



WILL CHAIN
WILL ON BLOCKCHAIN

MR. THITIPONG BOONTHANAKORN
MR. NARONGYOT SOONTCHARARAK
MR. SUBTAWEE NGANRUNGRUANG

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)
FACULTY OF ENGINEERING
KING MONGKUT'S UNIVERSITY OF TECHNOLOGY THONBURI
2022

Will Chain
Will On Blockchain

MR. THITIPONG BOONTHANAKORN
MR. NARONGYOT SOONTHARARAK
MR. SUBTAWEE NGANRUNGRUANG

A Project Submitted in Partial Fulfillment
of the Requirements for
the Degree of Bachelor of Engineering (Computer Engineering)
Faculty of Engineering
King Mongkut's University of Technology Thonburi
2022

Project Committee

..... (Asst.Prof. Marong Phadoongsidhi, Ph.D.)	Project Advisor
..... (Mrs. Piyanit Wepulanon , Ph.D.)	Committee Member
..... (Asst.Prof. Thumrongrat Amornraksa , Ph.D.)	Committee Member
..... (Asst.Prof. Surapont Toomnark)	Committee Member

Project Title	Will Chain Will On Blockchain
Credits	3
Member(s)	MR. THITIPONG BOONTHANAKORN MR. NARONGYOT SOONTHARARAK MR. SUBTAWEE NGANRUNGRUANG
Project Advisor	Asst.Prof. Marong Phadoongsidhi, Ph.D.
Program	Bachelor of Engineering
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	2022

Abstract

Will Chain is a platform developed with the aim of studying how Blockchain networks work and managing wills in real-world assets. and digital assets. Will Chain will include features for managing and keeping current wills. that can meet both real-life assets and digital assets There will be a feature that will support the addition of a will, namely a feature for delivering assets to heirs. If the conditions are the same as in the will Makes wills more secure because the system will not go through the hands of an intermediary but will only have that system. It is also convenient to make wills. and will have even greater coverage.

Keywords: Multihop ad hoc networks / Topology control / Single-Hop Throughput

หัวข้อปริญญานิพนธ์	Will Chain Will on Blockchain
หน่วยกิต	3
ผู้เขียน	นายฐิติพงศ์ บุนธนากร นายณรงค์ยศ สุนทรารักษ์ นายทรัพย์ทวี งานรุ่งเรือง
อาจารย์ที่ปรึกษา	ผศ.ดร.มารอง ผดุงสิทธิ์
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	2565

บทคัดย่อ

Will Chain เป็นแพลตฟอร์มที่ถูกพัฒนาขึ้นมาโดยมีวัตถุประสงค์เพื่อศึกษาการทำงานของเครือข่าย Blockchain และจัดการเกี่ยวกับพันธกรรมในด้านของสินทรัพย์ในโลกความเป็นจริง และสินทรัพย์ดิจิทัล โดยที่ Will Chain นั้นจะมีฟีเจอร์ในการจัดการและเก็บรักษาพันธกรรมที่มีอยู่ในปัจจุบัน ที่จะสามารถตอบโจทย์ได้ทั้งสินทรัพย์ในชีวิตจริงและสินทรัพย์ดิจิทัล โดยจะมีฟีเจอร์ที่จะรองรับการทำพันธกรรมเพิ่มเติมคือฟีเจอร์สำหรับการส่งมอบสินทรัพย์ให้กับทายาท ถ้ามีเงื่อนไขตรงกับในพันธกรรม ทำให้การทำพันธกรรมนั้นมีความปลอดภัยมากขึ้นเนื่องจากตัวระบบจะไม่ผ่านมือคนกลางแต่จะมีแค่ระบบนั้น อีกทั้งสะดวกในการทำพันธกรรม และจะมีความครอบคลุมที่มากยิ่งขึ้น

คำสำคัญ: การซุบเคลือบด้วยไฟฟ้า / การซุบเคลือบผิวเหล็ก / เคลือบผิวรังสี

กิตติกรรมประกาศ

การทำโครงการครั้งนี้สำเร็จลงได้ด้วยความช่วยเหลือของ ผู้ช่วยศาสตราจารย์ ดร. มารอง ผดุงสิทธิ์ ที่ปรึกษาโครงการ ซึ่งได้ให้ความกรุณา สละเวลาให้คำปรึกษา คำแนะนำ ข้อเสนอแนะอันมีประโยชน์อย่างมาก และความช่วยเหลือตลอดการทำโครงการนี้จนสำเร็จลุล่วงได้ด้วยดี ผู้จัดทำโครงการจึงขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ ผ.ศ.สุรพันธ์ ตุ่มนาค , ดร.ปิยนิษฐ์ เวปุลานนท์ และ รศ.ดร.ธำรงรัตน์ อมรรักษา ที่ได้สละเวลา ร่วมเป็นคณะกรรมการตรวจสอบโครงการในครั้งนี้

ท้ายที่สุดนี้ โครงการนี้อาจจะไม่สำเร็จเลยหากไม่มีเพื่อนในภาควิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรีที่ให้ ความช่วยเหลือ การสนับสนุน รวมทั้งคอยเป็นกำลังใจสำคัญเสมอมา

ทีมผู้จัดทำหวังว่าโครงการนี้จะก่อให้เกิดประโยชน์ต่อการทำปริญญานิพนธ์ในปัจจุบัน และสามารถครอบคลุมไปถึงปริญญานิพนธ์ของรุ่นพี่ปริญญานิพนธ์ที่ยังไม่มีเทคโนโลยีรองรับในตอนนี และเกิดการเปลี่ยนแปลงที่ดีขึ้นในอนาคต

สารบัญ

หน้า

ABSTRACT	ii
บทคัดย่อ	iii
กิตติกรรมประกาศ	iv
สารบัญ	v
สารบัญตาราง	vii
สารบัญรูปภาพ	viii
สารบัญสัญลักษณ์	ix
สารบัญคำศัพท์ทางเทคนิคและคำย่อ	x
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ	1
1.5 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ	1
1.6 การแยกย่อยงาน และวางแผนคำแนะนำจากอาจารย์ที่ปรึกษา	1
1.7 ตารางการดำเนินงาน	3
1.8 ผลการดำเนินงานในภาคการศึกษาที่ 1	3
1.9 ผลการดำเนินงานในภาคการศึกษาที่ 2	3
บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง	4
2.1 ทฤษฎีที่เกี่ยวข้อง	4
2.1.1 Blockchain	4
2.1.2 ERC-20	4
2.1.3 Ethereum Chain and ETH	4
2.1.4 Software Engineering	4
2.1.5 Smart Contract	5
2.1.6 Non-Fungible Token (NFT)	5
2.1.7 Fractional- Non-Fungible Token (F-NFT)	5
2.1.8 Asset (สินทรัพย์)	6
2.1.9 Digital Asset (สินทรัพย์ดิจิทัล)	6
2.1.10 พิธีกรรม	6
2.2 งานวิจัยที่เกี่ยวข้อง	7
2.2.1 CryptoWill	7
2.3 เทคนิคและเทคโนโลยีที่ใช้	7
2.3.1 Ethereum	7
2.3.2 Ethereum Chain	7
2.3.3 GitHub	7
2.3.4 MetaMask	8
2.3.5 NestJS	8
2.3.6 Next.js	8
2.3.7 Solidity	8
2.3.8 TypeScript	8
2.3.9 Web3.js	8

