



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

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

Copyright reserved

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 beneficiary. If the conditions are the same as in the will Makes wills more secure It is also convenient to make wills. and will have even greater coverage.

Keywords: Asset / Blockchain / Cryptocurrency / Digital Asset / Non-Fungible Token (NFT) / Smart Contract / Will

หัวข้อปริญญานิพนธ์	Will Chain Will on Blockchain
หน่วยกิต	3
ผู้เขียน	นายฐิติพงศ์ บุณธนากร 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 ทำให้การทำพันธกรรมนั้นมีความปลอดภัยมากขึ้น อีกทั้งสะดวกในการทำพันธกรรม และจะมีความครอบคลุมที่มากยิ่งขึ้น

คำสำคัญ: Asset / Blockchain / Cryptocurrency / Digital Asset / Non-Fungible Token (NFT) / Smart Contract / Will

กิตติกรรมประกาศ

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

ขอขอบพระคุณวราณัฐ สุทธิการณ ซึ่งได้ให้ความกรุณาสละเวลาให้คำแนะนำการออกแบบ Smart Contract และความช่วยเหลือตลอดการทำโครงการนี้

ขอขอบพระคุณ ผ.ศ.สุรพนธ์ ตุ่มนาค , ดร.ปิยนิตย์ เวฬุานนท์ และ รศ.ดร.ธำรงรัตน์ อมรรักษา ที่ได้สละเวลา ร่วมเป็นคณะกรรมการตรวจสอบโครงการในครั้งนี้

ท้ายที่สุดนี้ โครงการนี้อาจจะไม่สำเร็จเลยหากไม่มีเพื่อนในภาควิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรีที่ให้ ความช่วยเหลือ การสนับสนุน รวมทั้งคอยเป็นกำลังใจสำคัญเสมอมา

ทีมผู้จัดทำหวังว่าโครงการนี้จะก่อให้เกิดประโยชน์ต่อการทำปริญญานิพนธ์ในปัจจุบัน และสามารถครอบคลุมไปถึงปริญญานิพนธ์ของสินทรัพย์ดิจิทัลที่ยังไม่มีเทคโนโลยีรองรับในตอนนี และเกิดการเปลี่ยนแปลงที่ดีขึ้นในอนาคต

สารบัญ

หน้า

ABSTRACT	ii
บทคัดย่อ	iii
กิตติกรรมประกาศ	iv
สารบัญ	v
สารบัญตาราง	vii
สารบัญรูปภาพ	viii
สารบัญคำศัพท์ทางเทคนิคและคำย่อ	ix
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	2
1.7 ตารางการดำเนินงาน	3
1.8 ผลการดำเนินงานในภาคการศึกษาที่ 1	4
1.9 ผลการดำเนินงานในภาคการศึกษาที่ 2	4
บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง	5
2.1 ทฤษฎีที่เกี่ยวข้อง	5
2.1.1 Blockchain [1]	5
2.1.2 ERC-20 [2]	5
2.1.3 ERC-721 [3]	5
2.1.4 Assets Tokenization [4]	5
2.1.5 Ethereum Chain and ETH[5]	5
2.1.6 Soulbound Token (SBTs)[?]	6
2.1.7 Software Engineering [6, 7]	6
2.1.8 Smart Contract [8]	6
2.1.9 Non-Fungible Token (NFT) [9]	7
2.1.10 Asset (สินทรัพย์) [10]	7
2.1.11 Digital Asset (สินทรัพย์ดิจิทัล) [11]	7
2.1.12 พิณยกรรม [12]	7
2.2 งานวิจัยที่เกี่ยวข้อง	8
2.2.1 CryptoWill [13]	8
2.3 เทคนิคและเทคโนโลยีที่ใช้	8
2.3.1 Ethereum Chain [5]	8
2.3.2 GitHub [14]	8
2.3.3 MetaMask [15]	9
2.3.4 NestJS [16]	9
2.3.5 Next.js [17]	9
2.3.6 Solidity [18]	9
2.3.7 TypeScript [19]	9
2.3.8 Web3.js [20]	9

2.3.9 Hardhat [21]	10
2.3.10 Pinata [22]	10
2.3.11 IPFS [23]	10
บทที่ 3 การออกแบบและวิธีการดำเนินงาน	11
3.1 ระบบการทำงาน	11
3.1.1 ภาพรวมของระบบ	11
3.1.2 User Journey	11
3.2 Cryptocurrency Wallet	12
3.3 Diagram Unified Modelling Language (UML)	12
3.3.1 แผนภาพ Use Case Diagram	12
3.3.2 Use Case Narrative	13
3.3.3 Smart Contract	18
3.3.4 System Architecture Diagram	19
3.3.5 Sequence Diagram	20
3.4 ส่วนติดต่อผู้ใช้งาน (User Interface)	31
3.4.1 หน้าแรก	31
3.4.2 หน้าโปรไฟล์	31
3.4.3 หน้าโปรไฟล์ยืนยันการลงทะเบียน	31
3.4.4 หน้าโปรไฟล์ลงทะเบียนสำเร็จ	32
3.4.5 หน้ายินยอมของคุณ	32
3.4.6 หน้ายกเลิกการทำยินยอมของคุณ	33
3.4.7 หน้า Approve ยินยอมของคุณ	34
3.4.8 หน้าบันทึกยินยอม	34
3.4.9 หน้าจัดการสินทรัพย์ภายในยินยอมของคุณ	34
3.4.10 หน้าเพิ่มสินทรัพย์จริง	35
3.4.11 หน้ายินยอมที่ได้รับ	35
3.4.12 หน้ายืนยันรับยินยอม	36
3.4.13 หน้ารับยินยอมเสร็จสิ้น	36
3.5 ออกแบบการทดสอบ	37
บทที่ 4 ผลการดำเนินงาน	38
4.1 Site map	38
4.2 Token ที่ใช้ใน Will-Chain	38
4.3 Test Plan	38
4.4 Software Testing	40
4.4.1 Test Case	40
4.4.2 Test Smart Contract	40
4.5 การ Deploy Smart Contract ไปที่ Ethereum Chain	41
บทที่ 5 สรุปผล	44
5.1 สรุปปัญหาที่พบและวิธีการแก้ไข	44
5.1.1 Design System	44
5.1.2 Process Management	44
5.1.3 Blockchain	44
5.1.4 Smart Contract	44
5.1.5 Frontend	44
หนังสืออ้างอิง	45

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางการดำเนินงาน ประจําภาคการศึกษาที่ 1/2565	3
1.2 ตารางการดำเนินงาน ประจําภาคการศึกษาที่ 2/2565	3
3.1 ตารางแสดงรายละเอียดของ Use Case Connect MetaMask Wallet	13
3.2 ตารางแสดงรายละเอียดของ Use Case Create Will	13
3.3 ตารางแสดงรายละเอียดของ Use Case View Will	13
3.4 ตารางแสดงรายละเอียดของ Use Case Upload Will	14
3.5 ตารางแสดงรายละเอียดของ Use Case Delete Will	14
3.6 ตารางแสดงรายละเอียดของ Use Case View Beneficiary	14
3.7 ตารางแสดงรายละเอียดของ Use Case Set Beneficiary	15
3.8 ตารางแสดงรายละเอียดของ Use Case View Asset	15
3.9 ตารางแสดงรายละเอียดของ Use Case Add Asset	15
3.10 ตารางแสดงรายละเอียดของ Use Case Check User Status	16
3.11 ตารางแสดงรายละเอียดของ Use Case Active Will	16
3.12 ตารางแสดงรายละเอียดของ Use Case Claim Will	17
3.13 ตารางแสดงรายละเอียดของ Use Case Display Claim Asset	17
3.14 ตารางแสดงรายละเอียดของ Connect MetaMask Sequence Diagram	20
3.15 ตารางแสดงรายละเอียดของ Create Will Sequence Diagram	21
3.16 ตารางแสดงรายละเอียดของ View Will Sequence Diagram	22
3.17 ตารางแสดงรายละเอียดของ Upload Pdf Will Sequence Diagram	22
3.18 ตารางแสดงรายละเอียดของ Delete Will Sequence Diagram	23
3.19 ตารางแสดงรายละเอียดของ View Beneficiary Sequence Diagram	25
3.20 ตารางแสดงรายละเอียดของ Set Beneficiary Sequence Diagram	25
3.21 ตารางแสดงรายละเอียดของ View Assets Sequence Diagram	26
3.22 ตารางแสดงรายละเอียดของ Deposit Digital Assets Sequence Diagram	27
3.23 ตารางแสดงรายละเอียดของ Deposit Real Assets Sequence Diagram	27
3.24 ตารางแสดงรายละเอียดของ Active Will Sequence Diagram	28
3.25 ตารางแสดงรายละเอียดของ Claim Assets Sequence Diagram	29

สารบัญรูป

รูปที่	หน้า
2.1 ความแตกต่างระหว่าง Waterfall Method กับ Agile Method	6
2.2 ภาพแสดงหลักการทำงานของโปรเจค CryptoWill	8
3.1 ภาพรวมแสดงการทำงานของระบบ	11
3.2 แสดง User Journey	11
3.3 แสดงการทำงานของระบบทั้งหมด Use Case Diagram	12
3.4 แสดงการ interaction ของ Smart Contract ของระบบ Will Chain	18
3.5 แสดงภาพออกแบบสถาปัตยกรรมของระบบ Will Chain	19
3.6 แสดง Connect MetaMask Sequence Diagram	20
3.7 แสดง Create Will Sequence Diagram	21
3.8 แสดง View Will Sequence Diagram	22
3.9 แสดง Upload Will Sequence Diagram	23
3.10 แสดง Delete Will Sequence Diagram	24
3.11 แสดง View Beneficiary Sequence Diagram	25
3.12 แสดง Set Beneficiary Sequence Diagram	26
3.13 แสดง View Assets Sequence Diagram	26
3.14 แสดง Deposit Digital Assets Sequence Diagram	27
3.15 แสดง Deposit Real Assets Sequence Diagram	28
3.16 แสดง Active Will Sequence Diagram	29
3.17 แสดง Claim Assets Sequence Diagram	30
3.18 หน้าแรก	31
3.19 หน้าโปรไฟล์	31
3.20 หน้าโปรไฟล์ ยืนยันการลงทะเบียน	32
3.21 หน้าโปรไฟล์ ยืนยันการลงทะเบียนสำเร็จ	32
3.22 หน้าพินัยกรรมของคุณ	33
3.23 หน้ายกเลิกการทำพินัยกรรมของคุณ	33
3.24 หน้า Approve พินัยกรรมของคุณ	34
3.25 หน้าบันทึกพินัยกรรม	34
3.26 หน้าจัดการสินทรัพย์ภายในพินัยกรรมของคุณ	35
3.27 หน้าเพิ่มสินทรัพย์จริง	35
3.28 พินัยกรรมที่ได้รับ	36
3.29 หน้ายืนยันรับพินัยกรรม	36
3.30 หน้ารับพินัยกรรมเสร็จสิ้น	37
4.1 ผล Test Coverage ของ Smart Contract	41
4.2 แสดงผล code หลัง verified contract	42
4.3 แสดงฟังก์ชันสำหรับที่แสดงผล หลัง verified contract	43
4.4 แสดงฟังก์ชันที่มีการเขียนลงระบบ ethereum chain หลัง verified contract	43

สารบัญคำศัพท์ทางเทคนิคและคำย่อ

Asset	=	ทรัพย์สินที่เราถืออยู่ทั้งหมด เงินที่อยู่ในบัญชีทั้งหมดอยู่ในกระเป๋าทั้งหมดรวมทั้งสินที่เราถืออยู่ทั้งหมด
Blockchain	=	ระบบโครงข่ายในการเก็บบัญชีธุรกรรมออนไลน์
Cryptocurrency	=	สกุลเงินเข้ารหัส เป็นสินทรัพย์ดิจิทัล
Digital Asset	=	สิ่งที่มีมูลค่าและเราสามารถเป็นเจ้าของได้ แต่ไม่สามารถแตะต้องได้ทางกายภาพ
Non-Fungible Token(NFT)	=	สิ่งของที่มีความแตกต่างเฉพาะตัวไม่สามารถทดแทนกันได้หรือซื้อเป็นหน่วยย่อยได้
Smart Contract	=	กระบวนการทางดิจิทัล ที่กำหนดขั้นตอนการทำธุรกรรมโดยอัตโนมัติไว้ล่วงหน้า โดยไม่ต้องอาศัยตัวกลาง
Will	=	พันธกรรมที่เก็บค่าเสียหายสุดท้ายในการทำกิจการต่าง ๆ

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 ที่สามารถนำพินัยกรรมที่มีอยู่ในปัจจุบันนั้นเอาขึ้นระบบ Blockchain เพื่อเก็บรักษาพินัยกรรมนั้น และสามารถทำการสืบทอดสินทรัพย์ไปยังผู้รับพินัยกรรมได้ รวมไปถึงสินทรัพย์ดิจิทัลอีกด้วย โดยคำนึงถึงความปลอดภัยและความสะดวกสบายของผู้ใช้งาน

1.2 วัตถุประสงค์

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

1.3 ขอบเขตของโครงการ

- พัฒนาแพลตฟอร์มพินัยกรรมที่สามารถใช้งานได้บน Ethereum chain (Test-network) เท่านั้น
- พัฒนาแพลตฟอร์มสำหรับการจัดการพินัยกรรมทั้งสินทรัพย์ในโลกความเป็นจริง และสินทรัพย์ดิจิทัล
- ใช้ภาษา Solidity ในการพัฒนา Smart Contract

