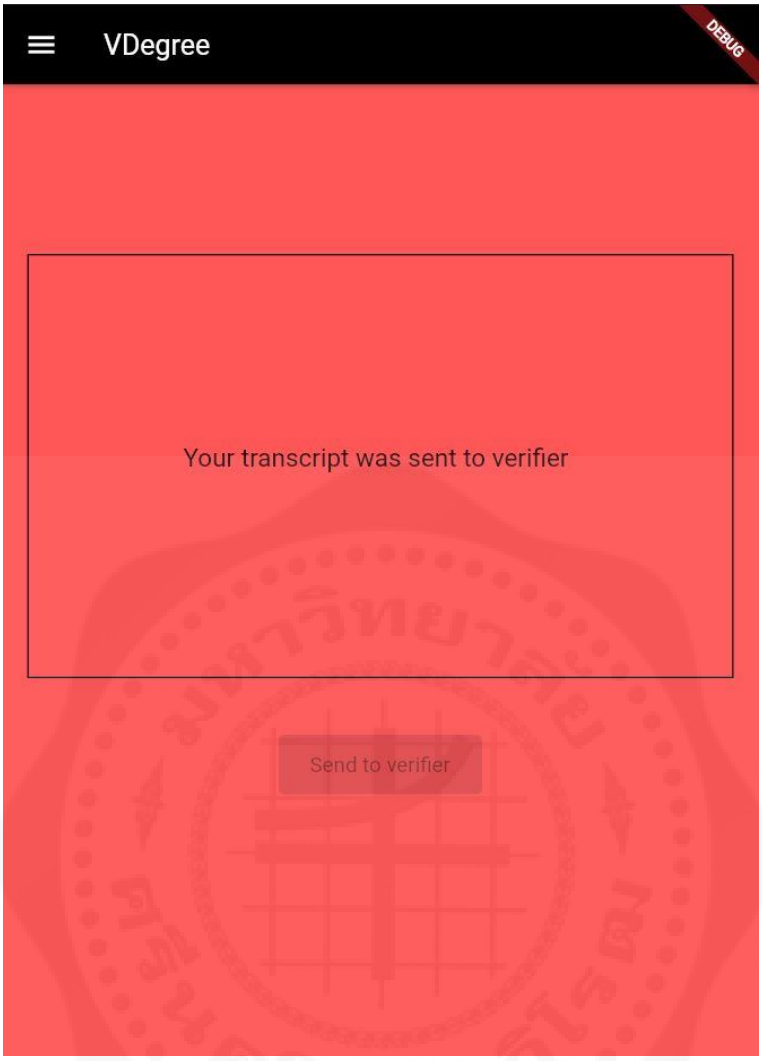
รูปภาพที่ 58 แถบเมนูของ Holder ในกรณีที่มีข้อมูล

จากรูปภาพที่ 58 แสดงแถบเมนูของ Holder โดยส่วนบนจะแสดง Role ของผู้ใช้ ส่วนต่อมาเป็น UUID ของผู้ใช้ และในส่วนสุดท้าย คือ Dropdown โดยมีทั้งหมด 2 Dropdown คือ Dropdown สำหรับเปลี่ยนไปยังหน้าอื่น ซึ่งสามารถไปที่หน้า Issuer, Verifier และ Login ได้ และ Dropdown ที่แสดงรายการ Transcript ที่ผู้ใช้มีอยู่ ซึ่งในกรณีนี้เป็นกรณีที่ยังมี Transcript อยู่



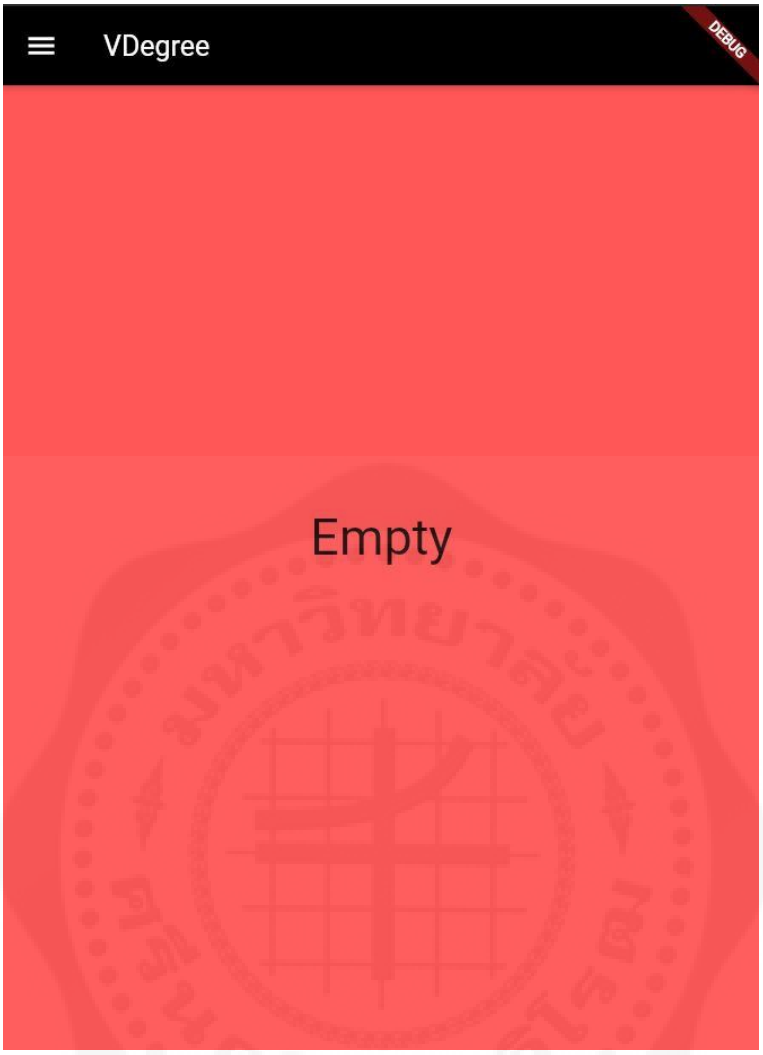
รูปภาพที่ 59 หน้าแสดงข้อมูลใน Transcript ของ Holder