2.3.10 Truffle	8
2.4 อัลกอริทึมในการประมวลผลข้อความ	9
2.4.1 อัลกอริทึม I	9
2.4.2 อัลกอริทึม I	9
2.5 เครื่องมือที่ใช้ในการพัฒนา	9
บทที่ 3 วิธีการดำเนินงาน	10
3.1 ข้อกำหนดและความต้องการของระบบ	10
3.2 สถาปัตยกรรมระบบ	10
3.3 Hardware Module 1	10
3.3.1 Component 1	10
3.3.2 Logical Circuit Diagram	10
3.4 Hardware Module 2	10
3.4.1 Component 1	10
3.4.2 Component 2	10
3.5 Path Finding Algorithm	10
3.6 Database Design	10
3.7 UML Design	10
3.8 GUI Design	10
3.9 การออกแบบการทดลอง	10
3.9.1 ตัวชี้วัดและปัจจัยที่ศึกษา	10
3.9.2 รูปแบบการเก็บข้อมูล	10
บทที่ 4 ผลการดำเนินงาน	11
4.1 ประสิทธิภาพการทำงานของระบบ	11
4.2 ความพึงพอใจการใช้งาน	11
4.3 การวิเคราะห์ข้อมูลและผลการทดลอง	11
บทที่ 5 บทสรุป	12
5.1 สรุปผลโครงงาน	12
5.2 ปัญหาที่พบและการแก้ไข	12
5.3 ข้อจำกัดและข้อเสนอแนะ	12
หนังสืออ้างอิง	13
APPENDIX	14
A ชื่อภาคผนวกที่ 1	15
B ชื่อภาคผนวกที่ 2	17

สารบัญตาราง

ตารางที่	หน้า
2.1 test table method1	9
3.1 test table x1	10

สารบัญรูป

รูปที่	หน้า
2.1 ความแตกต่างระหว่าง Waterfall Method กับ Agile Method	5
2.2 ภาพแสดงหลักการทำงานของโปรเจค CryptoWill	7
2.3 The network model	9

สารบัญสัญลักษณ์

SYMBOL

α	Test variable
λ	Interarival rate
μ	Service rate

UNIT

m^2
jobs/ second
jobs/ second

สารบัญคำศัพท์ทางเทคนิคและคำย่อ

Test	=	test
MANET	=	Mobile Ad Hoc Network

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ

ในปัจจุบันนั้นเทคโนโลยีเข้ามามีบทบาทในการใช้ชีวิตของผู้คนเป็นอย่างมาก ไม่ว่าจะเป็นในด้านของ การเงิน สินทรัพย์ เป็นต้น แต่ว่าจะมีในด้านของพินัยกรรมที่นับว่าเป็นเอกสารที่ไม่มีการใช้เทคโนโลยีเข้ามาช่วยเหลือในปัจจุบัน โดยยังที่จะต้องทำการเก็บรักษาไว้ที่ตัวเองหรือไม่ก็เก็บไว้ที่ทนายของตนเองทำให้บางครั้งพินัยกรรมนั้น ๆ อาจเกิดการเสียหายหรือสูญหายได้ หรือแม้กระทั่งอาจเกิดโอกาสเปลี่ยนแปลงจากบุคคลที่สามได้ ทำให้การทำพินัยกรรมในแต่ละครั้งมีความยุ่งยากและไม่ปลอดภัยสำหรับผู้ที่จะทำพินัยกรรม รวมถึงพินัยกรรมในส่วนนี้ยังครอบคลุมในด้านการสืบทอดสินทรัพย์ดิจิทัล อย่างเช่น Cryptocurrency ได้ เนื่องจากยังไม่มีเทคโนโลยีที่รองรับในปัจจุบัน

จึงเกิดแนวคิดที่จะสร้าง แพลตฟอร์มสำหรับการจัดการพินัยกรรมทั้งสินทรัพย์ในโลกความเป็นจริง และสินทรัพย์ดิจิทัลผ่านระบบ Blockchain ที่สามารถนำพินัยกรรมที่มีอยู่ในปัจจุบันนั้นมาขึ้นระบบ Blockchain เพื่อเก็บรักษาพินัยกรรมนั้น และสามารถทำการสืบทอดสินทรัพย์ไปยังผู้รับพินัยกรรมได้ รวมไปถึงสินทรัพย์ดิจิทัลอีกด้วย โดยคำนึงถึงความปลอดภัยและความสะดวกสบายของผู้ใช้งาน

1.2 วัตถุประสงค์

- เพื่อศึกษาเทคโนโลยี Blockchain
- เพื่อสร้างแพลตฟอร์มสำหรับการจัดการพินัยกรรมทั้งสินทรัพย์ในโลกความเป็นจริง และสินทรัพย์ดิจิทัล
- เพื่อให้พินัยกรรมในปัจจุบันสามารถครอบคลุมถึงสินทรัพย์ดิจิทัล
- เพื่อเก็บรักษาพินัยกรรมให้มีความปลอดภัยมากขึ้น
- เพื่ออำนวยความสะดวกในการเก็บพินัยกรรม

1.3 ขอบเขตของโครงการ

- พัฒนาแพลตฟอร์มสำหรับการจัดการพินัยกรรมทั้งสินทรัพย์ในโลกความเป็นจริง และสินทรัพย์ดิจิทัล
- ใช้ภาษา Solidity ในการพัฒนา Smart Contract

1.4 ประโยชน์ที่คาดว่าจะได้รับ

Will Chain เป็นการนำเทคโนโลยี Blockchain เพื่อการทำพินัยกรรมโดยจะสามารถถ่ายทอดมรดกที่เป็นสินทรัพย์ที่ระบบรองรับจากผู้ทำการเขียนพินัยกรรม ไปหาผู้รับสินทรัพย์ได้ด้วยรูปแบบของ NFT หรือ F-NFT