1.4 ประโยชน์ที่คาดว่าจะได้รับ

Will Chain เป็นการใช้เทคโนโลยี Blockchain เพื่อการทำพินัยกรรมโดยจะสามารถถ่ายทอดมรดกที่เป็นสินทรัพย์ที่ระบบรองรับจากผู้ทำการเขียนพินัยกรรม ไปหาผู้รับสินทรัพย์ได้ด้วยรูปแบบของ NFT

1.5 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ

โครงการนี้พัฒนาขึ้นมาจากการใช้ความรู้ในด้าน Blockchain Technology (Ethereum chain โดยใช้เครื่องมือพัฒนา Smart Contract ด้วยภาษา Solidity ในการพัฒนา) และใช้ความรู้เรื่อง 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. ทำการสำรวจหรือศึกษาค้นคว้าข้อมูลที่เกี่ยวข้องกับโครงงาน
 - ศึกษาเรื่องพินัยกรรม
 - ศึกษาเรื่องกฎหมาย
 - ศึกษาเรื่องสินทรัพย์
4. นำเสนอโครงการและข้อมูลที่ศึกษาค้นคว้าให้กับอาจารย์ที่ปรึกษา
5. จัดทำข้อเสนอโครงการ
6. นำเสนอข้อเสนอโครงการ
7. จัดทำรายงาน
 - รายงานบทที่ 1 จากข้อมูลข้อเสนอโครงงาน
 - รายงานบทที่ 2 จากข้อมูลการศึกษาค้นคว้าเกี่ยวกับทฤษฎีที่เกี่ยวข้อง
 - รายงานบทที่ 3 รายงานการออกแบบการทำงานของระบบเบื้องต้น
8. วิเคราะห์และออกแบบระบบ
 - ออกแบบการทำงาน Algorithms ของ Smart Contract ที่ใช้งานในระบบ
 - ออกแบบรูปแบบพินัยกรรมที่จะใช้ในระบบ
 - 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

ตารางที่ 1.1 ตารางการดำเนินงาน ประจำปีการศึกษาที่ 1/2565

ตารางการดำเนินงาน ประจำปีการศึกษาที่ 1/2565																							
ที่	หัวข้อการดำเนินงาน	ระยะเวลา																					
		สิงหาคม				กันยายน				ตุลาคม				พฤศจิกายน				ธันวาคม					
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
1	ศึกษาค้นคว้าที่มาของและ ความสำคัญของปัญหา																						
2	เสนอหัวข้อโครงการให้กับอาจารย์ ที่ปรึกษา																						
3	ทำการสำรวจหรือศึกษาค้นคว้า ข้อมูลที่เกี่ยวข้องกับโครงการ																						
4	นำเสนอโครงการและข้อมูลที่ศึกษาค้นคว้าให้กับอาจารย์ที่ปรึกษา																						
5	จัดทำข้อเสนอโครงการ																						
6	นำเสนอข้อเสนอโครงการ																						
7	จัดทำรายงาน																						
8	วิเคราะห์และออกแบบระบบ																						
9	ศึกษาและพัฒนา Blockchain และ Smart Contract																						
10	ศึกษาและพัฒนา Web application																						
11	นำเสนอโครงการ 3 บท																						

ตารางที่ 1.2 ตารางการดำเนินงาน ประจำปีการศึกษาที่ 2/2565

ตารางการดำเนินงาน ประจำปีภาคการศึกษาที่ 2/2565																			
ที่	หัวข้อการดำเนินงาน	ระยะเวลา																	
		มกราคม				กุมภาพันธ์				มีนาคม				เมษายน					
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
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																		

หมายเหตุ : ขั้นตอนการพัฒนา ปรับปรุง แก้ไข และทดสอบ Application ใช้การทำงานแบบ Agile Methodology

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 [1]

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 ที่เปิดให้ใช้งานได้เฉพาะกลุ่ม เท่านั้น โดยเป็นการผสมผสานแนวคิดระหว่าง Public Blockchain และ Private Blockchain ซึ่งส่วนมากเป็นการรวมตัวกันขององค์กรที่มีลักษณะธุรกิจ เหมือนกัน และต้องมีการแลกเปลี่ยนข้อมูลระหว่างกันอย่างสม่ำเสมอแล้วมารวมตัวกันตั้ง Blockchain ขึ้นมา ทั้งนี้เนื่องจาก ธุรกรรมและข้อมูลที่จัดเก็บ เป็นข้อมูลที่ เป็นความลับหรือข้อมูลส่วนตัวภายในองค์กร ส่งผลให้ไม่สามารถเปิดเผยข้อมูลดังกล่าวทั้งหมดแก่สาธารณชนได้ ดังนั้นผู้เข้าร่วม Blockchain เฉพาะกลุ่ม จำเป็นต้องได้รับ การอนุญาตจากตัวแทนเสียก่อน จึงจะสามารถเข้าใช้งานได้ ยกตัวอย่าง เช่น เครือข่ายระหว่างธนาคาร ที่ใช้ในการ แลกเปลี่ยนข้อมูลการทำธุรกรรม หรือแลกเปลี่ยนสินทรัพย์ภายในกลุ่ม

2.1.2 ERC-20 [2]

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

2.1.3 ERC-721 [3]

ERC-721 คือมาตรฐานที่ทำให้ข้อมูลที่เป็นดิจิทัลหรือโทเคนนั้นมีความเฉพาะตัว (Non-Fungible) โดยส่วนมากมักจะถูกนำไปใช้กับของสะสมต่าง ๆ ที่อยู่ในรูปแบบดิจิทัล ที่ต้องการให้มีความหายากและไม่เหมือนใคร ไม่สามารถทำซ้ำได้ เพราะว่ามันมีโค้ดที่สามารถระบุได้ว่าใครเป็นเจ้าของอย่างชัดเจน ในทางกลับกัน ERC-20 นั้นเป็นมาตรฐานที่จะทำให้ทุก ๆ โทเคนที่ถูกสร้างขึ้นมามีมาตรฐานดังกล่าวมีความเหมือนกัน

2.1.4 Assets Tokenization [4]

Asset tokenization คือการเปลี่ยนสินทรัพย์ในโลกจริงให้เป็นสินทรัพย์ดิจิทัล หรือโทเคน โดยใช้เทคโนโลยีบล็อกเชน ทำให้เราสามารถแปลงสินทรัพย์อย่าง ทอง อสังหาริมทรัพย์ หรืองานศิลปะให้เป็นโทเคนดิจิทัลได้

2.1.5 Ethereum Chain and ETH[5]

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 ที่ถูกพัฒนาขึ้นมานบน Blockchain Ethereum มีส่วนช่วยขับเคลื่อนการทำงานในระบบนิเวศของ Ethereum

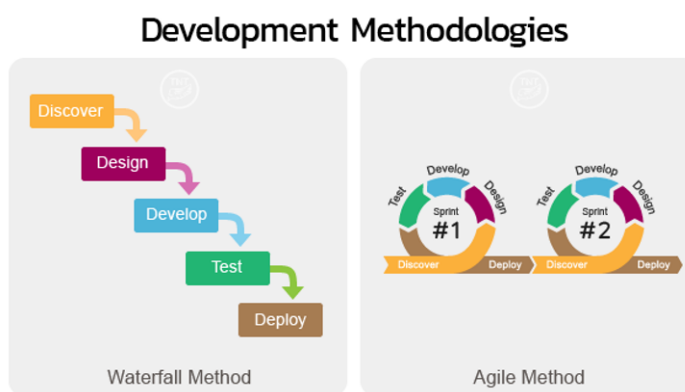
2.1.6 Soulbound Token (SBTs)[?]

คือ token การระบุตัวตนดิจิทัลที่แสดงถึงคุณลักษณะ และความสำเร็จที่ประกอบขึ้นเป็นบุคคลหรือนิติบุคคลภายในระบบ blockchain ได้โดย soulbound token จะไม่สามารถโอนได้ไปหาผู้อื่นได้

2.1.7 Software Engineering [6, 7]

2.1.7.1 Software Development Methodology

• Agile Software Development เป็นกระบวนการที่ช่วยลดการทำงานที่เป็นขั้นตอนและงานด้านการทำเอกสารลง แต่จะไปมุ่งเน้นในเรื่องการสื่อสารของทีมมากขึ้น เพื่อให้เกิดการพัฒนาสินค้าและบริการใหม่ๆ ได้รวดเร็วขึ้น แล้วจึงนำสิ่งที่ได้ไปให้ผู้ใช้งานกลุ่มตัวอย่าง (Target group) ทดสอบใช้งานจริง จากนั้นจึงรวบรวมผลทดสอบมาประเมินดูอีกครั้ง เพื่อใช้เป็นแนวทางในการแก้ไขปรับปรุงสินค้าและบริการนั้นๆ ให้ดีขึ้นทีละนิด ด้วยแนวทางนี้จะทำให้องค์กรสามารถพัฒนาสินค้าและบริการได้อย่างรวดเร็วและตอบโจทย์ผู้ใช้งานได้มากขึ้นอย่างสม่ำเสมอ



รูปที่ 2.1 ความแตกต่างระหว่าง Waterfall Method กับ Agile Method

• Scrum (สกรัม) คือการนำแนวคิดในการทำงานแบบ Agile (อไจล์) มาปฏิบัติตามขั้นตอนของสกรัม เพื่อระบุปัญหาที่มีความซับซ้อน เปลี่ยนแปลงบ่อย เพื่อให้สามารถส่งมอบผลิตภัณฑ์ที่ตอบสนองต่อการเปลี่ยนแปลงที่เกิดขึ้นได้อย่างรวดเร็ว ช่วยให้การพัฒนาผลิตภัณฑ์แบบ Agile มีขั้นตอนการทำงานและผลลัพธ์ที่ชัดเจน โปร่งใส สามารถตรวจสอบประสิทธิภาพของแต่ละขั้นตอนการทำงาน 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

2.1.8 Smart Contract [8]

Smart Contract หมายถึง กระบวนการทางดิจิทัล ที่กำหนดขั้นตอนการทำธุรกรรมโดยอัตโนมัติไว้ล่วงหน้า โดยไม่ต้องอาศัยตัวกลาง อย่างเช่น ธนาคาร ซึ่งการสร้าง Smart Contract ที่เป็นระบบอัตโนมัติอย่างเต็มรูปแบบ โดยคู่สัญญาทั้งสองฝ่ายจะมีการตกลงกันก่อนหน้านั้น ถึงขั้นตอน กลไก ในการทำการธุรกรรมดังกล่าว ซึ่งการพัฒนานี้ส่งผลกระทบต่อรูปแบบธุรกิจแบบดั้งเดิมของธนาคาร

2.1.9 Non-Fungible Token (NFT) [9]

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

2.1.10 Asset (สินทรัพย์) [10]

Asset หมายถึง ทรัพยากรที่มีและอยู่ในการควบคุมของกิจการ สินทรัพย์นั้นอาจจะเป็นสิ่งที่มีความเคลื่อนไหวหรือไม่มีความเคลื่อนไหว ซึ่งสามารถตีราคามูลค่าเป็นเงินได้ ทรัพยากรดังกล่าวเป็นผล ของ เหตุการณ์ในอดีต ซึ่งกิจการคาดว่าจะได้รับประโยชน์เชิงเศรษฐกิจจากทรัพยากรนั้นในอนาคต

2.1.11 Digital Asset (สินทรัพย์ดิจิทัล) [11]

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

2.1.12 พินัยกรรม [12]

พินัยกรรม หมายถึง การแสดงเจตนากำหนดการเผื่อตายซึ่งให้มีผลบังคับได้เมื่อถึงแก่ความตาย หรือถ้าเป็นภาษาพูดก็ได้แก่คำสั่งเสียของผู้ตาย โดยในการทำพินัยกรรม กฎหมายกำหนดรูปแบบไว้ 5 แบบด้วยกัน ดังนี้

2.1.12.1 พินัยกรรมแบบธรรมดา ผู้ทำต้องทำเป็นหนังสือ คือการพิมพ์ข้อความพินัยกรรมลงในกระดาษ มากน้อยหรือจำนวนกี่แผ่นก็ต้องแล้วแต่เนื้อหาหรือจำนวนทรัพย์สิน ลงวัน เดือน ปี ที่ทำให้ชัดเจน และผู้ทำต้องลงลายมือชื่อไว้ต่อหน้าพยานอย่างน้อย 2 คน และพยานต้องลงลายมือชื่อรับรองการทำพินัยกรรมในขณะทำด้วย

2.1.12.2 พินัยกรรมแบบเขียนเองทั้งฉบับ ผู้ทำพินัยกรรมจะทำเป็นเอกสารเขียนเองทั้งฉบับก็ได้ แต่ผู้ทำนั้นต้องเขียนพินัยกรรมนั้นด้วยลายมือตนเอง ลงวัน เดือน ปี ที่ทำ และที่สำคัญต้องลงลายมือชื่อผู้ทำด้วย กรณีนี้จะมีพยานมารับรู้การทำพินัยกรรมด้วยหรือไม่ก็ได้