จากรูปภาพที่ 59 แสดงหลังจาก Issuer อนุมัติ Transcript ให้ ข้อมูลของ Transcript จะแสดงอยู่ในกล่องข้อมูล และมีปุ่มสำหรับกดส่ง Transcript ไปให้ Verifier



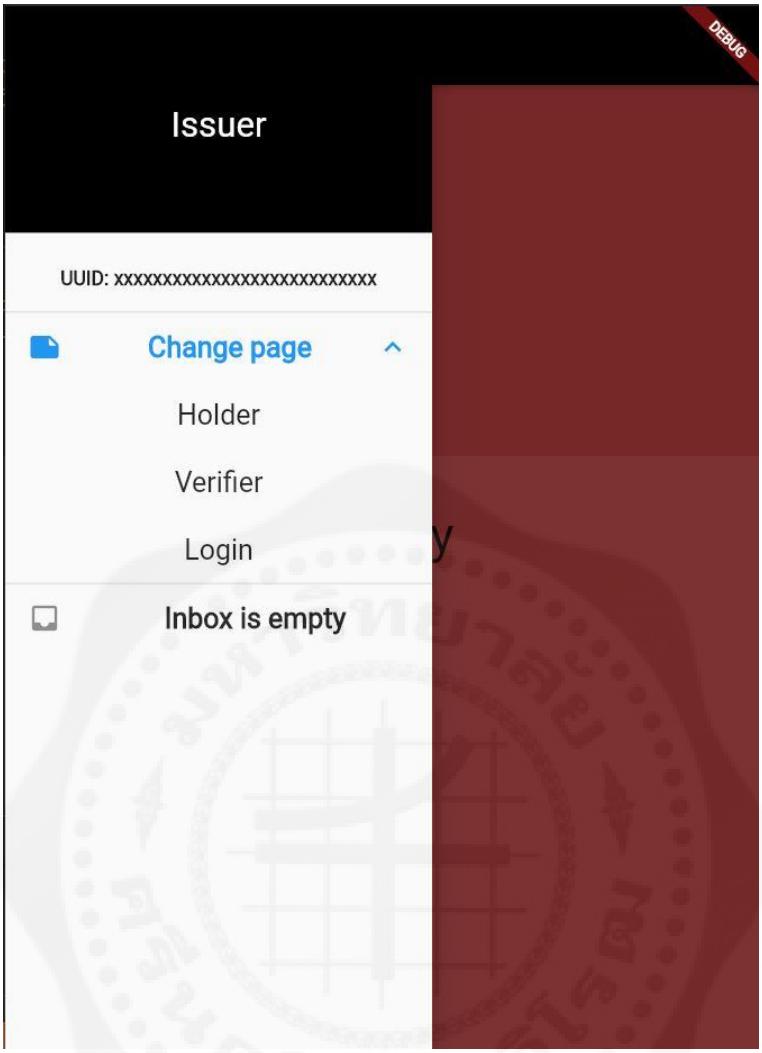
รูปภาพที่ 60 หน้าแสดงข้อมูลใน Transcript ของ Holder (ต่อ)

จากรูปภาพที่ 60 แสดงหลังจากกดส่ง Transcript ให้ Verifier แล้ว จะมีข้อความขึ้นที่ช่องแสดงข้อมูลว่า Your transcript was sent to verifier และในส่วนของปุ่มจะไม่สามารถกดได้



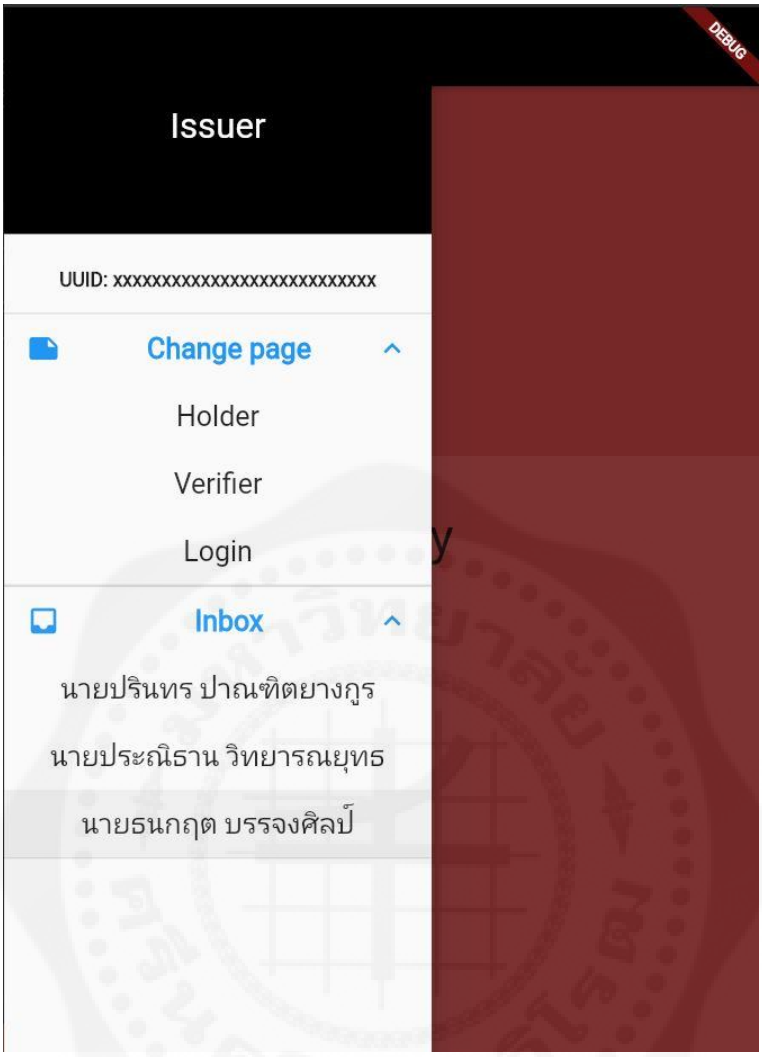
รูปภาพที่ 61 หน้าแรกของ Issuer

จากรูปภาพที่ 61 แสดงหน้าแรกของ Issuer ในกรณีที่ยังไม่คำร้องขอ Transcript จาก Holder เข้ามา บนหน้าแอปพลิเคชันจะแสดงข้อความว่า Empty