1.5 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ

โครงการนี้พัฒนาขึ้นมาจากการใช้ความรู้ในด้าน Blockchain Technology (Ethereum chain โดยใช้เครื่องมือพัฒนา Smart Contract ด้วยภาษา Solidity ในการพัฒนา) และใช้ความรู้เรื่อง NFT , F-NFT เพื่อใช้ในการเก็บข้อมูลพินัยกรรมของตัวโปรเจกของเรา รวมถึงการทำ Decentralize Application ที่ใช้ Next Typescript Framework ในการพัฒนาส่วนติดต่อกับผู้ใช้งานไปถึงความรู้ด้าน วิศวกรรมซอฟต์แวร์ และ ด้านพินัยกรรม เพื่อที่จะสามารถทำการถ่ายทอดพินัยกรรมได้ภายใน Decentralize Application

1.6 การแยกย่อยงาน และวางแผนคำแนะนำจากอาจารย์ที่ปรึกษา

1. ศึกษาค้นคว้าที่มาและความสำคัญของปัญหา
2. เสนอหัวข้อโครงการให้กับอาจารย์ที่ปรึกษา
3. ทำการสำรวจหรือศึกษาค้นคว้าข้อมูลที่เกี่ยวข้องกับโครงการ
 - ศึกษาเรื่องพินัยกรรม

- ศึกษาเรื่องกฎหมาย
 - ศึกษาเรื่องสินทรัพย์
4. นำเสนอโครงการและข้อมูลที่ศึกษาค้นคว้าให้กับอาจารย์ที่ปรึกษา
 5. จัดทำข้อเสนอโครงการ
 6. นำเสนอข้อเสนอโครงการ
 7. จัดทำรายงาน
 - รายงานบทที่ 1 จากข้อมูลข้อเสนอโครงการ
 - รายงานบทที่ 2 จากข้อมูลการศึกษาค้นคว้าเกี่ยวกับทฤษฎีที่เกี่ยวข้อง
 - รายงานบทที่ 3 รายงานการออกแบบการทำงานของระบบเบื้องต้น
 8. วิเคราะห์และออกแบบระบบ
 - ออกแบบการทำงาน Algorithms ของ Smart Contract ที่ใช้งานในระบบ
 - ออกแบบรูปแบบป็นยกรรมที่จะใช้ในระบบ
 - ออกแบบส่วนของผู้ใช้งาน (UX/UI)
 9. ศึกษาและพัฒนา Blockchain และ Smart Contract
 - ศึกษาการทำงานของ Blockchain ด้วย Ethereum chain
 - ศึกษาและพัฒนาส่วนของ Smart Contract ที่ใช้ในการควบคุมระบบด้วยภาษา Solidity
 - ศึกษาและพัฒนา NFT ในระบบ
 10. ศึกษาและพัฒนา Web application
 - ศึกษาและพัฒนาส่วนของผู้ใช้งานด้วย Next.js Typescript และ User Interface Framework อื่น ๆ
 - ศึกษาเกี่ยวกับ API ของหน่วยงานรัฐ
 11. นำเสนอโครงงาน 3 บท
 12. ศึกษาและพัฒนา Blockchain และ Smart Contract (ต่อจากภาคการศึกษาที่ 1)
 13. ทดสอบการทำงานของ Ethereum chain
 14. ปรับปรุงและแก้ไข Ethereum chain
 15. ศึกษาและพัฒนา Web application (ต่อจากภาคการศึกษาที่ 1)
 16. ทดสอบการทำงานของ Web application
 17. ปรับปรุงและแก้ไข Web application
 18. จัดทำรายงานโครงงานฉบับสมบูรณ์
 19. นำเสนอโครงงาน

1.7 ตารางการดำเนินงาน

1.8 ผลการดำเนินงานในภาคการศึกษาที่ 1

1. รูปเล่มรายงานโครงงาน 3 บท
2. ออกแบบการทำงานของ Smart contract
 - แบบจำลองโครงสร้างของ Smart Contract
 - แบบจำลองการทำงานของ Smart Contract
3. ออกแบบโครงสร้างของ Application
 - แผนผังภาพรวมของระบบ
 - แผนผังการทำงานของ Application
 - แบบจำลองส่วนติดต่อผู้ใช้งาน

1.9 ผลการดำเนินงานในภาคการศึกษาที่ 2

1. พัฒนา Blockchain
2. พัฒนา Web application (Will Chain)
3. เชื่อมต่อส่วนผู้ใช้งาน และ Smart Contract
4. ผลการทดสอบการใช้งาน
5. ทดสอบการใช้งาน
6. รายงานโครงงานฉบับสมบูรณ์

บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 Blockchain

Blockchain คือเทคโนโลยีการประมวลผลและจัดเก็บข้อมูลแบบกระจายศูนย์ หรือที่เรียกว่า Distributed Ledger Technology (DLT) ซึ่งเป็นรูปแบบการบันทึกข้อมูลที่ใช้หลักการ Cryptography ร่วมกับกลไก Consensus โดยข้อมูลที่ถูกบันทึกในระบบ Blockchain นั้นจะสามารถทำการแก้ไขเปลี่ยนแปลงได้ยาก ช่วยเพิ่มความถูกต้อง และความน่าเชื่อถือของข้อมูล โดยไม่ต้องอาศัยคนกลาง

Blockchain สามารถแบ่งออกได้เป็น 3 ประเภท โดยพิจารณาจากข้อกำหนดในการ เข้าร่วมเป็นสมาชิกของเครือข่ายคือ Blockchain แบบเปิดสาธารณะ (Public Blockchain) Blockchain แบบปิด (Private Blockchain) และ Blockchain แบบเฉพาะกลุ่ม (Consortium Blockchain)

2.1.1.1 Public Blockchain คือ Blockchain วงเปิดที่อนุญาตให้ทุกคนสามารถเข้าใช้งานไม่ว่า จะเป็นการอ่าน หรือการทำธุรกรรมต่าง ๆ ได้อย่างอิสระโดยไม่จำเป็นต้องขออนุญาต หรือรู้จักกันในอีกชื่อ คือ Permissionless Blockchain

2.1.1.2 Private Blockchain คือ Blockchain วงปิดที่เข้าใช้งานได้เฉพาะผู้ที่ได้รับ อนุญาตนั้นซึ่งส่วนใหญ่ถูกสร้างขึ้นเพื่อใช้งานภายในองค์กร ดังนั้นข้อมูลการทำธุรกรรมต่าง ๆ จะถูกจำกัดอยู่เฉพาะภายในเครือข่าย