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

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

2.1.12.5 พินัยกรรมแบบทำด้วยวาจา กรณีมีพฤติการณ์พิเศษที่บุคคลไม่สามารถทำพินัยกรรมแบบอื่นที่กล่าวมาข้างต้น เช่น การตกอยู่ในอันตรายใกล้ความตาย หรืออยู่ในระหว่างสงคราม หรือเกิดมีโรคระบาด เราสามารถทำพินัยกรรมแบบทำด้วยวาจาก็ได้ โดยผู้ทำพินัยกรรมต้องแสดงเจตนาทำพินัยกรรมต่อหน้าพยานอย่างน้อย 2 คนพร้อมกัน พยานต้องรับฟังข้อความนั้นแล้วไปแจ้งต่อทางราชการโดยเร็วที่สุด ทั้งยังต้องแจ้งวัน เดือน ปี สถานที่ทำพินัยกรรมและพฤติการณ์พิเศษนั้นด้วย เจ้าพนักงานต้องจดข้อความที่พยานแจ้งไว้ และพยาน 2 คนนั้นต้องลงลายมือชื่อไว้

ข้อจำกัดและข้อควรระวังในการทำพินัยกรรม

1. พินัยกรรมเป็นนิติกรรมที่ต้องทำตามแบบที่กำหนดเท่านั้น
2. ต้องเขียน วัน เดือน ปี ลงลายมือชื่อทั้งผู้ทำพินัยกรรมและผู้ที่เป็นพยาน
3. ผู้ที่เป็นพยานจะต้องไม่เป็นผู้อยู่หรือผู้ย่อนความสามารถ และต้องไม่เป็นผู้มีส่วนได้เสียในกองมรดกนั้นด้วย

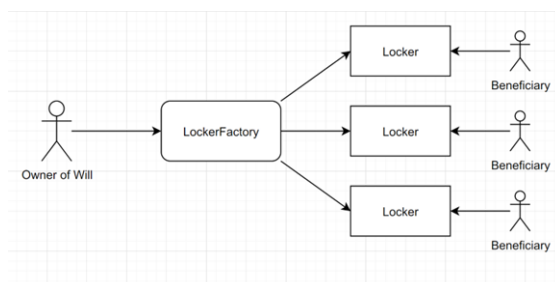
4. ผู้ทำพินัยกรรมต้องมีอายุ 15 ปีบริบูรณ์ขึ้นไป
5. พินัยกรรมควรจะต้องผู้จัดการมรดกโดยสามารถระบุผู้ทำหน้าที่ผู้จัดการมรดกที่เจ้ามรดกไว้ใจลงในพินัยกรรมไปได้เลย
6. สิทธิ หน้าที่ และความรับผิดชอบ ก็สามารถกำหนดในพินัยกรรมได้
7. ทรัพย์สินที่ระบุในพินัยกรรมต้องเป็นทรัพย์สินหรือสิทธิของผู้ทำพินัยกรรมเท่านั้น ทั้งต้องแยกสินส่วนตัวออกจากสินสมรสด้วย
8. เงินประกันชีวิต เงินบำเหน็จตกทอด เงินมีบ้านอายุตกทอด เงินฌาปนกิจสงเคราะห์ตกทอด ไม่อาจเป็นมรดกที่ระบุลงในพินัยกรรมได้ เพราะไม่ใช่ทรัพย์สินที่เจ้ามรดกมีอยู่ก่อนตาย

2.2 งานวิจัยที่เกี่ยวข้อง

ในส่วนนี้เป็นการสรุปเนื้อหาโดยรวมของงานวิจัยที่เกี่ยวข้องกับการพัฒนาโครงการงาน Will Chain Web application ที่มีการใช้งานในส่วนของ Digital Asset คือ CryptoWill

2.2.1 CryptoWill [13]

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



รูปที่ 2.2 ภาพแสดงหลักการทำงานของโปรเจกต์ CryptoWill

โดยเจ้าของพินัยกรรมจะทำพินัยกรรมและจะเก็บสินทรัพย์ไว้ใน blockchain และหลังจากนั้นจะส่งต่อให้ทายาทไปเมื่อถึงเวลาของพินัยกรรม

2.3 เทคนิคและเทคโนโลยีที่ใช้

2.3.1 Ethereum Chain [5]

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

2.3.2 GitHub [14]

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

2.3.3 MetaMask [15]

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)

2.3.4 NestJS [16]

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)

2.3.5 Next.js [17]

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

2.3.6 Solidity [18]

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

2.3.7 TypeScript [19]

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

2.3.8 Web3.js [20]

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

2.3.9 Hardhat [21]

Hardhat เป็น Development Environment ที่ทำให้เราสามารถพัฒนาตัว Smart Contract ได้โดย Hardhat สามารถทำ Compile, Deploy, Test, Debug ของตัว Smart contract และ สามารถ Test ตัว Smart Contract ได้บน Local Network ตัวเองได้

2.3.10 Pinata [22]

Pinata เป็น Development Environment ที่ทำให้เราสามารถพัฒนาตัว Smart Contract ได้โดย Hardhat สามารถทำ Compile, Deploy, Test, Debug ของตัว Smart contract และ สามารถ Test ตัว Smart Contract ได้บน Local Network ตัวเองได้

2.3.11 IPFS [23]

IPFS เป็นเครือข่ายกระจายอำนาจแบบเพียร์ทูเพียร์ที่ช่วยให้ผู้ใช้สำรองไฟล์และเว็บไซต์โดยการโฮสต์ไว้บนโหนดจำนวนมาก ถูกสร้างขึ้นโดย Protocol Labs เป็นบริการที่อาศัยเครือข่ายคอมพิวเตอร์แบบกระจายที่โฮสต์เนื้อหา เช่น หน้าเว็บ ไฟล์ และแอปที่มีเรอร์ ซึ่งทั้งหมดนี้คุณสามารถดึงขึ้นมาได้โดยการป้อนลิงก์ Pinatas เป็นบริการโฮสต์ NFT ที่ใช้ IPFS เพื่อสำรองข้อมูลของสะสม crypto สำหรับคู่ค้าเช่น Rarible และ Sorare

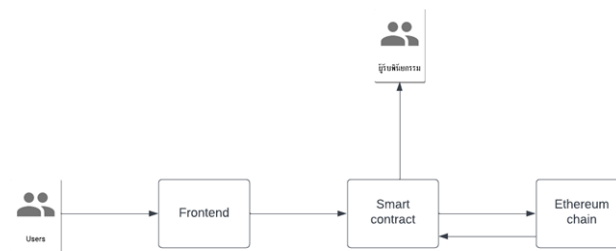
บทที่ 3 การออกแบบและวิธีการดำเนินงาน

เอกสารรายงานบทนี้จะกล่าวถึงระบบการทำงานของโครงการ รวมถึงแผนภาพต่าง ๆ ที่ใช้อธิบายการทำงานในส่วนต่าง ๆ ของระบบ การออกแบบส่วนติดต่อผู้ใช้งาน (User Interface) โดยอะแกรมของระบบ รวมถึงขั้นตอนและวิธีการดำเนินงาน

3.1 ระบบการทำงาน

3.1.1 ภาพรวมของระบบ

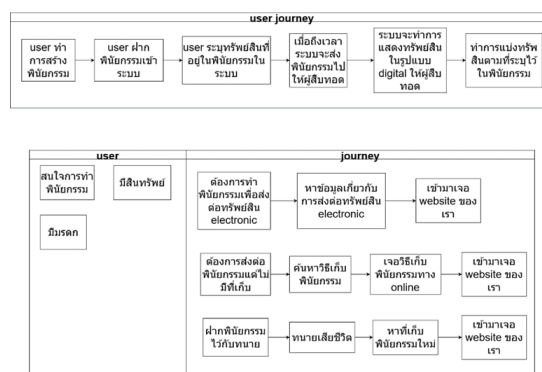
โดยภาพรวมของ Will Chain (Web application) จะประกอบไปด้วย 3 ส่วนหลักดังนี้



รูปที่ 3.1 ภาพรวมแสดงการทำงานของระบบ

- ส่วนติดต่อผู้ใช้งานหรือ Frontend จะเป็นส่วนที่ผู้ใช้งานเห็น และใช้งาน
- Smart Contract จะเป็นส่วนที่ควบคุมการทำงานของระบบทั้งหมด โดยที่ผู้ใช้งานจะเข้าใช้งานผ่านทาง Frontend และจะส่งชุดคำสั่งมาเพื่อที่จะสั่งให้ Smart Contract นั้นทำงาน และจะส่งข้อมูลไปเก็บใน Blockchain ต่อไป
- Ethereum chain สำหรับเก็บข้อมูลการใช้งานของผู้ใช้งาน

3.1.2 User Journey



รูปที่ 3.2 แสดง User Journey

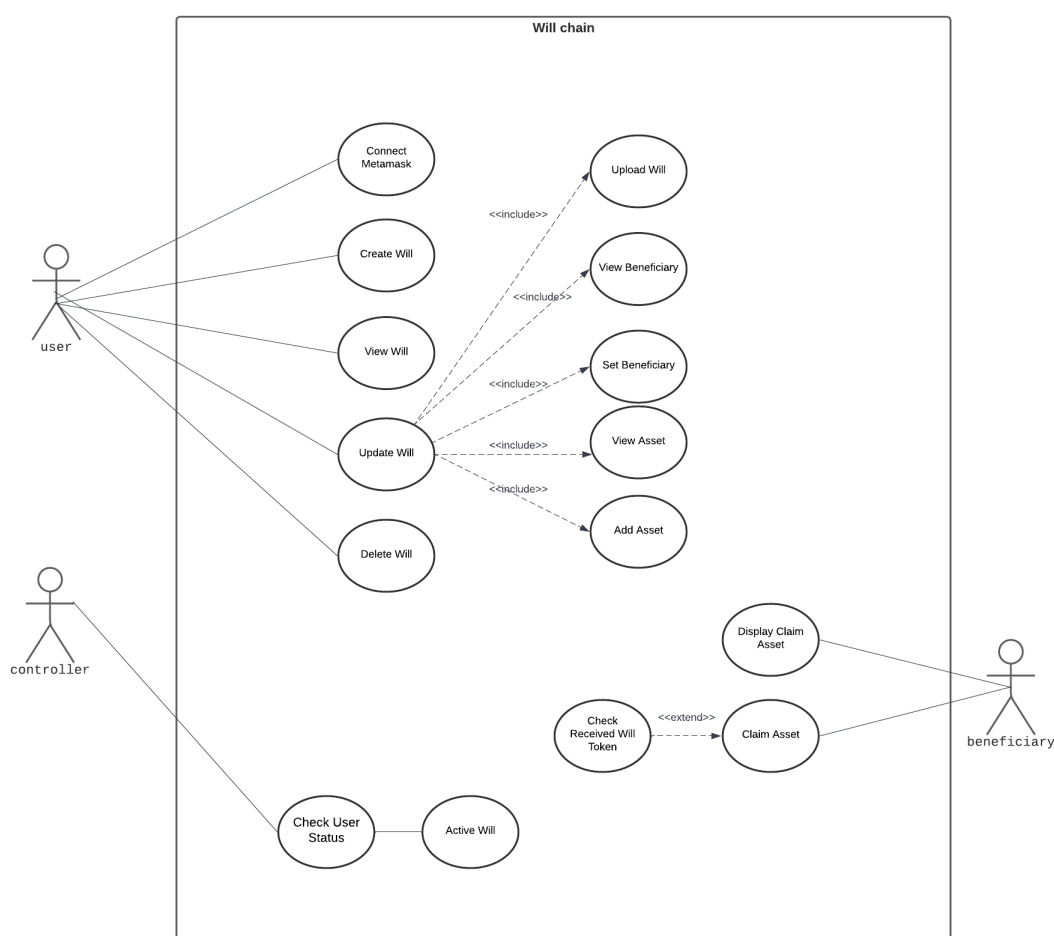
3.2 Cryptocurrency Wallet

ในการออกแบบระบบการทำงานของ Will Chain web application ได้เลือกใช้งาน Cryptocurrency Wallet ที่สามารถเชื่อมต่อกับ Ethereum chain ได้ เพื่อที่จะทำให้สามารถทดสอบ และใช้งานจริงได้บน Ethereum chain โดย Cryptocurrency Wallet โดยเลือกใช้เป็น Metamask Wallet

3.3 Diagram Unified Modelling Language (UML)

หลังจากได้เขียนความต้องการ และฟังก์ชันแล้ว จึงทำการออกแบบและเขียนแผนภาพไดอะแกรมต่าง ๆ เพื่อให้สามารถเข้าใจระบบการทำงานมากยิ่งขึ้น โดยมีรายละเอียดดังนี้

3.3.1 แผนภาพ Use Case Diagram



รูปที่ 3.3 แสดงการทำงานของระบบทั้งหมด Use Case Diagram

จากรูปแสดง Use Case แสดงการทำงานของระบบทั้งหมดโดยจะมีผู้ใช้งาน (User) ที่ต้องการฝากพินัยกรรมไว้ในระบบทำการใช้งานระบบผ่าน platform ของ Will Chain Web application โดยผู้ใช้งานจำเป็นต้องเชื่อมต่อกระเป๋าเงินอิเล็กทรอนิกส์ของ meta mask ก่อนหลังจากนั้นจึงจะสามารถ สร้าง,ลบหรือ upload พินัยกรรมพร้อมทั้งมีระบบจัดการทรัพย์สินที่ผู้ใช้นั้นใช้แนบไว้พร้อมกับพินัยกรรม และจะมีระบบที่ทำการตรวจสอบสถานะของผู้ใช้งานพื้นที่ที่จะทำการส่งผ่านพินัยกรรมไปยังผู้รับผลประโยชน์เมื่อถึงเวลา จากแผนภาพ Use Case Diagram ตามรูปที่ สามารถอธิบายรายละเอียดการทำงานของแต่ละ Use Case ได้ดังต่อไปนี้ โดยจะกล่าวถึงในหัวข้อ Use Case Narrative ถัดไป