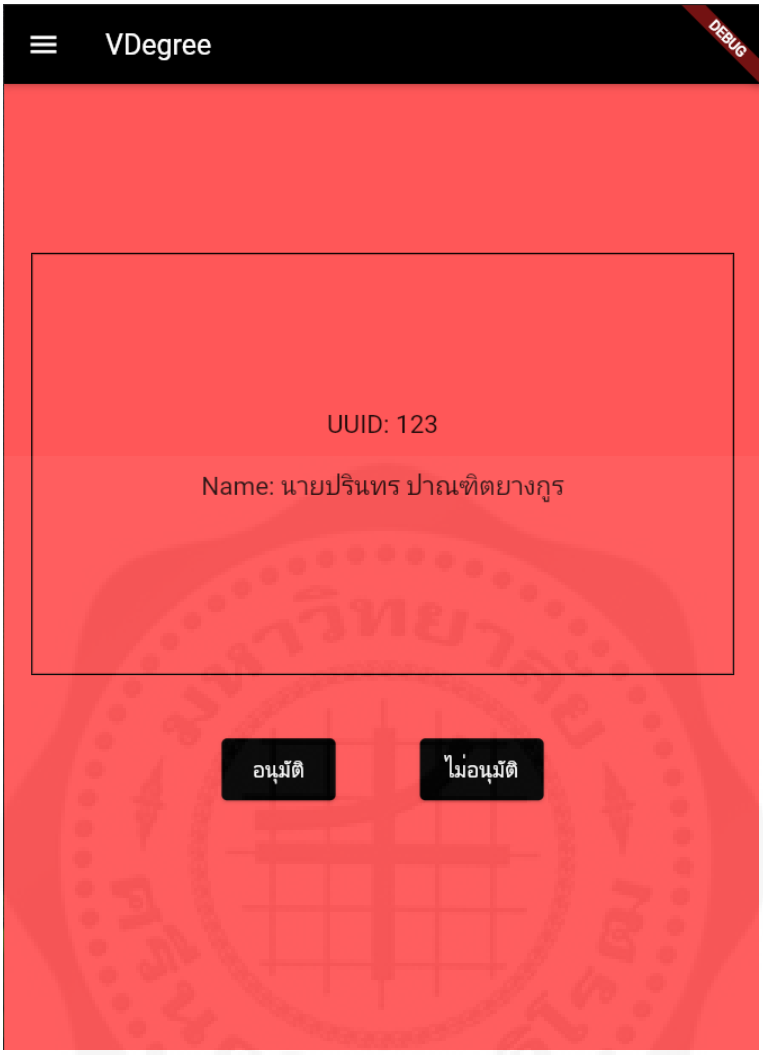
รูปภาพที่ 62 แถบเมนูของ Issuer ในกรณีที่ไม่มีข้อมูล

จากรูปภาพที่ 62 แสดงแถบเมนูของ Issuer โดยส่วนบนจะแสดง Role ของผู้ใช้ ส่วนต่อมาเป็น UUID ของผู้ใช้ และในส่วนสุดท้าย คือ Dropdown โดยมีทั้งหมด 2 Dropdown คือ Dropdown สำหรับเปลี่ยนไปยังหน้าอื่น ซึ่งสามารถไปที่หน้า Holder, Verifier และ Login ได้ และ Dropdown ที่แสดงรายการคำร้องขอ Transcript จาก Holder ที่ส่งเข้ามา ซึ่งในกรณีนี้เป็นกรณีที่ยังไม่มีคำร้องขอ Transcript เข้ามา



รูปภาพที่ 63 แถบเมนูของ Issuer ในกรณีที่มีข้อมูล

จากรูปภาพที่ 63 แสดงแถบเมนูของ Issuer โดยส่วนบนจะแสดง Role ของผู้ใช้ ส่วนต่อมาเป็น UUID ของผู้ใช้ และในส่วนสุดท้าย คือ Dropdown โดยมีทั้งหมด 2 Dropdown คือ Dropdown สำหรับเปลี่ยนไปยังหน้าอื่น ซึ่งสามารถไปที่หน้า Holder, Verifier และ Login ได้ และ Dropdown ที่แสดงรายการคำร้องขอ Transcript จาก Holder ที่ส่งเข้ามา ซึ่งในกรณีนี้เป็นกรณีที่ยังมีคำร้องขอ Transcript เข้ามา