2.1.1.3 Consortium Blockchain คือ Blockchain ที่ เปิดให้ใช้งานได้เฉพาะกลุ่ม เท่านั้น โดยเป็นการผสมผสานแนวคิดระหว่าง Public Blockchain และ Private Blockchain ซึ่งส่วนมากเป็นการรวมตัวกันขององค์กรที่มีลักษณะธุรกิจ เหมือนกัน และต้องมีการแลกเปลี่ยนข้อมูลระหว่างกันอย่างสม่ำเสมอแล้วมารวมตัวกันตั้ง Blockchain ขึ้นมา ทั้งนี้เนื่องจาก ธุรกรรมและข้อมูลที่จัดเก็บ เป็นข้อมูลที่ เป็นความลับหรือข้อมูลส่วนตัวภายในองค์กร ส่งผลให้ไม่สามารถเปิดเผยข้อมูลดังกล่าวทั้งหมดแก่สาธารณะชนได้ ดังนั้นผู้เข้าร่วม Blockchain เฉพาะกลุ่ม จำเป็นต้องได้รับ การอนุญาตจากตัวแทนเสียก่อน จึงจะสามารถเข้าใช้งานได้ ยกตัวอย่าง เช่น เครือข่ายระหว่างธนาคาร ที่ใช้ในการ แลกเปลี่ยนข้อมูลการทำธุรกรรม หรือแลกเปลี่ยนสินทรัพย์ภายในกลุ่ม

2.1.2 ERC-20

ERC-20 เป็น Protocol มาตรฐานสำหรับการสร้างโทเคนบน Ethereum blockchain โดยมีชื่อเต็มคือ Ethereum Request for Comments ซึ่งมาตรฐาน ERC-20 ถูกนำมาใช้ตั้งแต่ปี 2015 และในปัจจุบันมีโทเคนจำนวนมากที่รองรับ ERC-20

2.1.3 Ethereum Chain and ETH

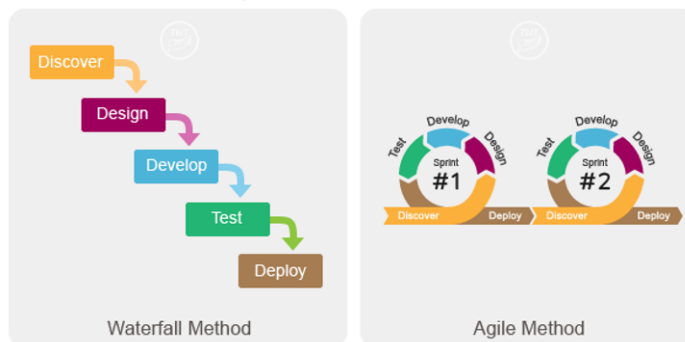
Ethereum คือแพลตฟอร์มบน Blockchain Network ที่ทำงานด้วย Smart Contract มีลักษณะแพลตฟอร์มเป็นรูปแบบ Decentralized Platform แบบ Open Source ทำให้นักพัฒนาสามารถเข้ามาพัฒนา แก้ไข หรือดัดแปลงโค้ดได้ทุกคน พร้อมทั้งกำหนดเงื่อนไขต่าง ๆ สำหรับนำไปใช้งานบน Blockchain โดยมี Smart Contract ดำเนินการและระบบจะทำงานตามเงื่อนไขโปรแกรมที่กำหนดมา ทำให้ผู้ใช้งาน Blockchain ของ Ethereum ทำธุรกรรมได้ โดยไม่ต้องผ่านตัวกลางอื่น นอกจากนี้ การประยุกต์ใช้ Smart Contract และศักยภาพประมวลผลโดยรวมของแพลตฟอร์มที่สูงกว่า Bitcoin และเหรียญ Ether หรือเหรียญ ETH คือ สกุลเงินดิจิทัลอย่างหนึ่ง ที่ถูกพัฒนาขึ้นมาบน Blockchain Ethereum มีส่วนช่วยขับเคลื่อนการทำงานในระบบนิเวศของ Ethereum

2.1.4 Software Engineering

2.1.4.1 Software Development Methodology

- Agile Software Development เป็นกระบวนการที่ช่วยลดการทำงานที่เป็นขั้นตอนและงานด้านการทำเอกสาร แต่จะไม่มุ่งเน้นในเรื่องการสื่อสารของทีมมากขึ้น เพื่อให้เกิดการพัฒนาสินค้าและบริการใหม่ ๆ ได้รวดเร็วขึ้น แล้วจึงนำสิ่งที่ได้ไปให้ผู้ใช้งานกลุ่มตัวอย่าง (Target group) ทดสอบใช้งานจริง จากนั้นจึงรวบรวมผลทดสอบมาประเมินดูอีกครั้ง เพื่อใช้เป็นแนวทางในการแก้ไขปรับปรุงสินค้าและบริการนั้นๆ ให้ดีขึ้นทีละนิด ด้วยแนวทางนี้จะทำให้องค์กรสามารถพัฒนาสินค้าและบริการได้อย่างรวดเร็วและตอบโจทย์ผู้ใช้งานได้มากขึ้นอย่างสม่ำเสมอ

Development Methodologies



รูปที่ 2.1 ความแตกต่างระหว่าง Waterfall Method กับ Agile Method

- Scrum (สกรัม) คือการนำแนวคิดในการทำงานแบบ Agile (อไจล์) มาปฏิบัติตามขั้นตอนของสกรัม เพื่อระบุปัญหาที่มีความซับซ้อน เปลี่ยนแปลงบ่อย เพื่อให้สามารถส่งมอบผลิตภัณฑ์ที่ตอบสนองต่อการเปลี่ยนแปลงที่เกิดขึ้นได้อย่างรวดเร็ว ช่วยให้การพัฒนาผลิตภัณฑ์แบบ Agile มีขั้นตอนการดำเนินงานและผลลัพธ์ที่ชัดเจน โปร่งใส สามารถตรวจสอบประสิทธิภาพของแต่ละขั้นตอนการดำเนินงาน สามารถปรับปรุงและวัดผลการปรับปรุงที่เกิดขึ้นได้

- Kanban ที่มาเริ่มต้นมาจากระบบการทำงานของ Toyota ซึ่งประสบความสำเร็จอย่างมากจนทำให้สามารถผลิตรถออกมาได้ไวกว่าคู่แข่งทั่วโลกจนครองตลาดไปได้มาก สำหรับวงการ Software ได้ถูก David J. Anderson จับนำมาปรับปรุงให้เข้ากับ Software Development เพื่อการพัฒนา Software ได้อย่างรวดเร็วที่สุดด้วยเช่นกัน และสุดท้ายถูกนำไปเป็นส่วนหนึ่งของ Lean Software Development รวมไปถึงถูกจัดให้เป็น Agile อีกแบบหนึ่งนอกเหนือไปจาก Scrum อีกด้วย Kanban มีกฎอยู่แค่ 3 ข้อ