3.3.2 Use Case Narrative

3.3.2.1 Use Case Connect Wallet

ตารางที่ 3.1 ตารางแสดงรายละเอียดของ Use Case Connect MetaMask Wallet

Use Case Name:	Connect MetaMask wallet	
Actors:	User	
Pre-Condition:	User ต้องทำการสร้างกระเป๋าเงิน MetaMask	
Post-Condition:	กระเป๋าเงินเชื่อมต่อกับ platform	
Brief Description:	User	System
Flow of Event:	1.เลือกเมนู Connect Wallet 3.ยืนยันการเชื่อมต่อ MetaMask Wallet	2.รอผู้ใช้งานเลือก Account และยืนยันการเชื่อมต่อ 4.เชื่อมต่อ MetaMask Wallet กับ Platform
Exception:		

3.3.2.2 Use Case Create Will

ตารางที่ 3.2 ตารางแสดงรายละเอียดของ Use Case Create Will

Use Case Name:	Create Will	
Actors:	User	
Pre-Condition:	User ต้องทำการเชื่อมบัญชีกับกระเป๋าเงิน meta mask	
Post-Condition:	พินัยกรรมถูกบันทึกเข้าระบบ	
Brief Description:	User	System
Flow of Event:	1.เลือกเมนู Create Will 3.ยืนยันการสร้างพินัยกรรม	2.รอผู้ใช้งานกรอกรายละเอียดพินัยกรรมให้เสร็จ 4.บันทึกพินัยกรรมเข้าสู่ระบบ
Exception:		

3.3.2.3 Use Case View Will

ตารางที่ 3.3 ตารางแสดงรายละเอียดของ Use Case View Will

Use Case Name:	View Will	
Actors:	User	
Pre-Condition:	User ต้องมีพินัยกรรมที่ถูกสร้างไว้เรียบร้อยแล้ว	
Post-Condition:	ระบบทำการแสดงพินัยกรรมในระบบให้ User	
Brief Description:	User	System
Flow of Event:	1.เลือกเมนู View Will 3.เลือกพินัยกรรมที่อยู่ในระบบที่ต้องการดู	2.ระบบทำการแสดงพินัยกรรมที่ถูกบันทึกในระบบ 4.ทำการแสดงพินัยกรรมที่ User เลือก
Exception:		

3.3.2.4 Use Case Upload Will

ตารางที่ 3.4 ตารางแสดงรายละเอียดของ Use Case Upload Will

Use Case Name:	Upload Will	
Actors:	User	
Pre-Condition:	User ต้องทำการเชื่อมบัญชีกับกระเป๋าเงิน meta mask	
Post-Condition:	พินัยกรรมถูกบันทึกเข้าระบบ	
Brief Description:	User	System
Flow of Event:	1.เลือกเมนู Upload Will 2.ยืนยันการ Upload พินัยกรรม	3.บันทึกพินัยกรรมเข้าสู่ระบบ
Exception:		

3.3.2.5 Use Case Delete Will

ตารางที่ 3.5 ตารางแสดงรายละเอียดของ Use Case Delete Will

Use Case Name:	Delete Will	
Actors:	User	
Pre-Condition:	User ต้องมีพินัยกรรมที่ถูกสร้างไว้เรียบร้อยแล้ว	
Post-Condition:	พินัยกรรมถูกลบออกจากระบบ	
Brief Description:	User	System
Flow of Event:	1.เลือกเมนู Delete Will 3.เลือกพินัยกรรมที่อยู่ในระบบ 4.ลบพินัยกรรมในระบบ 5.ยืนยันการลบพินัยกรรมในระบบ	2.ระบบทำการแสดงพินัยกรรมที่ถูกบันทึกในระบบ 6. ทำการนำพินัยกรรมออกจากระบบ
Exception:		

3.3.2.6 Use Case View Beneficiary

ตารางที่ 3.6 ตารางแสดงรายละเอียดของ Use Case View Beneficiary

Use Case Name:	View Beneficiary	
Actors:	User	
Pre-Condition:	ระบบต้องมีการเพิ่ม beneficiary แล้ว	
Post-Condition:	ระบบทำการแสดง beneficiary	
Brief Description:	User	System
Flow of Event:	1.เลือกพินัยกรรมที่ต้องการดู 3.เลือกเมนู View Beneficiary	2.ระบบทำการแสดงพินัยกรรมที่ถูกบันทึกในระบบ 4.ระบบจะทำการแสดง Beneficiary ของพินัยกรรม
Exception:		

3.3.2.7 Use Case Set Beneficiary

ตารางที่ 3.7 ตารางแสดงรายละเอียดของ Use Case Set Beneficiary

Use Case Name:	Set Beneficiary	
Actors:	User	
Pre-Condition:	ระบบต้องสร้างพินัยกรรมก่อน	
Post-Condition:	ผู้รับพินัยกรรมถูกบันทึกเข้าสู่ระบบ	
Brief Description:	User	System
Flow of Event:	1.เลือกพินัยกรรมที่ต้องการเพิ่ม beneficiary 3.เลือกเมนู Set Baneficiary	2.ระบบทำการแสดงพินัยกรรมที่ถูกบันทึกในระบบ 4.ระบบจะทำการบันทึก Beneficiary ของพินัยกรรมเข้าสู่ระบบ
Exception:		

3.3.2.8 Use Case View Asset

ตารางที่ 3.8 ตารางแสดงรายละเอียดของ Use Case View Asset

Use Case Name:	View Asset	
Actors:	User	
Pre-Condition:	ระบบต้องมีการเพิ่ม Asset แล้ว	
Post-Condition:	ระบบทำการแสดง Asset	
Brief Description:	User	System
Flow of Event:	1.เลือกพินัยกรรมที่ต้องการดู 3.เลือกเมนู View Asset	2.ระบบทำการแสดงพินัยกรรมที่ถูกบันทึกในระบบ 4.ระบบจะทำการแสดง Asset ของพินัยกรรม
Exception:		

3.3.2.9 Use Case Add Asset

ตารางที่ 3.9 ตารางแสดงรายละเอียดของ Use Case Add Asset

Use Case Name:	Add Asset	
Actors:	User	
Pre-Condition:	User ต้องทำการ upload พินัยกรรมเข้าสู่ระบบ	
Post-Condition:	พินัยกรรมถูกบันทึกเข้าระบบ	
Brief Description:	User	System
Flow of Event:	1.เลือกเมนู Add Asset 2.ทำการ upload สิ้นทรัพย์	3.บันทึกสินทรัพย์ไว้ในพินัยกรรม
Exception:		

3.3.2.10 Use Case Check User Status

ตารางที่ 3.10 ตารางแสดงรายละเอียดของ Use Case Check User Status

Use Case Name:	Check user status	
Actors:	Controller	
Pre-Condition:	ระบบต้องทำการเชื่อม API กับเว็บไซต์กรมการปกครอง	
Post-Condition:	ระบบทำการตรวจสอบสถานะของ User	
Brief Description:	User	System
Flow of Event:	1.Controller ตรวจสอบสถานะของ User 2.ทำการ active will เพื่อดำเนินการส่งไปหา beneficiary	3.ระบบทำการเปลี่ยนสถานะของพินัยกรรม และส่งไปหาผู้รับพินัยกรรม
Exception:		

3.3.2.11 Use Case Active Will

ตารางที่ 3.11 ตารางแสดงรายละเอียดของ Use Case Active Will

Use Case Name:	Active Wil	
Actors:	Controller	
Pre-Condition:	ระบบต้องเช็คสถานะ User Status เป็น Active	
Post-Condition:	ระบบทำให้พินัยกรรมดำเนินการส่งต่อไปหา Beneficiary	
Brief Description:	User	System
Flow of Event:	1.ทำการดำเนินการส่งพินัยกรรม ตามเลขบัตรประชาชนที่กำหนด	2.ทำการส่งพินัยกรรมจากระบบไปหา Beneficiary
Exception:		

3.3.2.12 Use Case Claim Will

ตารางที่ 3.12 ตารางแสดงรายละเอียดของ Use Case Claim Will

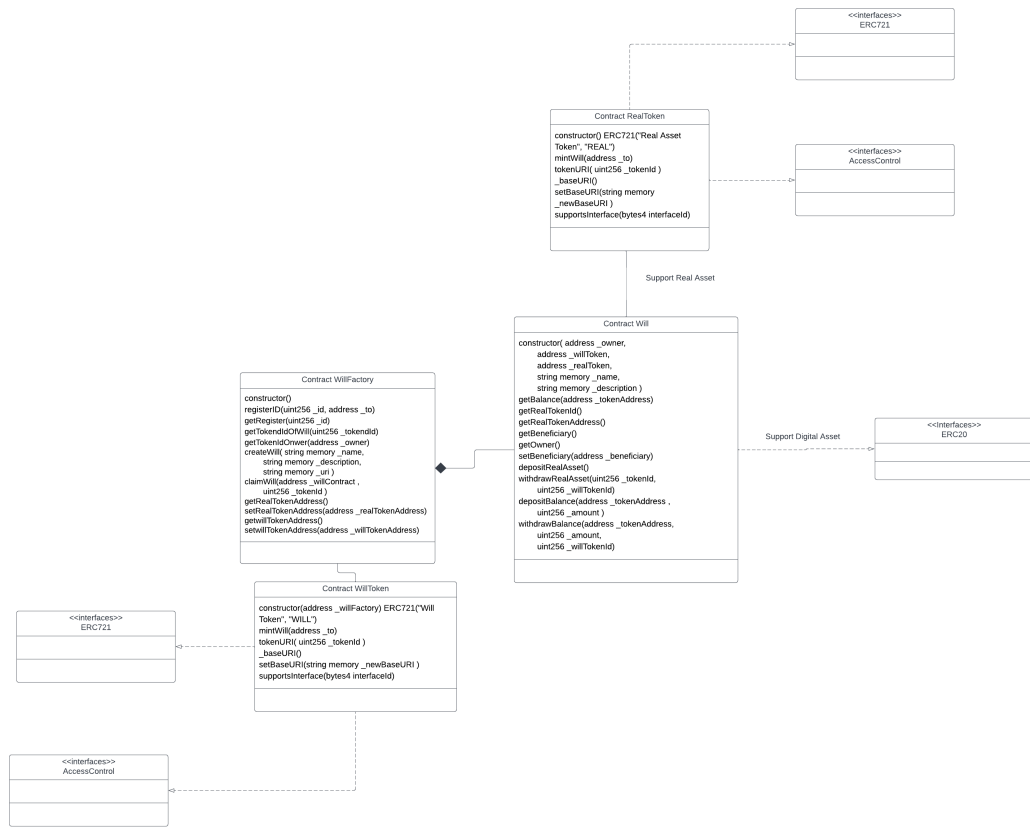
Use Case Name:	Claim Will	
Actors:	User	
Pre-Condition:	ระบบทำการแจ้งเตือนว่าผู้ทำพินัยกรรมเสียชีวิตแล้ว	
Post-Condition:	ผู้รับผลประโยชน์เข้ามารับพินัยกรรมและสินทรัพย์	
Brief Description:	User	System
Flow of Event:	1.เลือกเมนู Claim Will 3.เลือกพินัยกรรมในระบบที่สามารถรับได้	2.ระบบทำการแสดงพินัยกรรมที่ผู้รับผลประโยชน์สามารถรับได้ 4.ทำการส่งพินัยกรรมและสินทรัพย์ให้ผู้รับผลประโยชน์
Exception:		

3.3.2.13 Use Case Display Claim Asset

ตารางที่ 3.13 ตารางแสดงรายละเอียดของ Use Case Display Claim Asset

Use Case Name:	Display Claim Asset	
Actors:	Beneficiary	
Pre-Condition:	ระบบจะต้องมีพินัยกรรมที่ดำเนินการแล้ว	
Post-Condition:	แสดงสินทรัพย์จากพินัยกรรมที่สามารถรับได้	
Brief Description:	User	System
Flow of Event:	1.เลือกเมนู Display Will	2.แสดงสินทรัพย์ที่ผู้รับผลประโยชน์ได้รับ
Exception:		