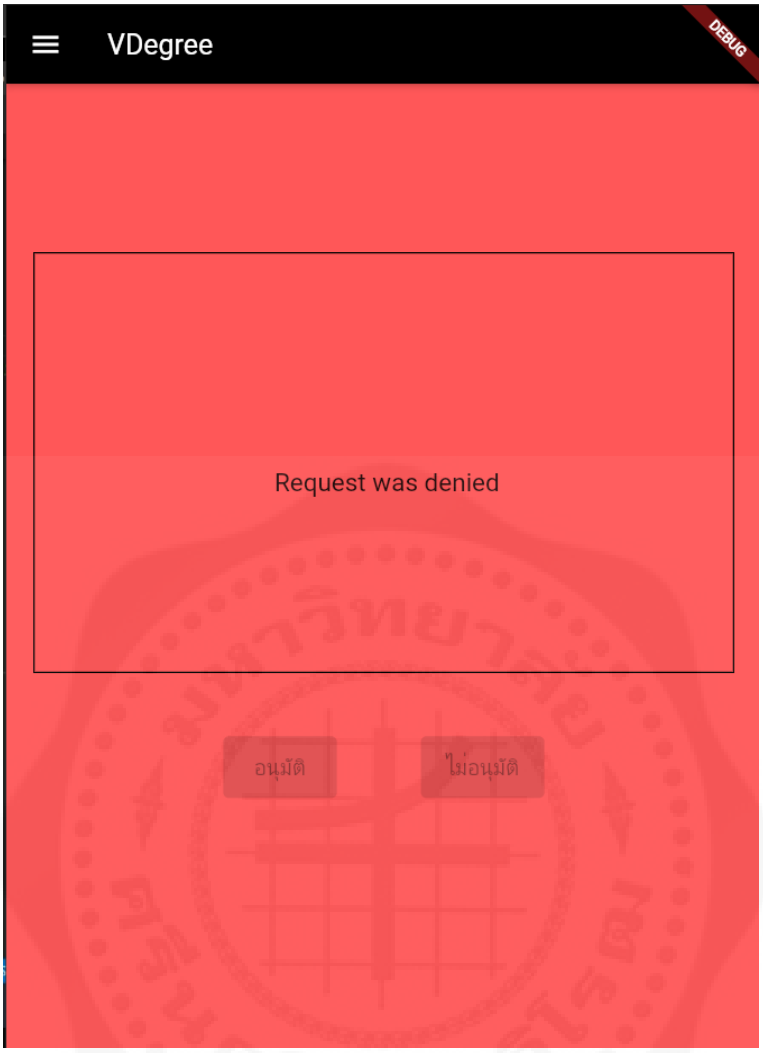


รูปภาพที่ 64 หน้าแสดงรายละเอียดคำร้องขอ Transcript

จากรูปภาพที่ 64 แสดงหลังจากกดที่รายการคำร้องขอ Transcript ในแถบเมนูแล้ว จะแสดงรายละเอียดต่างๆ ของผู้ที่ส่งคำร้องขอ ประกอบไปด้วย UUID และชื่อของ Holder และให้ Issuer เลือกได้ว่าจะอนุมัติหรือไม่

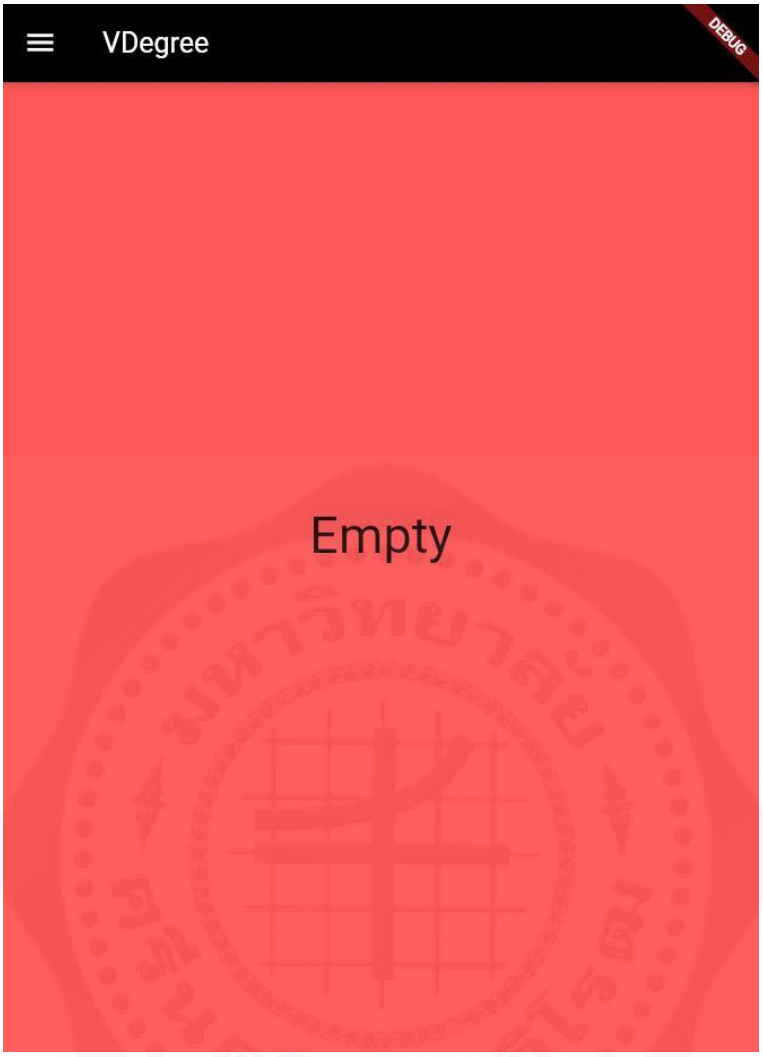
รูปภาพที่ 65 หน้าแสดงตัวอย่างกรณีที่อนุมัติ Transcript

จากรูปภาพที่ 65 แสดงหน้าแสดงตัวอย่างกรณีที่ Issuer กดอนุมัติ Transcript จะมีข้อความขึ้นว่า Request was approved และปุ่มจะไม่สามารถกดได้