- Visualize the workflow – แสดง flow การทำงานของระบบให้ออกมาให้เห็นภาพอย่างชัดเจน สามารถบอกได้ว่าขณะนี้งานไปติดขัดที่จุดไหน อย่างไรให้ชัดเจน
- Limit Work In Progress (WIP) – จุดหลักของ Kanban เลยคือการ limit งานต่อหนึ่งหน่วยย่อย เช่นงานสำหรับ Development ห้ามถือเกิน 2 งานเพื่อป้องกันไม่ให้งาน Overload มากเกินไป และจะทำให้สูญเสียเวลาไปมากกว่าที่ควรจะเป็น
- Measure the lead time – วัดผลการทำงานและปรับปรุงให้ดียิ่งขึ้นไปอีก ตรงนี้จะเรียกว่า Cycle time หรือค่าเฉลี่ยที่ Card 1 อันจะอยู่บนบอร์ดตั้งแต่เริ่มต้นไปจนถึงขึ้นบน production จริง

2.1.5 Smart Contract

Smart Contract หมายถึง กระบวนการทางดิจิทัล ที่กำหนดขั้นตอนการทำธุรกรรมโดยอัตโนมัติไว้ล่วงหน้า โดยไม่ต้องอาศัยตัวกลาง อย่างเช่น ธนาคาร ซึ่งการสร้าง Smart Contract ที่เป็นระบบอัตโนมัติอย่างเต็มรูปแบบ โดยคู่สัญญาทั้งสองฝ่ายจะมีการตกลงกันก่อนหน้านี้ ถึงขั้นตอน กลไก ในการทำการธุรกรรมดังกล่าว ซึ่งการพัฒนาที่ส่งผลกระทบต่อรูปแบบธุรกิจแบบดั้งเดิมของธนาคาร

2.1.6 Non-Fungible Token (NFT)

NFT ย่อมาจาก Non-Fungible Token เป็นชื่อเรียกของ Cryptocurrency ประเภทหนึ่ง เป็นสินทรัพย์ดิจิทัลที่มีเพียงชิ้นเดียวในโลก ไม่สามารถทำซ้ำหรือคัดลอกได้ ต่อให้มีการก๊อปปี้ไป แต่ต้นฉบับของจริงจะมีอยู่เพียงหนึ่งเดียวเท่านั้น ส่วนโทเคน NFT ก็เป็นเหมือนโฉนด เพื่อแสดงความเป็นเจ้าของสินทรัพย์ชิ้นนั้น

2.1.7 Fractional- Non-Fungible Token (F-NFT)

F-NFT จะเปรียบเหมือนธุรกิจประเภทห้างหุ้นส่วนของกิจการมีแห่งเดียวในโลก มีผู้ร่วมลงทุนอย่างน้อย 2 คนขึ้นไป ทุกคนเป็นเจ้าของกิจการ มีอำนาจในการตัดสินใจร่วมกันในทุก ๆ เรื่อง ดังนั้น F-NFT จึงเป็นเพียงสัดส่วนของ NFT ทั้งหมดที่ถูกแบ่งออกเป็นส่วนย่อย ทำให้ผู้คนจำนวนมากสามารถอ้างสิทธิ์ความเป็นเจ้าของชิ้นส่วนของ NFT เดียวกันได้

2.1.8 Asset (สินทรัพย์)

Asset หมายถึง ทรัพยากรที่มีและอยู่ในการควบคุมของกิจการ สินทรัพย์นี้อาจจะเป็นสิ่งที่มีตัวตนหรือไม่มีตัวตนก็ได้ ซึ่งสามารถตีราคามูลค่าเป็นเงินได้ ทรัพยากรดังกล่าวเป็นผล ของ เหตุการณ์ในอดีต ซึ่งกิจการคาดว่าจะได้รับประโยชน์เชิงเศรษฐกิจจากทรัพยากรนั้นในอนาคต

2.1.9 Digital Asset (สินทรัพย์ดิจิทัล)

คือ "สิ่งที่มีมูลค่าและเราสามารถเป็นเจ้าของได้ แต่ไม่สามารถแตะต้องได้ทางกายภาพ" สิ่งเหล่านั้นถูกสร้างขึ้นในระบบดิจิทัล และเก็บไว้ในอุปกรณ์อิเล็กทรอนิกส์อย่าง คอมพิวเตอร์ ฮาร์ดแวร์ แล็ปท็อป หรือ อุปกรณ์เก็บข้อมูลต่าง ๆ เป็นต้น

2.1.10 พินัยกรรม

พินัยกรรม หมายถึง การแสดงเจตนากำหนดการเผื่อตายซึ่งให้มีผลบังคับได้เมื่อถึงแก่ความตาย หรือถ้าเป็นภาษาพูดก็ได้แก่คำสั่งเสียของผู้ตาย โดยในการทำพินัยกรรม กฎหมายกำหนดรูปแบบไว้ 5 แบบด้วยกัน ดังนี้

2.1.10.1 พินัยกรรมแบบธรรมดา ผู้ทำต้องทำเป็นหนังสือ คือการพิมพ์ข้อความพินัยกรรมลงในกระดาษ มากน้อยหรือจำนวนก็แผ่นก็ต้องแล้วแต่เนื้อหาหรือจำนวนทรัพย์สิน ลงวัน เดือน ปี ที่ทำให้ชัดเจน และผู้ทำต้องลงลายมือชื่อไว้ต่อหน้าพยานอย่างน้อย 2 คน และพยานต้องลงลายมือชื่อรับรองการทำพินัยกรรมในขณะทำด้วย

2.1.10.2 พินัยกรรมแบบเขียนเองทั้งฉบับ ผู้ทำพินัยกรรมจะทำเป็นเอกสารเขียนเองทั้งฉบับก็ได้ แต่ผู้ทำนั้นต้องเขียนพินัยกรรมนั้นด้วยลายมือตนเอง ลงวัน เดือน ปี ที่ทำ และที่สำคัญต้องลงลายมือชื่อผู้ทำด้วย กรณีนี้จะมีพยานมารับรู้การทำพินัยกรรมด้วยหรือไม่ก็ได้

2.1.10.3 พินัยกรรมแบบเอกสารฝ่ายเมือง เป็นแบบพินัยกรรมที่ต้องอาศัยกระบวนการโดยเฉพาะที่มีเจ้าหน้าที่รัฐเข้ามาเกี่ยวข้อง ผู้ทำพินัยกรรมต้องไปแจ้งความประสงค์โดยให้ถ้อยคำข้อความของตนแก่เจ้าพนักงานที่เขตหรืออำเภอพร้อมพยานอย่างน้อย 2 คน เจ้าพนักงานจะอ่านข้อความให้ผู้ทำพินัยกรรมและพยานฟัง เมื่อเห็นว่าถูกต้องครบถ้วนแล้ว ผู้ทำพินัยกรรมพร้อมพยานทั้งสองต้องลงลายมือชื่อไว้ ต่อจากนั้น เจ้าพนักงานจะลงลายมือชื่อ วัน เดือน ปี ที่ทำ พร้อมประทับตราตำแหน่ง