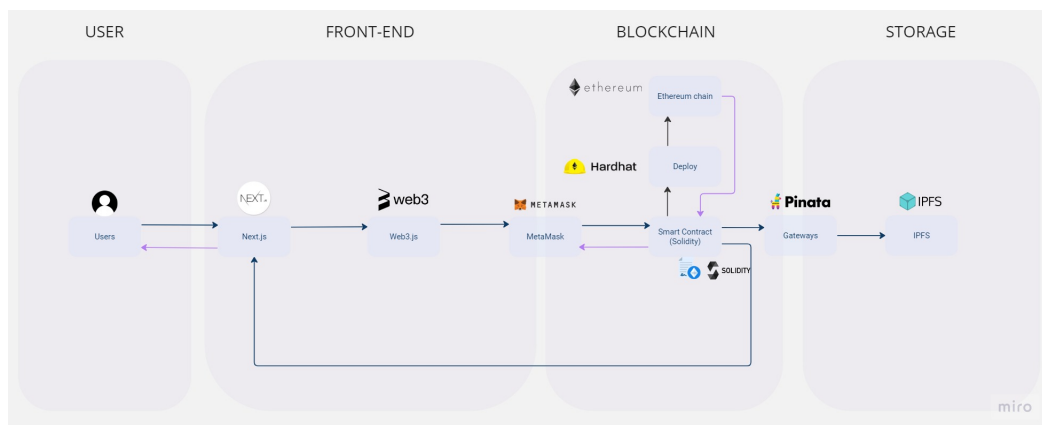
3.3.3 Smart Contract



รูปที่ 3.4 แสดงการ interaction ของ Smart Contract ของระบบ Will Chain

จากรูปแสดงการทำงานของตัว Smart Contract ของระบบ จะแบ่งเป็น 2 contract ที่ทำหน้าที่ต่างกันโดย will factory จะทำหน้าที่เก็บข้อมูลของทุกพินัยกรรมและรวมถึงการจัดการพินัยกรรมของระบบทั้งหมดโดย will factory สามารถทำการสร้างพินัยกรรมที่เป็น NFT ที่ใช้ ethereum standards ERC 721 และทำการ mint เก็บไว้ที่ตัวเจ้าของพินัยกรรมและเมื่อเกิดเหตุการณ์เสียชีวิตที่ได้รับจากระบบจะมีการ setStatusWill ให้เปลี่ยนเป็น active เพื่อที่จะทำการส่งต่อไปหาผู้รับพินัยกรรม ส่วน contract ต่อมาคือ will เป็น contract ที่ทำหน้าที่เกี่ยวกับจัดการมรดกของแต่ละพินัยกรรม โดยสามารถจัดการได้โดยเจ้าของพินัยกรรมว่าภายในพินัยกรรมของตัวเอง

3.3.4 System Architecture Diagram



รูปที่ 3.5 แสดงภาพออกแบบสถาปัตยกรรมของระบบ Will Chain

ระบบของ Will Chain มีส่วนติดต่อกับระบบอื่น ๆ แยกตามประเภทดังนี้

Actor ของระบบ

- User เป็นบุคคลที่ต้องการทำพันธกรรมของ Will Chain

Front-end ของระบบ

- Next.js จะทำหน้าที่แสดงผล UI ของเว็บไซต์ Will Chain ในการทำพันธกรรมต่าง ๆ
- Web3.js จะทำหน้าที่ interact กับ method ต่าง ๆ ใน smart contract
- MetaMask จะทำหน้าที่เป็นตัว wallet สำหรับเก็บทรัพย์สินของเราและยังทำหน้าที่เป็นตัว login สำหรับใช้งานในระบบ

Blockchain ของระบบ

- Smart contract จะทำหน้าที่คอยจัดการ transaction ภายใน Ethereum chain
- Deploy จะทำหน้าที่ deploy smart contract ขึ้นไปที่ ethereum chain
- Ethereum chain จะทำหน้าที่เก็บข้อมูล transaction และการทำพันธกรรมต่าง ๆ ของระบบ
- Pinata จะทำหน้าที่เป็น gateway ในการใช้งาน storage ของระบบพันธกรรม

Storage ของระบบ

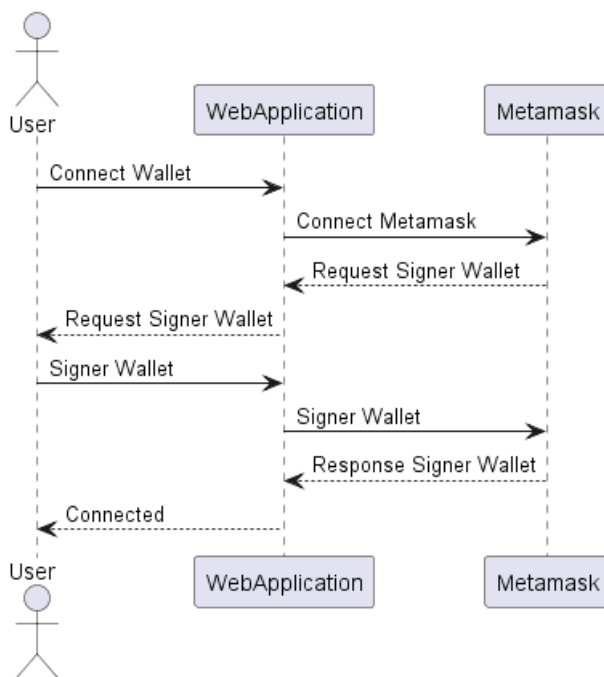
- IPFS จะทำหน้าที่เป็น storage ของระบบพันธกรรม

3.3.5 Sequence Diagram

3.3.5.1 Connect MetaMask

ตารางที่ 3.14 ตารางแสดงรายละเอียดของ Connect MetaMask Sequence Diagram

Sequence Name:	Connect MetaMask
Actors:	User
Pre-Condition:	User จะต้องทำการ Connect MetaMask เพื่อเป็นการ login ใช้ระบบ



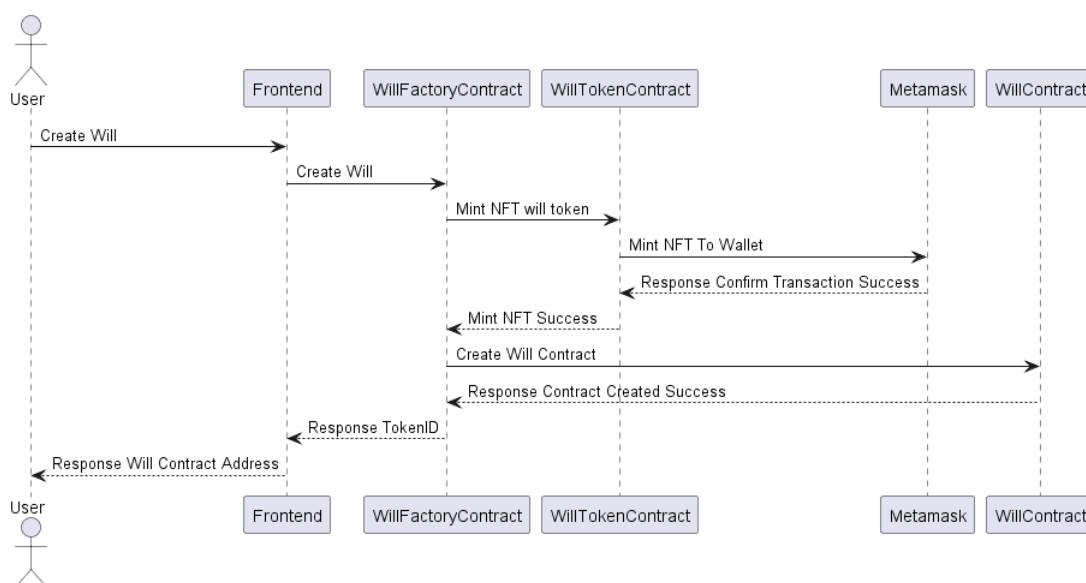
รูปที่ 3.6 แสดง Connect MetaMask Sequence Diagram

จากรูป จะเห็นได้ว่าเมื่อผู้ใช้งานทำงาน Connect MetaMask แล้วทาง Web Application จะไปเรียกใช้ MetaMask ที่ทำการติดตั้งไว้ใน Web browser ที่ทำการใช้งานอยู่ว่าขอทำการขอ Signer Wallet เพื่อทำการเชื่อมต่อ Web Application ซึ่งหลังจากทำการ Signer Wallet จาก User แล้วตัว Metamask จะทำการ Connected กับ Web Application

3.3.5.2 Create Will

ตารางที่ 3.15 ตารางแสดงรายละเอียดของ Create Will Sequence Diagram

Sequence Name:	Create Will
Actors:	User
Pre-Condition	User จะทำการ Create Will ที่เป็นการเขียนพินัยกรรมผ่านเว็บไซต์



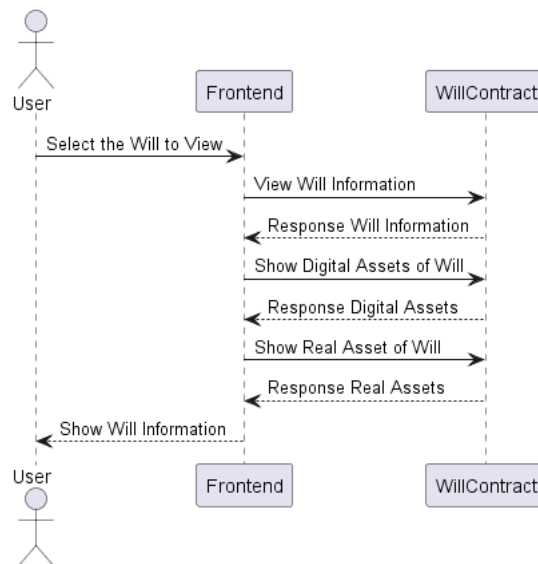
รูปที่ 3.7 แสดง Create Will Sequence Diagram

จากรูป ผู้ใช้เลือกใช้งาน Create Will จะแสดงหน้าจะแสดงฟอร์มสำหรับการทำพินัยกรรมผ่านระบบ จะทำการ Mint Will Token ไปที่ Metamask หลังจาก Mint NFT เสร็จสิ้นก็ทำการสร้าง Will Contract ที่จะทำหน้าที่จัดการสินทรัพย์หรือรายละเอียดพินัยกรรมต่างๆ ภายในพินัยกรรม หลังจากนั้นจะทำการแสดงผล Token id และ พินัยกรรม ที่ User มีอยู่

3.3.5.3 View Will

ตารางที่ 3.16 ตารางแสดงรายละเอียดของ View Will Sequence Diagram

Sequence Name:	View Will
Actors:	User
Pre-Condition	User ต้องการดูรายละเอียดพินัยกรรมที่เขียน



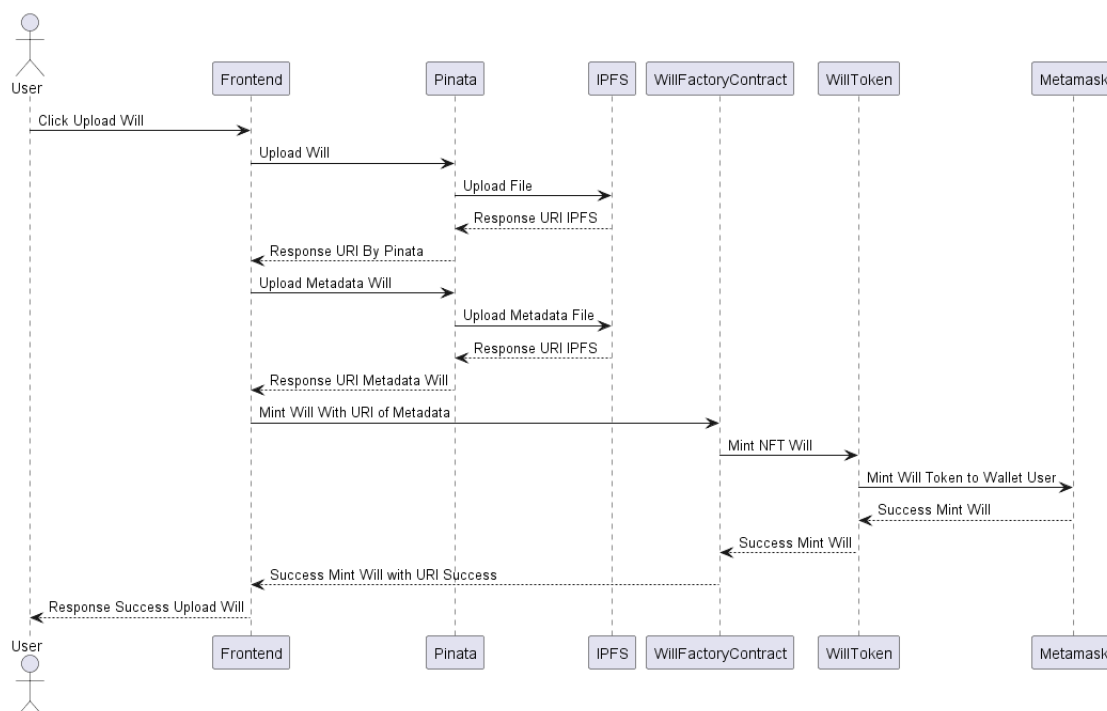
รูปที่ 3.8 แสดง View Will Sequence Diagram

จากรูป ผู้ใช้ทำการเลือกพินัยกรรมที่ต้องการจะแสดงโดยจะทำการเรียกใช้ฟังก์ชันเพื่อดูรายละเอียดใน Will Contract หลังจากนั้น Frontend จะเรียกใช้ฟังก์ชันแสดงสินทรัพย์ดิจิทัล และ เรียกใช้ฟังก์ชันแสดงสินทรัพย์จริง หลังจากนั้นจะทำการแสดงผลของรายละเอียดพินัยกรรม

3.3.5.4 Upload Will

ตารางที่ 3.17 ตารางแสดงรายละเอียดของ Upload Pdf Will Sequence Diagram

Sequence Name:	Upload Will
Actors:	User
Pre-Condition	User จะทำการ Upload ที่เป็นการเขียนพินัยกรรมด้วยลายมือ