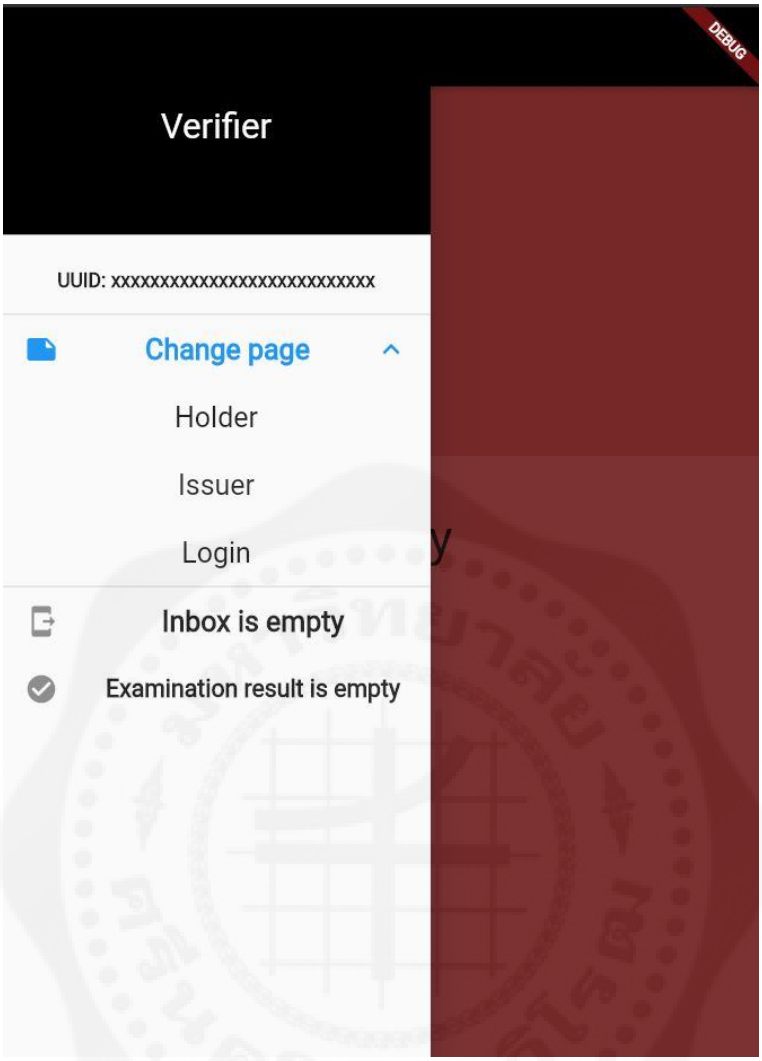
รูปภาพที่ 66 หน้าแสดงตัวอย่างกรณีที่ไม่อนุมัติ Transcript

จากรูปภาพที่ 66 แสดงหน้าแสดงตัวอย่างกรณีที่ Issuer กดไม่อนุมัติ Transcript จะมีข้อความขึ้นว่า Request was denied และปุ่มจะไม่สามารถกดได้



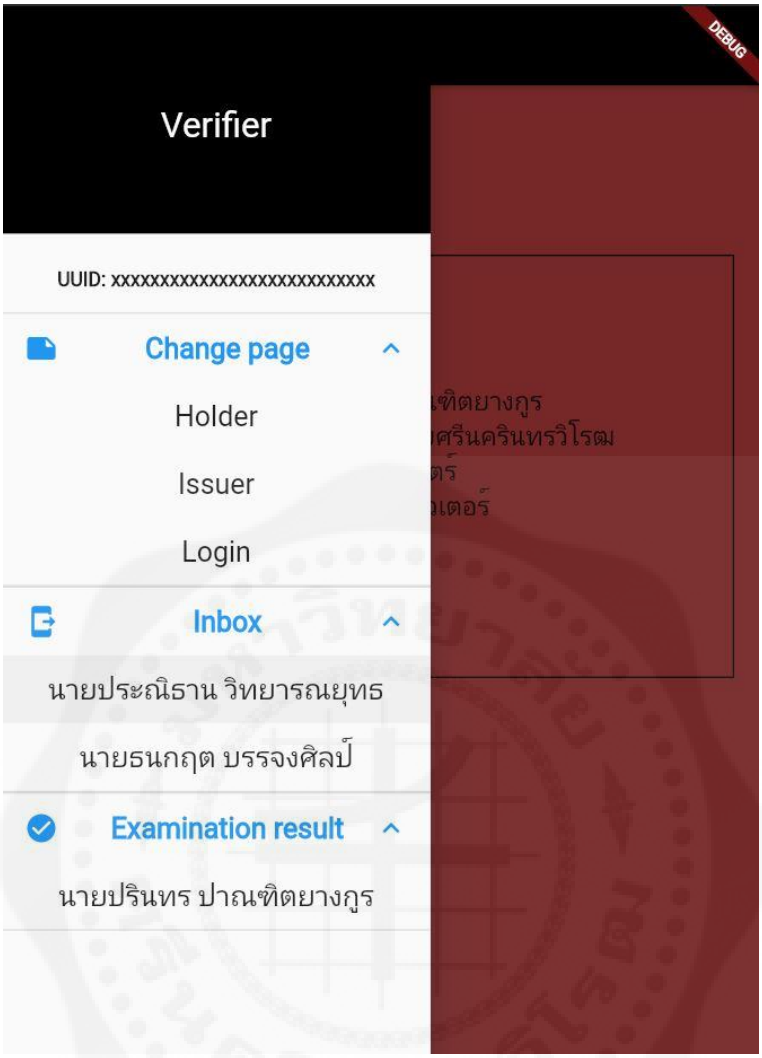
รูปภาพที่ 67 หน้าแรกของ Verifier

จากรูปภาพที่ 67 แสดงหน้าแรกของ Verifier ในกรณีที่ยังไม่มี Transcript จาก Holder ส่งเข้ามา บนหน้าแอปพลิเคชันจะแสดงข้อความว่า Empty