2.1.10.4 พินัยกรรมแบบเอกสารลับ ผู้ทำพินัยกรรมทำพินัยกรรมแล้วปิดผนึก และนำไปที่ทำการอำเภอหรือเขต ผู้ทำพินัยกรรมต้องลงลายมือชื่อและพยานอีกอย่างน้อย 2 คน และให้ถ้อยคำต่อบุคคลเหล่านั้นว่าเป็นพินัยกรรมของตน เจ้าหน้าที่จะบันทึกถ้อยคำลง วัน เดือน ปี ที่ทำพินัยกรรมแสดงไว้บนซองและประทับตราตำแหน่งไว้เป็นสำคัญโดยผู้ทำพินัยกรรม พยานและเจ้าหน้าที่ต้องลงลายมือชื่อไว้หน้าซองตรงที่ปิดผนึก

2.1.10.5 พินัยกรรมแบบทำด้วยวาจา กรณีมีพฤติการณ์พิเศษที่บุคคลไม่สามารถทำพินัยกรรมแบบอื่นที่กล่าวมาข้างต้น เช่น การตกอยู่ในภยันตรายใกล้ความตาย หรืออยู่ในระหว่างสงคราม หรือเกิดมีโรคร้าย เราสามารถทำพินัยกรรมแบบทำด้วยวาจาก็ได้ โดยผู้ทำพินัยกรรมต้องแสดงเจตนาทำพินัยกรรมต่อหน้าพยานอย่างน้อย 2 คนพร้อมกัน พยานต้องรับฟังข้อความนั้นแล้วไปแจ้งต่อทางราชการโดยเร็วที่สุด ทั้งยังต้องแจ้งวัน เดือน ปี สถานที่ทำพินัยกรรมและพฤติการณ์พิเศษนั้นด้วย เจ้าพนักงานต้องจดข้อความที่พยานแจ้งไว้ และพยาน 2 คนนั้นต้องลงลายมือชื่อไว้

ข้อจำกัดและข้อควรระวังในการทำพินัยกรรม

1. พินัยกรรมเป็นนิติกรรมที่ต้องทำตามแบบที่กำหนดเท่านั้น
2. ต้องเขียน วัน เดือน ปี ลงลายมือชื่อทั้งผู้ทำพินัยกรรมและผู้ที่เป็นพยาน
3. ผู้ที่เป็นพยานจะต้องไม่เป็นคู่เยาว์หรือผู้หย่อนความสามารถ และต้องไม่เป็นผู้มีส่วนได้เสียในกองมรดกนั้นด้วย
4. ผู้ทำพินัยกรรมต้องมีอายุ 15 ปีบริบูรณ์ขึ้นไป
5. พินัยกรรมควรจะต้องผู้จัดการมรดกโดยสามารถระบุผู้ทำหน้าที่ผู้จัดการมรดกที่เจ้ามรดกไว้ใจลงในพินัยกรรมไปได้
6. สิทธิ หน้าที่ และความรับผิดชอบ ก็สามารถกำหนดในพินัยกรรมได้

เลย

.7 ทรัพย์สินที่ระบุในพินัยกรรมต้องเป็นทรัพย์สินหรือสิทธิของผู้ทำพินัยกรรมเท่านั้น ทั้งต้องแยกสินส่วนตัวออกจากสินสมรสด้วย

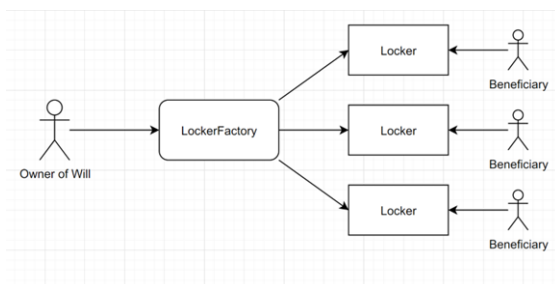
.8 เงินประกันชีวิต เงินบำเหน็จตกทอด เงินมีบ้านอายุตกทอด เงินฌาปนกิจสงเคราะห์ตกทอด ไม่อาจเป็นมรดกที่ระบุลงในพินัยกรรมได้ เพราะไม่ใช่ทรัพย์สินที่เจ้ามรดกมีอยู่ก่อนตาย

2.2 งานวิจัยที่เกี่ยวข้อง

ในส่วนนี้เป็นการสรุปเนื้อหาโดยรวมของงานวิจัยที่เกี่ยวข้องกับการพัฒนาโครงการงาน will on blockchain ที่มีการใช้งานในส่วนของ digital asset คือ CryptoWill

2.2.1 CryptoWill

โปรเจกต์ได้อธิบายวิธีการทำระบบ CryptoWill ด้วยการให้ user ทำการเลือกเหรียญที่ต้องการทำ Smart Contract ที่ต้องการส่งให้ทายาทและหลังจากนั้นตัวระบบจะทำการให้กำหนดเวลาของการ contract นี้จะส่งต่อเมื่อไหร่ อย่างเช่น ถ้าตั้ง 2 ปี ผู้ใช้งานจะต้องมาก่อนเวลาที่จะเกิด contract นี้ โดยรูปแบบของการทำจะมีวิธีการดำเนินการดังรูป



รูปที่ 2.2 ภาพแสดงหลักการทำงานของโปรเจก CryptoWill

โดยเจ้าของพินัยกรรมจะทำพินัยกรรมและจะเก็บสินทรัพย์ไว้ใน blockchain และหลังจากนั้นจะส่งต่อให้ทายาทไปเมื่อถึงเวลาของพินัยกรรม

2.3 เทคนิคและเทคโนโลยีที่ใช้

2.3.1 Ethereum

Ethereum (ETH) คือเหรียญหนึ่งในเหรียญดิจิทัลที่ใช้เทคโนโลยีบล็อกเชนทำงานอยู่เบื้องหลัง โดยเหรียญนี้มีจุดเด่นในการทำ

2.3.2 Ethereum Chain

Ethereum คือแพลตฟอร์มบน Blockchain Network ที่ทำงานด้วย Smart Contract มีลักษณะแพลตฟอร์มเป็นรูปแบบ Decentralized Platform แบบ Open Source ทำให้นักพัฒนาสามารถเข้ามาพัฒนา แก้ไข หรือดัดแปลงโค้ดได้ทุกคน พร้อมทั้งกำหนดเงื่อนไขต่าง ๆ สำหรับนำไปใช้งานบน Blockchain โดยมี Smart Contract ดำเนินการและระบบจะทำงานตามเงื่อนไขโปรแกรมที่กำหนดมา ทำให้ผู้ใช้งาน Blockchain ของ Ethereum ทำธุรกรรมได้ โดยไม่ต้องผ่านตัวกลางอื่น