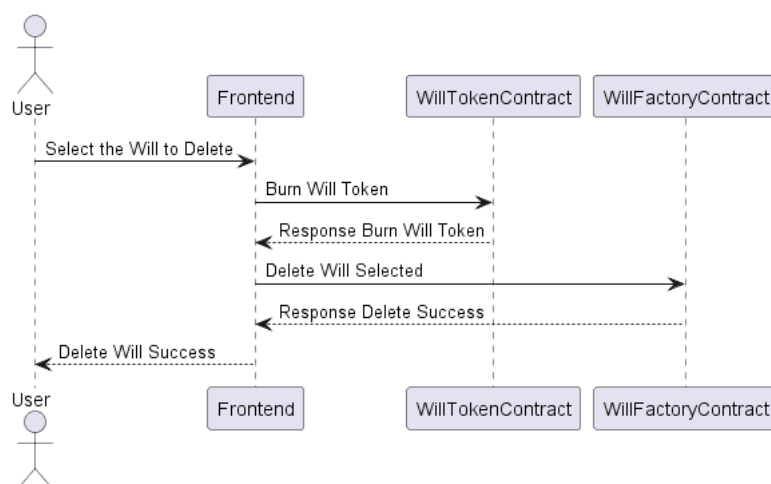
รูปที่ 3.9 แสดง Upload Will Sequence Diagram

จากรูป ผู้ใช้เลือกใช้งาน Upload พินัยกรรมที่เขียนด้วยมือระบบจะแสดงฟอร์มที่ใช้สำหรับการ Upload พินัยกรรมไปที่ Pinata และ Pinata ทำหน้าที่ Upload File ไปที่ IPFS หลังจากนั้น IPFS จะส่ง URI ไปที่ Pinata จะทำการส่งต่อไปที่ Frontend หลังจากนั้น Upload Metadata Will หลังจากนั้น Pinata จะ Upload Metadata Will ไปที่ IPFS หลังจากนั้น IPFS จะส่ง URI ไปที่ Pinata หลังจากนั้น ส่งไป Frontend เพื่อทำการ mint Will Token ออกมาโดยส่ง mint ไปที่ Will Factory Contract เพื่อทำการจัดการ Mint Will Token หลังจากนั้นจะทำการ Mint Will Token ไปที่ Metamask wallet ของ User หลังจากนั้น Frontend จะแสดงผลอัปเดตพินัยกรรมเสร็จสิ้น

3.3.5.5 Delete Will

ตารางที่ 3.18 ตารางแสดงรายละเอียดของ Delete Will Sequence Diagram

Sequence Name:	Delete Will
Actors:	User
Pre-Condition	User ต้องการที่จะลบพินัยกรรมที่เขียน



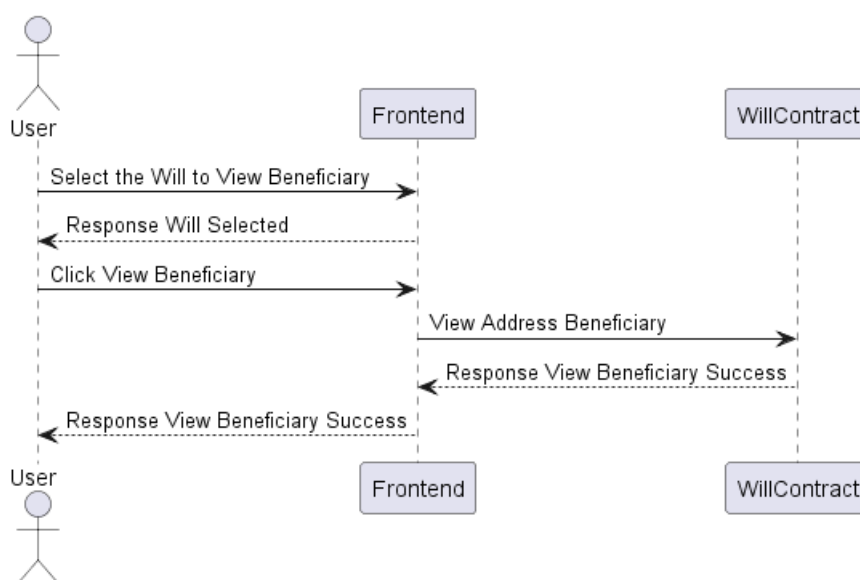
รูปที่ 3.10 แสดง Delete Will Sequence Diagram

จากรูป ผู้ใช้ต้องการทำการลบ พินัยกรรมที่มีอยู่ในระบบ โดยระบบจะทำการ Burn Will Token ของเหรียญที่แทนพินัยกรรมนั้น หลังจากนั้นทำการลบพินัยกรรมที่ทำการเลือกไว้ใน WillFactoryContract หลังจากลบเสร็จสิ้นจะแสดงผลหน้าจอว่าลบพินัยกรรมสำเร็จ

3.3.5.6 View Beneficiary

ตารางที่ 3.19 ตารางแสดงรายละเอียดของ View Beneficiary Sequence Diagram

Sequence Name:	View Beneficiary
Actors:	User
Pre-Condition	User ต้องการดูผู้รับพินัยกรรมที่อยู่ระบบ



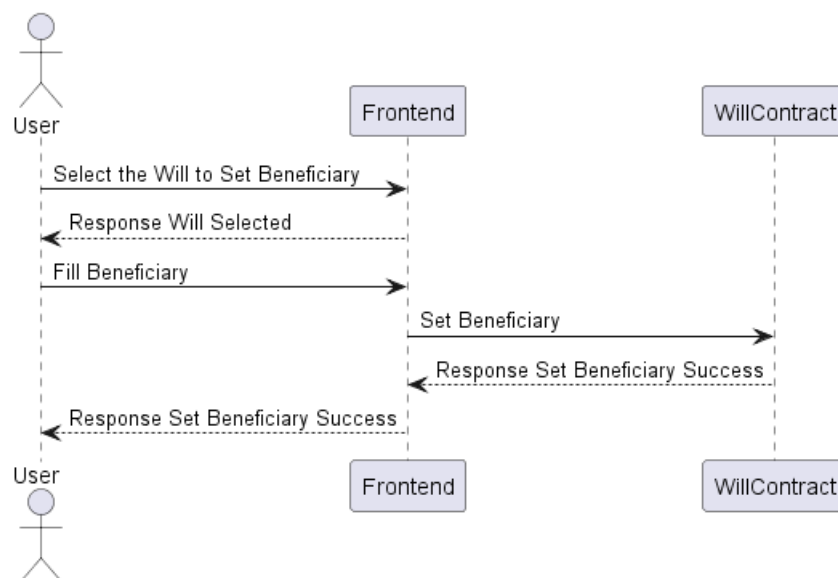
รูปที่ 3.11 แสดง View Beneficiary Sequence Diagram

จากรูป ผู้ใช้ต้องเลือกพินัยกรรมที่ต้องการดูผู้รับพินัยกรรม และจะแสดงรายละเอียดพินัยกรรมหลังจากนั้นคลิกดูพินัยกรรมและ Frontend จะเรียกฟังก์ชันแสดงผู้รับพินัยกรรมจาก Will Contract หลังจากได้รับข้อมูลจาก Will Contract ตัว Frontend จะทำการแสดงผล Address ของ Beneficiary

3.3.5.7 Set Beneficiary

ตารางที่ 3.20 ตารางแสดงรายละเอียดของ Set Beneficiary Sequence Diagram

Sequence Name:	Set Beneficiary
Actors:	User
Pre-Condition	User ต้องการเลือกผู้รับพินัยกรรมที่อยู่ระบบ



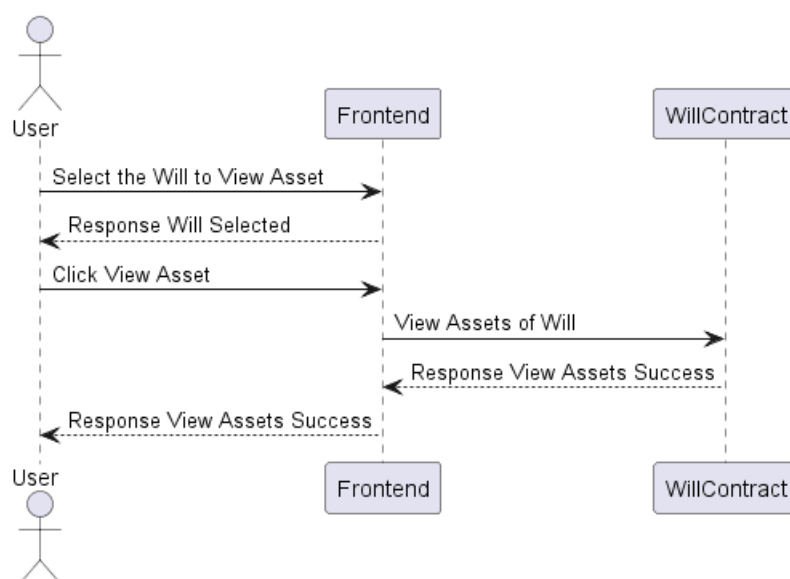
รูปที่ 3.12 แสดง Set Beneficiary Sequence Diagram

จากรูป ผู้ใช้ต้องเลือกพินัยกรรมที่ต้องการดูผู้รับพินัยกรรม และจะแสดงรายละเอียดพินัยกรรมหลังจากนั้นคลิกช่อง Beneficiary และใส่เลข Address ของ Beneficiary หลังจากนั้น Frontend จะทำการเรียกฟังก์ชัน setBeneficiary เพื่อทำการเพิ่มผู้รับพินัยกรรมใน will contract และหลังจากเพิ่มพินัยกรรมเสร็จสิ้นก็ทำการแสดงผลว่าเพิ่มผู้รับพินัยกรรมเสร็จสิ้น

3.3.5.8 View Assets

ตารางที่ 3.21 ตารางแสดงรายละเอียดของ View Assets Sequence Diagram

Sequence Name:	View Assets
Actors:	User
Pre-Condition	User ต้องการดูสินทรัพย์ที่เชื่อมต่อ Smart Contract



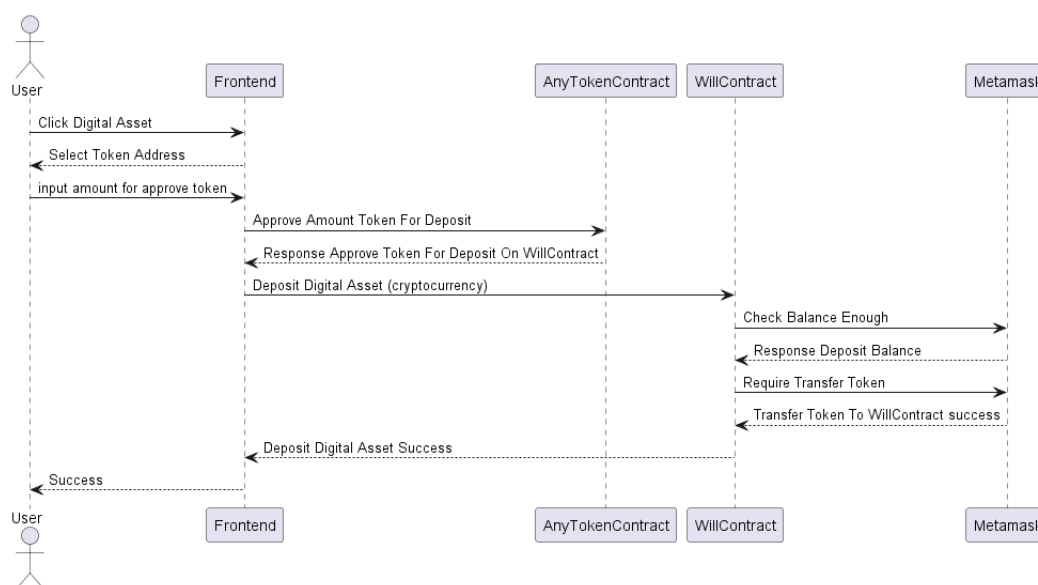
รูปที่ 3.13 แสดง View Assets Sequence Diagram

จากรูป ผู้ใช้ต้องเลือกพินัยกรรมที่ต้องการดูผู้รับพินัยกรรม และจะแสดงรายละเอียดพินัยกรรมหลังจากนั้นคลิกดูสินทรัพย์และ Frontend จะเรียกฟังก์ชันแสดงสินทรัพย์จาก Will Contract หลังจากได้รับข้อมูลจาก Will Contract ตัว Frontend จะทำการแสดงผลสินทรัพย์เสร็จสิ้น

3.3.5.9 Deposit Digital Assets

ตารางที่ 3.22 ตารางแสดงรายละเอียดของ Deposit Digital Assets Sequence Diagram

Sequence Name:	Deposit Digital Assets
Actors:	User
Pre-Condition	User ต้องการเพิ่มสินทรัพย์ดิจิทัลที่เชื่อมต่อ Smart Contract



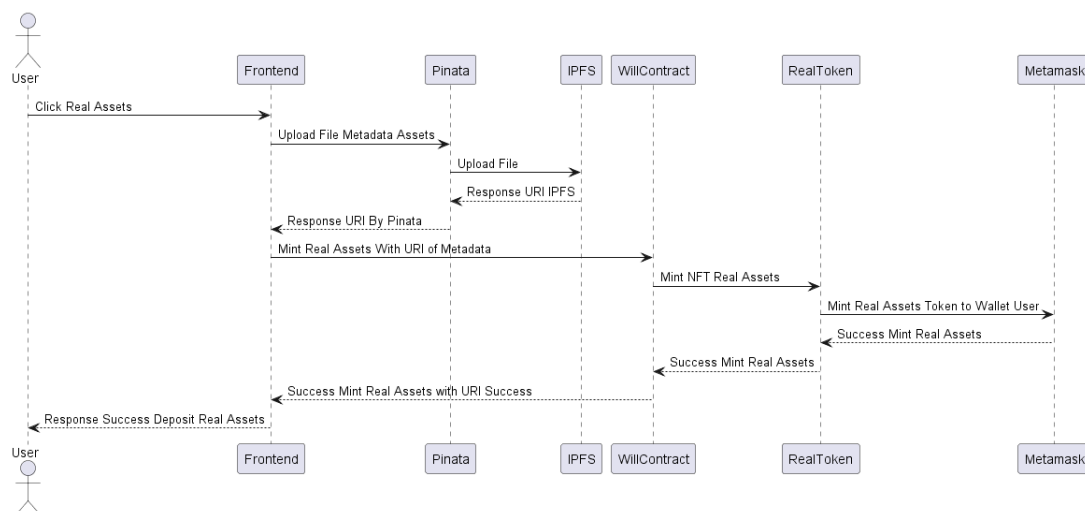
รูปที่ 3.14 แสดง Deposit Digital Assets Sequence Diagram