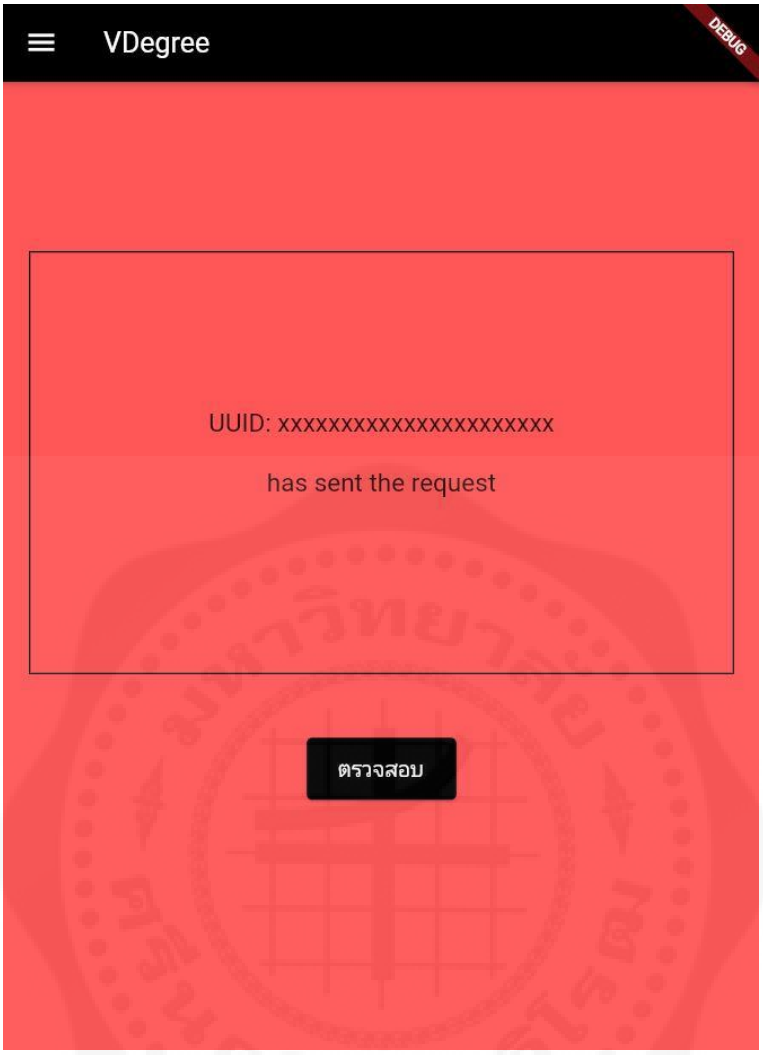
รูปภาพที่ 68 แถบเมนูของ Verifier ในกรณีที่ไม่มีข้อมูล

จากรูปภาพที่ 68 แสดงแถบเมนูของ Verifier โดยส่วนบนจะแสดง Role ของผู้ใช้ ส่วนต่อมาเป็น UUID ของผู้ใช้ และในส่วนสุดท้าย คือ Dropdown ซึ่งมีทั้งหมด 3 Dropdown โดย Dropdown แรกใช้สำหรับเปลี่ยนไปยังหน้าอื่น ซึ่งสามารถไปที่หน้า Holder, Issuer และ Login ได้ ส่วน Dropdown ที่ 2 แสดงรายการ Transcript ที่ Holder ส่งเข้ามา และ Dropdown สุดท้าย แสดงรายการของผลการตรวจสอบ Transcript ซึ่งในกรณีนี้เป็นกรณีที่ยังไม่มีรายการของ Transcript ที่ส่งเข้ามา และยังไม่มีการตรวจสอบ Transcript



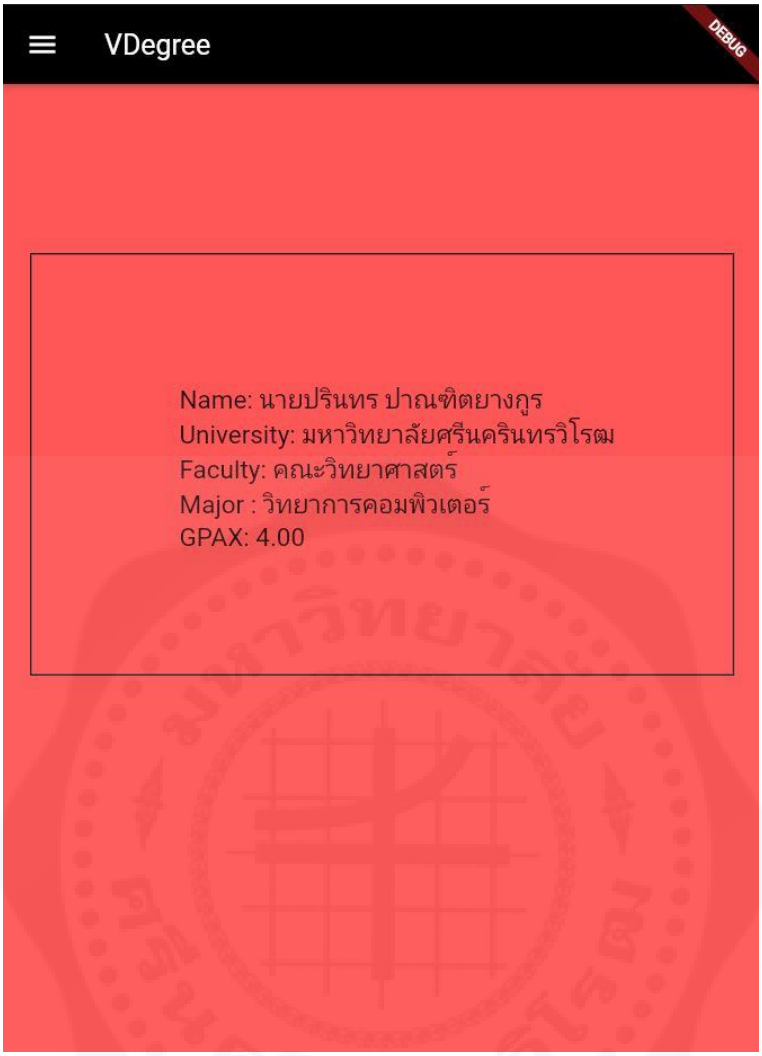
รูปภาพที่ 69 แถบเมนูของ Verifier ในกรณีที่มีข้อมูล