2.3.3 GitHub

Git คือ Version Control ที่ถูกพัฒนาขึ้นมาเพื่อใช้ในกระบวนการพัฒนาซอฟต์แวร์ คือ ระบบที่ถูกพัฒนาขึ้นมาเพื่อใช้สำหรับการติดตาม ตรวจสอบ การพัฒนา แก้ไข Source Code ไฟล์ต่าง ๆ ในขั้นตอนการพัฒนา ที่สามารถตรวจสอบ ได้ทุกตัวอักษร ทุกบรรทัด ทุกไฟล์ที่มีการแก้ไข และยังมีคุณลักษณะที่สนับสนุนการทำงานแบบ Agile อีกด้วย จึงทำให้เราสามารถทำงานได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

2.3.4 MetaMask

MetaMask หรือ MetaMask Wallet กระเป๋าเงินสินทรัพย์ดิจิทัล เป็น Wallet สำหรับเก็บ Cryptocurrency บนระบบนิเวศของ Ethereum ทุกชนิด ในกลุ่ม ERC-20 ซึ่ง Metamask พัฒนาโดยบริษัท ConsenSys โดยมีผู้ก่อตั้งคือ Joseph Lubin เมื่อปี 2016 (Joseph Lubin ยังเป็นผู้ร่วมก่อตั้ง Ethereum และ เคยยังเคยเป็น Speaker ในงาน Techsauce Global Summit)

2.3.5 NestJS

NestJS เป็น Framework สำหรับ Build Node.js ในฝั่ง Server-side Applications โดยสนับสนุนการทำงานแบบ

- TypeScript เต็มรูปแบบ
- OOP (Object Oriented Programming)
- FP (Functional Programming)
- FRP (Functional Reactive Programming)

2.3.6 Next.js

Next.js คือ JavaScript webapps framework ถูกสร้างขึ้น on top จาก library ดัง ๆ อย่าง React, Webpack, และ Babel และมีจุดเด่นที่ server-side rendering ที่สามารถ render หน้าเว็บบน server แทนที่จะ render บน browser ได้ จึงทำให้ข้อมูลที่ส่งให้ฝั่ง client นั้นถูก render เสร็จเรียบร้อยแล้ว ทำให้ฝั่ง client สามารถนำไปแสดงผลได้ทันที

2.3.7 Solidity

Solidity คือภาษาสำหรับการสร้าง Smart Contract เป็นภาษาที่ได้รับอิทธิพลมาจาก C ++, Python และ JavaScript ที่สำคัญเลยก็คือเป็นภาษาชนิดที่ statically typed และเป็นภาษาแบบ Object Oriented (OO) เพราะว่ามีคุณสมบัติของการสืบทอดและการทำ struct เป็นต้น

2.3.8 TypeScript

TypeScript เป็นภาษาโปรแกรมที่รวมความสามารถที่ ES2015 เองมีอยู่ สิ่งที่เพิ่มขึ้นมาคือสนับสนุน Type System รวมถึงคุณสมบัติอื่นๆที่เพิ่มมากขึ้น เช่น Enum และความสามารถที่เพิ่มขึ้นของการโปรแกรมเชิงวัตถุ TypeScript นั้นเป็น transpiler เหมือน Babel นั้นหมายความว่าตัวแปลภาษาของ TypeScript จะแปลโค้ดที่เราเขียนให้เป็น JavaScript อีกทีหนึ่ง จึงมั่นใจได้ว่าผลลัพธ์สุดท้ายจะสามารถใช้งานได้บนเว็บเบราว์เซอร์ทั่วไป

2.3.9 Web3.js

Web3.js เป็น JavaScript API ที่ทำให้ส่วนติดต่อผู้ใช้งานสามารถติดต่อและเรียกใช้ฟังก์ชันจากฝั่งของ Ethereum ได้ โดย Web3.js สามารถส่ง API ไปติดต่อกับฝั่ง Smart Contract ให้สร้าง Transaction สำหรับเรียกใช้ Methods หรือ Get ค่าตัวแปรต่าง ๆ บน Smart Contract ที่อยู่บน Ethereum Blockchain ได้

2.3.10 Truffle

Truffle Suite เป็นเครื่องมือ Open source สำหรับพัฒนา Decentralized Application บน Blockchain ที่รองรับ Ethereum virtual machine นอกจากนี้ Truffle Suite ยังเป็น Development Environment, Testing Environment และ Asset Pipeline ที่มุ่งเน้น ให้การทำงานของนักพัฒนาซอฟต์แวร์ง่ายขึ้น และการใช้ Truffle Suite จะมี

- Build-in smart contract compiler
- Automated contract testing with Mocha and Chai
- Pipeline ที่สามารถปรับแต่งได้

- Scriptable deployment and migrations framework
- Network management for deploying to many public and private networks
- Interactive console for direct contract communication
- Instant rebuilding of assets during development
- External script runner that executes scripts within a Truffle environment

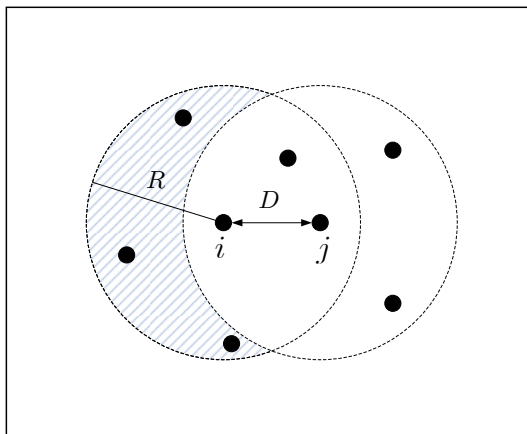
ตารางที่ 2.1 test table method1

Center	Center	left aligned	Right	Right aligned
Center	Center	left aligned	Right	Right aligned
Center	Center	left aligned	Right	Right aligned
Center	Center	left aligned	Right	Right aligned
Center	Center	left aligned	Right	Right aligned

2.4 อัลกอริทึมในการประมวลผลข้อความ

2.4.1 อัลกอริทึม I

You can place the figure and refer to it as รูปที่ 2.3. The figure and table numbering will be run and updated automatically when you add/remove tables/figures from the document.



รูปที่ 2.3 The network model

2.4.2 อัลกอริทึม I

Add more subsections as you want.

2.5 เครื่องมือที่ใช้ในการพัฒนา

บทที่ 3 วิธีการดำเนินงาน

Explain the design (how you plan to implement your work) of your project. Adjust the section titles below to suit the types of your work. Detailed physical design like circuits and source codes should be placed in the appendix.