จากรูป ผู้ใช้ต้องการเพิ่มสินทรัพย์ดิจิทัลจะทำการคลิกเพื่อสินทรัพย์ดิจิทัลและทำการเลือกเหรียญที่ต้องการฝากไว้ในพินัยกรรม และ Frontend จะทำการ Approve จำนวนของตัว Token นั้นที่จะทำการ Deposit เข้า Smart Contract หลังจากนั้น Frontend จะทำการ Deposit Digital Assets ไปที่ Will Contract และจะทำการเช็คยอดเงินคงเหลือใน Metamask ว่าเพียงพอต่อการฝากและหลังจากนั้นจะใช้ฟังก์ชันใน Will Contract Transfer เหรียญที่อยู่ใน Metamask ไปที่ Will Contract และหลังจากนั้น Frontend จะแสดงผลการเพิ่มสินทรัพย์ดิจิทัลเสร็จสิ้น

3.3.5.10 Deposit Real Assets

ตารางที่ 3.23 ตารางแสดงรายละเอียดของ Deposit Real Assets Sequence Diagram

Sequence Name:	Deposit Real Assets
Actors:	User
Pre-Condition	User ต้องการเพิ่มสินทรัพย์จริงที่เชื่อมต่อ Smart Contract



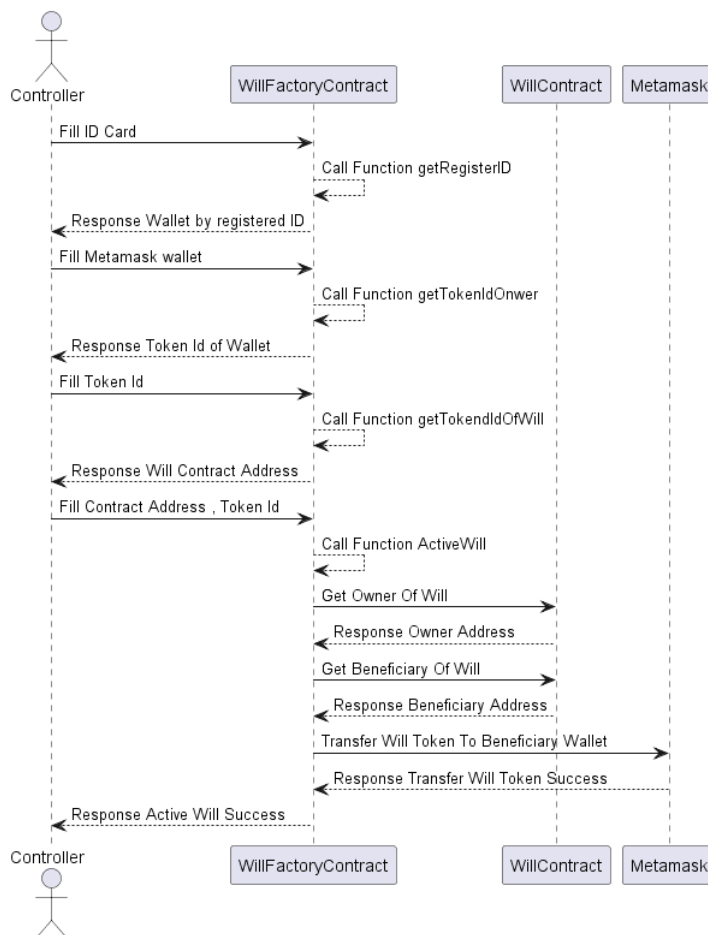
รูปที่ 3.15 แสดง Deposit Real Assets Sequence Diagram

จากรูป ผู้ใช้ต้องการเพิ่มสินทรัพย์จริงจะทำการคลิกเพื่อสินทรัพย์ดิจิทัล Frontend จะทำการอัปโหลด Metadata ของ Assets ไปที่ Pinata และ Pinata ทำหน้าที่ Upload File ไปที่ IPFS หลังจากนั้น IPFS จะส่ง URI ไปที่ Pinata จะทำการส่งต่อไปที่ Frontend หลังจากนั้น Frontend ทำการ mint Real Token ออกมาโดยส่ง mint ไปที่ Will Contract เพื่อทำการจัดการ Mint Real Token หลังจากนั้นจะทำการ Mint Real Token ไปที่ Metamask wallet ของ User หลังจากนั้น Frontend จะแสดงผลเพิ่มสินทรัพย์จริงเสร็จสิ้น

3.3.5.11 Active Will

ตารางที่ 3.24 ตารางแสดงรายละเอียดของ Active Will Sequence Diagram

Sequence Name:	Active Will
Actors:	Controller
Pre-Condition	ผู้กำกับพันยกรรมต้องการเริ่มการสืบทอดพันยกรรม



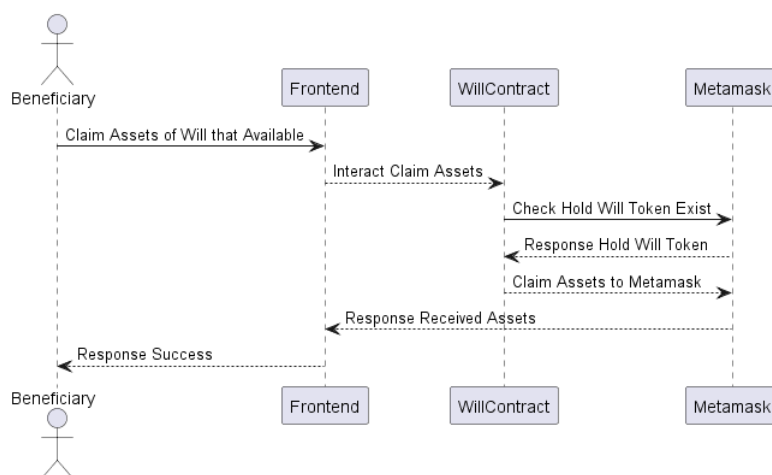
รูปที่ 3.16 แสดง Active Will Sequence Diagram

จากรูป ผู้ควบคุมจะทำการกรอก ID Card ไว้และเพื่อนำเลข Wallet Address ที่ทำการ register กับระบบไว้หลังจากนั้นจะกรอกเลขกระเป๋าเพื่อนำเลข Token id ที่เจ้าของพินัยกรรมถืออยู่มีอะไรบ้างหลังจากนั้น จะทำการกรอก token id เพื่อนำเลข Will Factory Contract address ไปทำการกรอกฟังก์ชัน ActiveWill เพื่อทำการให้พินัยกรรมนี้สามารถทำงานได้ โดยจะใช้กระเป๋าของเจ้าของพินัยกรรมและกระเป๋าของผู้รับพินัยกรรม และทำการส่ง Will Token ไปที่ กระเป๋าของผู้รับพินัยกรรม

3.3.5.12 Claim Assets

ตารางที่ 3.25 ตารางแสดงรายละเอียดของ Claim Assets Sequence Diagram

Sequence Name:	Claim Asset
Actors:	Beneficiary
Pre-Condition	ทายาทจะทำการรับสินทรัพย์ที่ได้รับจากการเขียนพินัยกรรม



รูปที่ 3.17 แสดง Claim Assets Sequence Diagram

จากรูป ทายาทจะรับสินทรัพย์ที่ได้รับจากการเขียนพินัยกรรมโดย Will Contract จะไปเช็คใน Metamask ว่า ลูกรู้ Will Token ที่สามารถ interact กับ Will Contract นี้ไหม หลังจากนั้นจะให้ผู้รับพินัยกรรมรับสินทรัพย์ที่อยู่ใน Will Contract ได้ไปที่ Metamask หลังจากนั้น Frontend จะแสดงผลรับสินทรัพย์เสร็จสิ้น

3.4 ส่วนติดต่อผู้ใช้งาน (User Interface)

การออกแบบส่วนติดต่อผู้ใช้งาน โดยการออกแบบ Will Chain ได้คำนึงถึงการใช้งานของผู้ใช้งาน

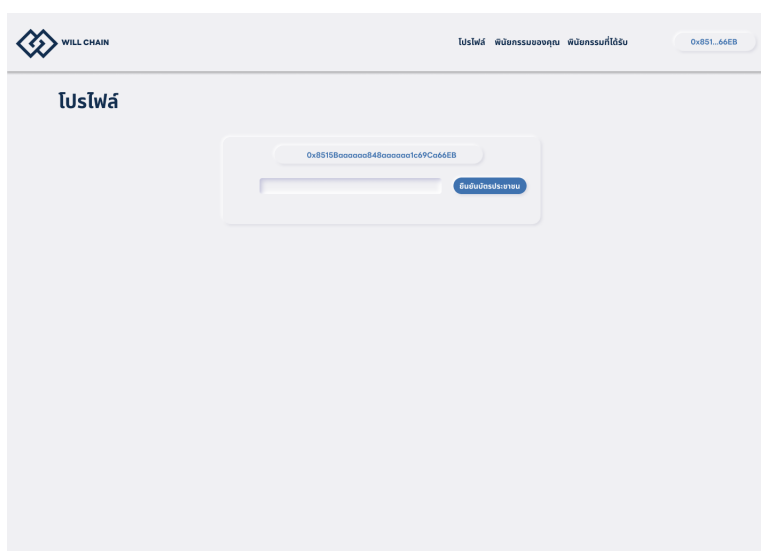
3.4.1 หน้าแรก



รูปที่ 3.18 หน้าแรก

จากรูปเป็นหน้าแรกของแพลตฟอร์ม Web application Will Chain ที่ยังไม่ได้ทำการเข้าสู่ระบบ โดยจะประกอบไปด้วย Concept ของแพลตฟอร์ม รวมไปถึงการเข้าถึงคู่มือการใช้งาน และยังสามารถที่ปุ่มขวาบนเพื่อเชื่อมต่อกับ MetaMask Wallet

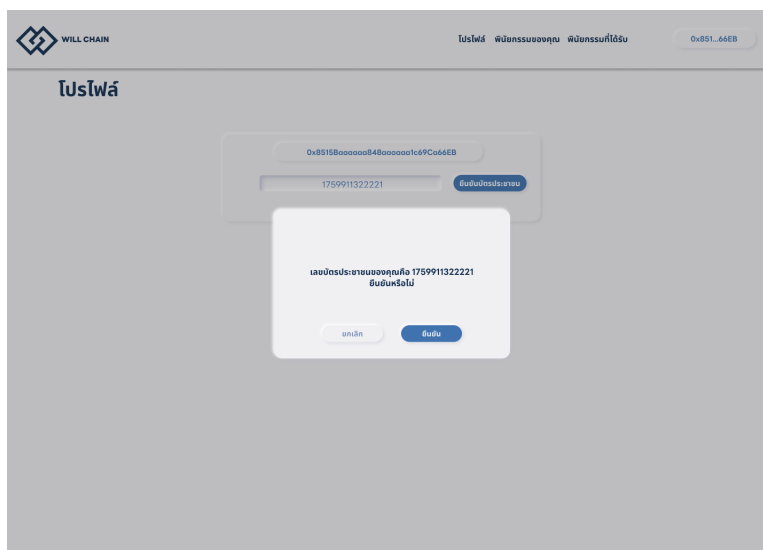
3.4.2 หน้าโปรไฟล์



รูปที่ 3.19 หน้าโปรไฟล์

จากรูปเป็นหน้าโปรไฟล์จะต้องทำการ Connect Metamask โดยจะแสดงเลข Public key ทางด้านขวาบน จะเข้ามาสู่หน้าโปรไฟล์ ที่จะมีแสดงเลข Public key ของ MetaMask และมีช่องให้ใส่เลขบัตรประจำตัวประชาชน เพื่อยืนยันสำหรับการสืบทอดพันธุกรรม