จากรูปภาพที่ 69 แสดงแถบเมนูของ Verifier โดยส่วนบนจะแสดง Role ของผู้ใช้ ส่วนต่อมาเป็น UUID ของผู้ใช้ และในส่วนสุดท้าย คือ Dropdown ซึ่งมีทั้งหมด 3 Dropdown โดย Dropdown แรกใช้สำหรับเปลี่ยนไปยังหน้าอื่น ซึ่งสามารถไปที่หน้า Holder, Issuer และ Login ได้ ส่วน Dropdown ที่ 2 แสดงรายการ Transcript ที่ Holder ส่งเข้ามา และ Dropdown สุดท้าย แสดงรายการของผลการตรวจสอบ Transcript ซึ่งในกรณีนี้เป็นกรณีที่ยังมีรายการของ Transcript ที่ส่งเข้ามา และยังไม่มีการตรวจสอบ Transcript



รูปภาพที่ 70 หน้าแสดงรายละเอียดของผู้ที่ส่ง Transcript เข้ามา

จากรูปภาพที่ 70 แสดงหลังจากกดที่รายการ Transcript ที่ Holder ส่งเข้ามา จะแสดงรายละเอียดต่างๆ ของ Holder ประกอบไปด้วย UUID และชื่อของ Holder โดยปุ่มให้กดตรวจสอบดูรายละเอียดของ Transcript



รูปภาพที่ 71 หน้าแสดงรายละเอียด Transcript ที่ส่งเข้ามา

จากรูปภาพที่ 71 แสดงหลังจากกดตรวจสอบ Transcript จะแสดงข้อมูลภายใน Transcript โดยประกอบไปด้วย ชื่อ มหาวิทยาลัย คณะ สาขา และเกรดเฉลี่ยของ Holder

บทที่ 5
สรุปผล อภิปรายผล และข้อเสนอแนะ

5.1 สรุปผล

จากการทดสอบประสิทธิภาพของระบบบน Amazon Web Services (AWS) พบว่าสามารถเข้ารหัสและถอดรหัสข้อมูลได้จริง โดย Holder สามารถเข้ารหัสข้อมูล Timestamp เพื่อเป็นหนึ่งในข้อมูลที่ใช้ในการยืนยันตัวตนสำหรับใช้ในการร้องขอ Digital Transcript จาก Issuer ได้ และสามารถถอดรหัสข้อมูล Digital Transcript ที่ถูกส่งมาจาก Issuer ได้ ในส่วนของ Issuer นั้นสามารถถอดรหัสข้อมูลคำร้องขอ Digital Transcript ที่ส่งมาจาก Holder ได้ และสามารถเข้ารหัส Digital Transcript ที่จะส่งให้ Holder ได้ และในส่วนสุดท้าย คือ ส่วนของ Verifier สามารถถอดรหัส Digital Transcript เพื่อยืนยันว่าข้อมูลมาจาก Issuer จริง อีกทั้งการพัฒนาระบบบน AWS ยังช่วยเพิ่มประสิทธิภาพของระบบได้อีกด้วย

จากการทดสอบร้องขอและเขียนข้อมูลเข้าไปในระบบ โดยแบ่งเป็น 5 ระดับ คือ 10 Transactions, 100 Transactions, 1000 Transactions, 5000 Transactions และ 10000 Transactions พบว่าระบบสามารถรองรับ Transactions ได้ทุกระดับ โดยใช้เวลาประมวลผลเฉลี่ยประมาณ 3 Transactions/วินาที

จากการทดสอบการทำงานของระบบสำหรับให้ผู้ใช้เข้ามาติดต่อใช้งานผ่านแอปพลิเคชัน พบว่าผู้ใช้สามารถเข้ามาติดต่อใช้งานระบบได้จริง โดยผู้ใช้ที่เป็น Holder สามารถร้องขอ Digital Transcript จาก Issuer และสามารถส่ง Digital Transcript ให้กับ Verifier ได้จริง ผู้ใช้ที่เป็น Issuer สามารถสร้างและส่ง Digital Transcript ให้กับ Holder ได้จริง และผู้ใช้ที่เป็น Verifier สามารถนำ Digital Transcript ที่ได้จาก Holder มาตรวจสอบความถูกต้องของข้อมูลได้จริง

5.2 อภิปรายผล

จากการพัฒนาระบบ พบว่าเทคโนโลยี Blockchain สามารถช่วยเพิ่มความน่าเชื่อถือ ความปลอดภัย และรักษาความถูกต้องของข้อมูลได้จริงตามที่คาดการณ์ไว้ เนื่องจากการที่ Holder, Issuer และ Verifier จะร้องขอข้อมูลหรืออ่านข้อมูลที่ส่งมาจากผู้อื่น จะใช้ Public Key และ Private Key ในการยืนยันตัวตนของผู้ส่งสารและรับสาร ทำให้สามารถที่จะเข้ารหัสข้อมูลหรือถอดรหัสข้อมูลจากผู้อื่นที่ส่งมา

5.3 ข้อเสนอแนะ

จากการพัฒนาระบบ พบว่าระบบสามารถรองรับการร้องขอข้อมูลได้มากถึง 10000 Transactions แต่เมื่อส่งคำร้องขอไปเป็นจำนวนมากจะส่งผลให้ระบบทำงานได้ช้าลง จึงควรเพิ่มประสิทธิของระบบบน AWS ซึ่งค่าใช้จ่ายจะเพิ่มตามไปด้วย สำหรับการเข้าใช้งานระบบ จะเข้าใช้งานผ่านแอปพลิเคชัน ซึ่งแอปพลิเคชันที่สร้างไว้เป็นเพียงแบบจำลอง ควรนำไปพัฒนาเพิ่มเติมและนำไปใช้งานจริง อีกทั้งการนำไปใช้จริงระบบนี้ควรต้องมีการทำ Know Your Customer (KYC) ก่อนเพราะการเข้าใช้ Blockchain ก็ยังมีความเสี่ยงไม่ควรให้ใครก็ได้สามารถเข้าถึงได้ ทั้งนี้การทำ KYC ขึ้นอยู่กับผู้ดูแลไม่ว่าจะเป็นการยืนยันจากบัตรประชาชน หรือนำข้อมูลต่างๆเข้าไปยืนยันที่สำนักงาน

บรรณานุกรม

[1] F. 46 C. Matthew Grant, “Easy to get fake degrees creating real problems – FOX 46 Charlotte,” Sep. 22, 2019. https://www.fox46.com/news/easy-to-get-fake-degrees-creating-real-problems/ (accessed Sep. 25, 2020).