3.1 ข้อกำหนดและความต้องการของระบบ

3.2 สถาปัตยกรรมระบบ

ตารางที่ 3.1 test table x1

SYMBOL		UNIT
α	Test variable	m ²
λ	Interarrival rate	jobs/ second
μ	Service rate	jobs/ second

3.3 Hardware Module 1

3.3.1 Component 1

3.3.2 Logical Circuit Diagram

3.4 Hardware Module 2

3.4.1 Component 1

3.4.2 Component 2

3.5 Path Finding Algorithm

3.6 Database Design

3.7 UML Design

3.8 GUI Design

3.9 การออกแบบการทดลอง

3.9.1 ตัวชี้วัดและปัจจัยที่ศึกษา

3.9.2 รูปแบบการเก็บข้อมูล

บทที่ 4 ผลการดำเนินงาน

You can title this chapter as **Preliminary Results** ผลการดำเนินงานเบื้องต้น or **Work Progress** ความก้าวหน้าโครงการ for the progress reports. Present implementation or experimental results here and discuss them. ใส่เฉพาะหัวข้อที่เกี่ยวข้องกับงานที่ทำ

- 4.1 ประสิทธิภาพการทำงานของระบบ
- 4.2 ความพึงพอใจการใช้งาน
- 4.3 การวิเคราะห์ข้อมูลและผลการทดลอง

บทที่ 5 บทสรุป

This chapter is optional for proposal and progress reports but is required for the final report.

5.1 สรุปผลโครงการ

สรุปว่าโครงการบรรลุตามวัตถุประสงค์ที่ตั้งไว้หรือไม่ อย่างไร

5.2 ปัญหาที่พบและการแก้ไข

State your problems and how you fixed them.

5.3 ข้อจำกัดและข้อเสนอแนะ

ข้อจำกัดของโครงการ What could be done in the future to make your projects better.

หนังสืออ้างอิง

1. I. Norros, 1995, "On the use of Fractional Brownian Motion in the Theory of Connectionless Networks," **IEEE J. Select. Areas Commun.**, vol. 13, no. 6, pp. 953--962, Aug. 1995.
2. H.S. Kim and N.B. Shroff, 2001, "Loss Probability Calculations and Asymptotic Analysis for Finite Buffer Multiplexers," **IEEE/ACM Trans. Networking**, vol. 9, no. 6, pp. 755--768, Dec. 2001.
3. D.Y. Eun and N.B. Shroff, 2001, "A Measurement-Analytic Framework for QoS Estimation Based on the Dominant Time Scale," in **Proc. IEEE INFOCOM'01**, Anchorage, AK, Apr. 2001.

ภาคผนวก A

ชื่อภาคผนวกที่ 1

ใส่หัวข้อตามความเหมาะสม

This is where you put hardware circuit diagrams, detailed experimental data in tables or source codes, etc..

This appendix describes two static allocation methods for fGn (or fBm) traffic. Here, λ and C are respectively the traffic arrival rate and the service rate per dimensionless time step. Their unit are converted to a physical time unit by multiplying the step size Δ . For a fBm self-similar traffic source, Norros [1] provides its EB as

$$C = \lambda + (\kappa(H)\sqrt{-2\ln \epsilon})^{1/H} a^{1/(2H)} x^{-(1-H)/H} \lambda^{1/(2H)} \quad (\text{A.1})$$

where $\kappa(H) = H^H(1-H)^{(1-H)}$. Simplicity in the calculation is the attractive feature of (A.1).

The MVA technique developed in [2] so far provides the most accurate estimation of the loss probability compared to previous bandwidth allocation techniques according to simulation results. Consider a discrete-time queueing system with constant service rate C and input process λ_n with $\mathbb{E}\{\lambda_n\} = \lambda$ and $\text{Var}\{\lambda_n\} = \sigma^2$. Define $X_n \equiv \sum_{k=1}^n \lambda_k - Cn$. The loss probability due to the MVA approach is given by

$$\epsilon \approx \alpha e^{-m_x/2} \quad (\text{A.2})$$

where

$$m_x = \min_{n \geq 0} \frac{((C - \lambda)n + B)^2}{\text{Var}\{X_n\}} = \frac{((C - \lambda)n^* + B)^2}{\text{Var}\{X_{n^*}\}} \quad (\text{A.3})$$

and

$$\alpha = \frac{1}{\lambda\sqrt{2\pi\sigma^2}} \exp\left(\frac{(C - \lambda)^2}{2\sigma^2}\right) \int_C^\infty (r - C) \exp\left(\frac{(r - \lambda)^2}{2\sigma^2}\right) dr \quad (\text{A.4})$$

For a given ϵ , we numerically solve for C that satisfies (A.2). Any search algorithm can be used to do the task. Here, the bisection method is used.

Next, we show how $\text{Var}\{X_n\}$ can be determined. Let $C_\lambda(l)$ be the autocovariance function of λ_n . The MVA technique basically approximates the input process λ_n with a Gaussian process, which allows $\text{Var}\{X_n\}$ to be represented by the autocovariance function. In particular, the variance of X_n can be expressed in terms of $C_\lambda(l)$ as

$$\text{Var}\{X_n\} = nC_\lambda(0) + 2 \sum_{l=1}^{n-1} (n-l)C_\lambda(l) \quad (\text{A.5})$$

Therefore, $C_\lambda(l)$ must be known in the MVA technique, either by assuming specific traffic models or by off-line analysis in case of traces. In most practical situations, $C_\lambda(l)$ will not be known in advance, and an on-line measurement algorithm developed in [3] is required to jointly determine both n^* and m_x . For fGn traffic, $\text{Var}\{X_n\}$ is equal to $\sigma^2 n^{2H}$, where $\sigma^2 = \text{Var}\{\lambda_n\}$, and we can find the n^* that minimizes (A.3) directly. Although λ can be easily measured, it is not the case for σ^2 and H . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.

ภาคผนวก B

ชื่อภาคผนวกที่ 2

ใส่หัวข้อตามความเหมาะสม

Next, we show how $\text{Var}\{X_n\}$ can be determined. Let $C_\lambda(l)$ be the autocovariance function of λ_n . The MVA technique basically approximates the input process λ_n with a Gaussian process, which allows $\text{Var}\{X_n\}$ to be represented by the autocovariance function. In particular, the variance of X_n can be expressed in terms of $C_\lambda(l)$ as

$$\text{Var}\{X_n\} = nC_\lambda(0) + 2 \sum_{l=1}^{n-1} (n-l)C_\lambda(l) \quad (\text{B.1})$$

Add more topic as you need

Therefore, $C_\lambda(l)$ must be known in the MVA technique, either by assuming specific traffic models or by off-line analysis in case of traces. In most practical situations, $C_\lambda(l)$ will not be known in advance, and an on-line measurement algorithm developed in [3] is required to jointly determine both n^* and m_x . For fGn traffic, $\text{Var}\{X_n\}$ is equal to $\sigma^2 n^{2H}$, where $\sigma^2 = \text{Var}\{\lambda_n\}$, and we can find the n^* that minimizes (A.3) directly. Although λ can be easily measured, it is not the case for σ^2 and H . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.