[2] ไทยรัฐ, “มี‘10ด็อกเตอร์เก๊’ สอนในมหาลัยดัง ร้องดีเอสไอ-พันม.สันติภาพฯ,” *ไทยรัฐออนไลน์*, Jan. 17, 2017. https://www.thairath.co.th/content/837526 (accessed Aug. 24, 2020).

[3] netWAY, “เปรียบเทียบ SSL,” *ssl.in.th*. https://ssl.in.th/ssl-comparison/?filter=Digital+Signing (accessed Aug. 25, 2020).

[4] Srinakharinwirot University, “จำนวนผู้สำเร็จการศึกษา ประจำปีการศึกษา 2561,” Oct. 2019. Accessed: Aug. 24, 2020. [Online]. Available: http://edservices.op.swu.ac.th/Portals/26/Documents/%E0%B8%AA%E0%B8%96%E0%B8%B4%E0%B8%95%E0%B8%B4/%E0%B8%88%E0%B8%B3%E0%B8%99%E0%B8%A7%E0%B8%99%E0%B8%9C%E0%B8%B9%E0%B9%89%E0%B8%AA%E0%B8%B3%E0%B9%80%E0%B8%A3%E0%B9%87%E0%B8%88%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%A8%E0%B8%B6%E0%B8%81%E0%B8%A9%E0%B8%B2/%E0%B8%88%E0%B8%B3%E0%B8%99%E0%B8%A7%E0%B8%99%E0%B8%9C%E0%B8%B9%E0%B9%89%E0%B8%AA%E0%B8%B3%E0%B9%80%E0%B8%A3%E0%B9%87%E0%B8%88%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%A8%E0%B8%B6%E0%B8%81%E0%B8%A9%E0%B8%B2%202561.pdf?ver=2019-10-30-133616-267.

[5] Tanya Sattaya-aphitan, “ทบทวนความรู้เรื่อง Blockchain,” *Medium*, Aug. 28, 2018. https://medium.com/@drtan/ทบทวนความรู้เรื่อง-blockchain-3691dded54a3 (accessed Aug. 23, 2020).

[6] VEEDVIL, “ทำความเข้าใจเทคโนโลยี Blockchain จุดเปลี่ยนของหลายอุตสาหกรรม ,” *VEEDVIL*, Mar. 08, 2017. http://www.veedvil.com/news/blockchain/ (accessed Aug. 24, 2020).

[7] Nattaphon, “Digital Signature คืออะไร,” *BCIRCLE*, Sep. 30, 2017. https://www.bcircle.co.th/2017/09/30/digital-signature/ (accessed Aug. 23, 2020).

[8] E. Recommendation, I. C. T. Standard, and E. Transactions, “ข้อเสนอแนะมาตรฐานด้านเทคโนโลยีสารสนเทศและการสื่อสารที่จำเป็นต่อธุรกรรมทางอิเล็กทรอนิกส์ ETDA Recommendation on ICT Standard for Electronic Transactions.”

[9] Techopedia, “What is Key Management? ,” *Techopedia*, Feb. 04, 2014. https://www.techopedia.com/definition/10285/key-management (accessed Aug. 23, 2020).

[10] Townsend Security, “The Definitive Guide to Encryption Key Management Fundamentals,” *Townsend Security*. https://info.townsendsecurity.com/definitive-guide-to-encryption-key-management-fundamentals (accessed Aug. 23, 2020).

[11] Mindphp, “Certificate Authority คืออะไร ,” *Mindphp*, Jul. 12, 2016. https://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/3633-certificate-authority-%E0%B9%80%E0%B8%8B%E0%B8%AD%E0%B8%97%E0%B8%B4%E0%B8%9F%E0%B8%9F%E0%B8%B0%E0%B9%80%E0%B8%84%E0%B8%97-%E0%B8%AD%E0%B8%AD%E0%B8%98%E0%B8%AD%E0%B8%A3%E0%B8%B5%E0%B8%97%E0%B8%B5-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html (accessed Aug. 23, 2020).

[12] S. Pongnumkul, C. Siripanpornchana, and S. Thajchayapong, “Performance analysis of private blockchain platforms in varying workloads,” *2017 26th International Conference on Computer Communications and Networks, ICCCN 2017*, 2017, doi: 10.1109/ICCCN.2017.8038517.

[13] Y. Hu, Y. Xiong, W. Huang, and X. Bao, “KeyChain: Blockchain-based key distribution,” in *Proceedings - 2018 4th International Conference on Big Data Computing and Communications, BIGCOM 2018*, Oct. 2018, pp. 126–131, doi: 10.1109/BIGCOM.2018.00027.

[14] Z. A. Lux, F. Beierle, S. Zickau, and S. Gondor, “Full-text Search for Verifiable Credential Metadata on Distributed Ledgers,” *2019 6th International Conference on Internet of Things: Systems, Management and Security, IOTSMS 2019*, pp. 519–528, 2019, doi: 10.1109/IOTSMS48152.2019.8939249.

[15] D. Age, “ตรวจปริญญาปลอมโดยใช้ Blockchain,” 2018, [Online]. Available: http://www.digitalagemag.com/learning/vocab/blockchain/blockchain-check-fake-degree.