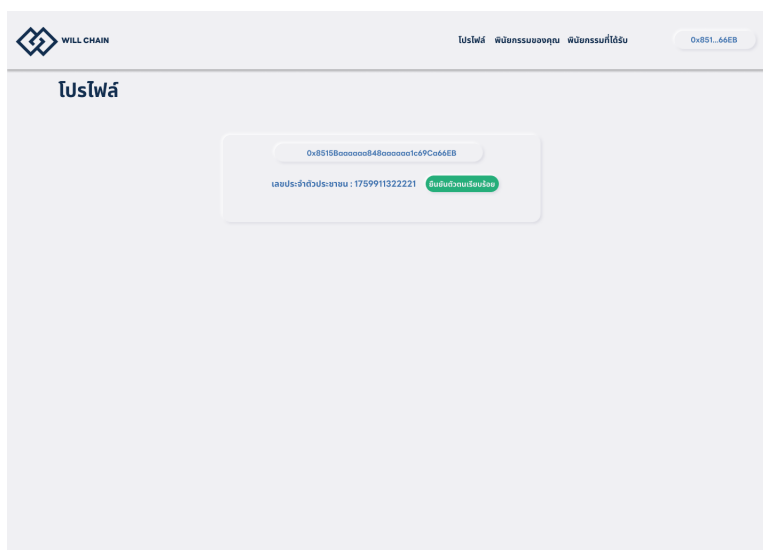
3.4.3 หน้าโปรไฟล์ยืนยันการลงทะเบียน



รูปที่ 3.20 หน้าโพรไฟล์ ยืนยันการลงทะเบียน

จากรูปเป็นหน้าโพรไฟล์หลักจากปุ่มยืนยันการลงทะเบียนด้วยรหัสเลขบัตรประชาชน

3.4.4 หน้าโพรไฟล์ลงทะเบียนสำเร็จ



รูปที่ 3.21 หน้าโพรไฟล์ ยืนยันการลงทะเบียนสำเร็จ

จากรูปเป็นหน้าโพรไฟล์ที่เมื่อทำการกรอกการลงทะเบียนด้วยเลขบัตรประชาชน

3.4.5 หน้าพินัยกรรมของคุณ

WILL CHAIN

โปรไฟล์ สิทธิประโยชน์ของคุณ ฟิลิปปินส์ของคุณ ฟิลิปปินส์ของคุณ 0x851...66EB

ฟิลิปปินส์ของคุณ

ฟิลิปปินส์ของคุณ: 0x8FE05C873EA7BB16D35d310B4EF296eB4BA960f

เลือกฟิลิปปินส์ของคุณ:

ชื่อฟิลิปปินส์: เลือกฟิลิปปินส์ของคุณ

ค่าธรรมเนียมฟิลิปปินส์: เลือกฟิลิปปินส์ของคุณ

ฟิลิปปินส์ของคุณ:

ผู้ส่งฟิลิปปินส์: 0x851...66EB

ผู้รับฟิลิปปินส์: 0x851...66EB

สิทธิประโยชน์ที่ได้รับ:

สิทธิประโยชน์ที่ได้รับ:

ฟิลิปปินส์ของคุณ: 0x8FE05C873EA7BB16D35d310B4EF296eB4BA960f

เลือกฟิลิปปินส์ของคุณ:

ชื่อฟิลิปปินส์: เลือกฟิลิปปินส์ของคุณ

ค่าธรรมเนียมฟิลิปปินส์: เลือกฟิลิปปินส์ของคุณ

ฟิลิปปินส์ของคุณ:

รูปที่ 3.22 หน้าฟิลิปปินส์ของคุณ

จากรูปเป็นหน้าฟิลิปปินส์ของคุณ จากการกดที่เมนู “ฟิลิปปินส์ของคุณ” ด้านบน โดยที่ในหน้านี้จะแสดงฟิลิปปินส์ที่มีอยู่ในระบบของผู้ใช้คนนี้ โดยที่ใน 1 ฟิลิปปินส์ จะแสดงเลขฉบับที่ของฟิลิปปินส์ สถานะของฟิลิปปินส์ และสามารถกดปุ่ม “ดูฟิลิปปินส์” เพื่อดูฟิลิปปินส์ที่เป็นไฟล์ฉบับจริงได้ โดยในตารางด้านล่างนั้นจะแสดงจำนวนผู้รับฟิลิปปินส์และรายชื่อของคนที่ได้รับฟิลิปปินส์ ต่อมาคือแสดงสิทธิประโยชน์ทั้งหมดที่อยู่ในฟิลิปปินส์ฉบับนี้ และแสดงสิทธิประโยชน์สุดท้ายคือแสดงสิทธิประโยชน์ดิจิทัลทั้งหมดในฟิลิปปินส์นั้น และแสดงชนิดของสิทธิประโยชน์ดิจิทัล โดยที่สามารถกดที่ปุ่ม “ดูข้อมูล” เพื่อดูข้อมูลเพิ่มเติม และสามารถกดปุ่ม “ลบฟิลิปปินส์” เพื่อลบฟิลิปปินส์นั้นออกจากระบบได้

3.4.6 หน้ายกเลิกการทำฟิลิปปินส์ของคุณ

WILL CHAIN

โปรไฟล์ สิทธิประโยชน์ของคุณ ฟิลิปปินส์ของคุณ ฟิลิปปินส์ของคุณ 0x851...66EB

ฟิลิปปินส์ของคุณ

ฟิลิปปินส์ของคุณ: 0x8FE05C873EA7BB16D35d310B4EF296eB4BA960f

เลือกฟิลิปปินส์ของคุณ:

ชื่อฟิลิปปินส์: เลือกฟิลิปปินส์ของคุณ

ค่าธรรมเนียมฟิลิปปินส์: เลือกฟิลิปปินส์ของคุณ

ฟิลิปปินส์ของคุณ:

ผู้ส่งฟิลิปปินส์: 0x851...66EB

ผู้รับฟิลิปปินส์: 0x851...66EB

สิทธิประโยชน์ที่ได้รับ:

สิทธิประโยชน์ที่ได้รับ:

ฟิลิปปินส์ของคุณ: 0x8FE05C873EA7BB16D35d310B4EF296eB4BA960f

เลือกฟิลิปปินส์ของคุณ:

ชื่อฟิลิปปินส์: เลือกฟิลิปปินส์ของคุณ

ค่าธรรมเนียมฟิลิปปินส์: เลือกฟิลิปปินส์ของคุณ

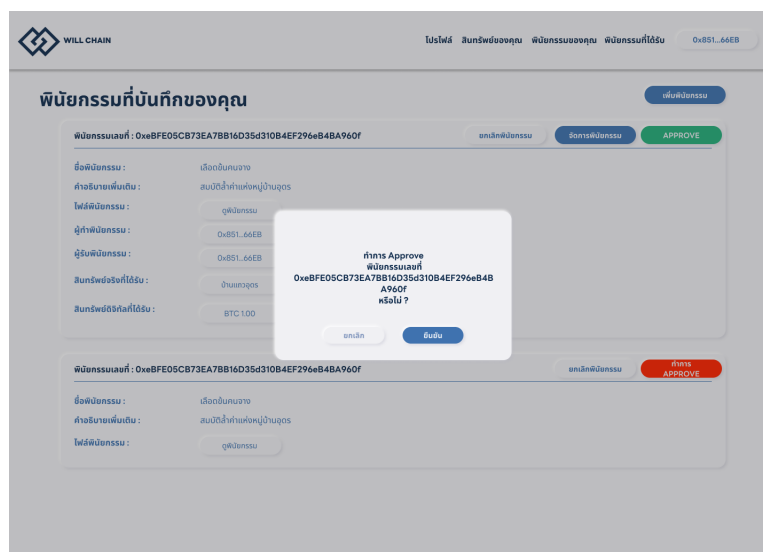
ฟิลิปปินส์ของคุณ:

ยกเลิกการทำฟิลิปปินส์ของคุณ
0x8FE05C873EA7BB16D35d310B4EF296eB4BA960f
หรือไม่ ?

รูปที่ 3.23 หน้ายกเลิกการทำฟิลิปปินส์ของคุณ

จากรูปเป็นจะเป็นหน้ายกเลิกการทำฟิลิปปินส์ จากการกดที่เมนู "ยกเลิกฟิลิปปินส์" ในหน้าฟิลิปปินส์ของคุณ โดยที่หน้านี้จะแสดงยืนยันการลบฟิลิปปินส์ฉบับนั้น

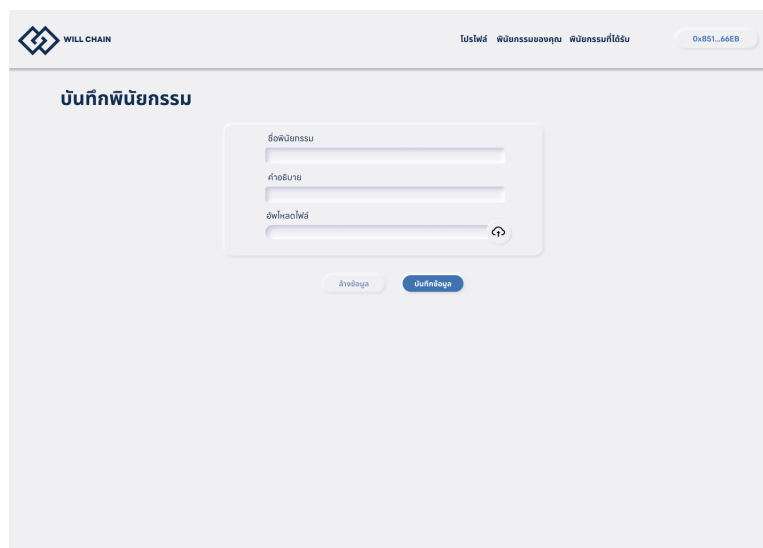
3.4.7 หน้า Approve พินัยกรรมของคุณ



รูปที่ 3.24 หน้า Approve พินัยกรรมของคุณ

จากรูปจะเป็นหน้า Approve พินัยกรรม จากการที่กดเมนู "ทำการ Approve" ในหน้าพินัยกรรมของคุณ โดยที่หน้านี้จะแสดงยืนยันการ Approve ของพินัยกรรมให้ระบบของ Will Chain ดูแลเรื่องพินัยกรรมให้

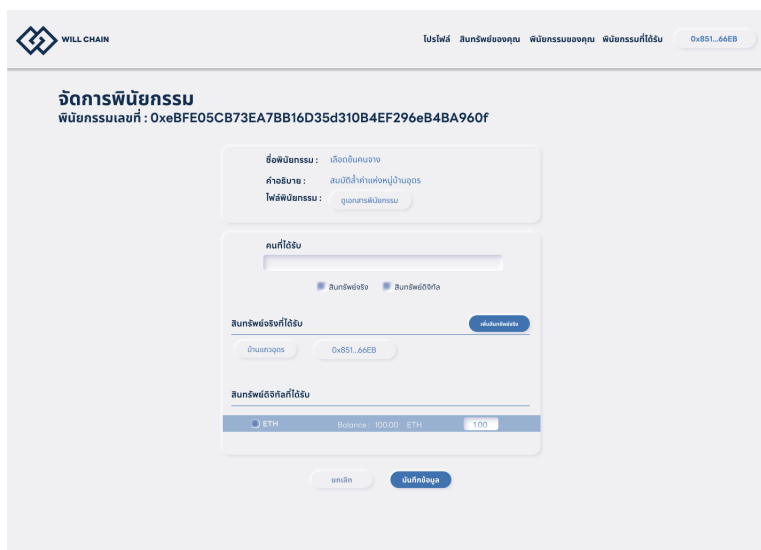
3.4.8 หน้าบันทึกพินัยกรรม



รูปที่ 3.25 หน้าบันทึกพินัยกรรม

จากรูปเป็นหน้าบันทึกพินัยกรรม โดยสามารถบันทึกพินัยกรรมได้ที่หน้านั้น โดยจะเข้าหน้านั้นหลังจากกดที่ปุ่ม “เพิ่มพินัยกรรม” ในหน้าพินัยกรรมของคุณ โดยที่จะมีฟอร์มคือพินัยกรรมที่จะให้ใส่ชื่อพินัยกรรม รายละเอียดของพินัยกรรมฉบับนี้และอัปโหลดไฟล์พินัยกรรมฉบับจริง

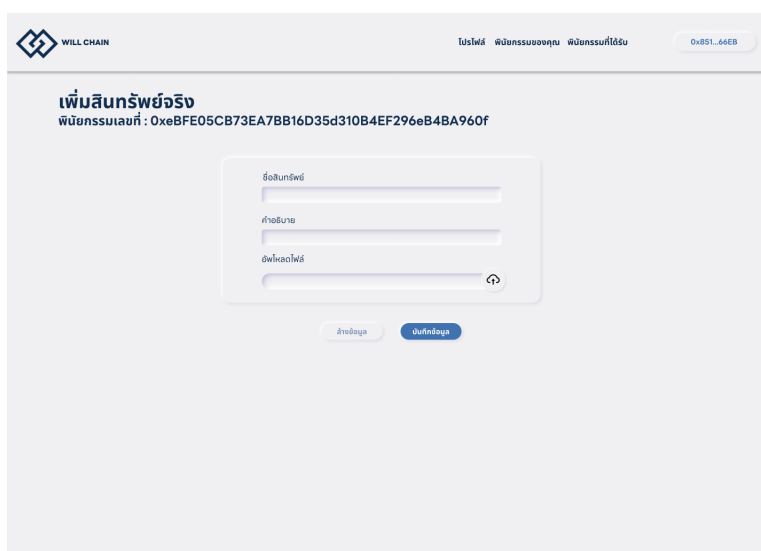
3.4.9 หน้าจัดการสินทรัพย์ภายในพินัยกรรมของคุณ



รูปที่ 3.26 หน้าจัดการสินทรัพย์ภายในพินัยกรรมของคุณ

จากรูปเป็นจะเป็นหน้ายกเลิกการทำพินัยกรรม จากการที่กดเมนู "จัดการพินัยกรรม" ในหน้าพินัยกรรมของคุณ โดยที่หน้านี้จะแสดงการจัดการพินัยกรรมโดยจะแสดงเลขพินัยกรรม , รายละเอียดพินัยกรรม และพินัยกรรมฉบับจริงในรูปแบบไฟล์ โดยหน้านี้นี้จะทำการใส่เลขกระเป๋า metamask ของผู้รับพินัยกรรมได้ , สามารถเพิ่มเหรียญดิจิทัลได้ , และสามารถเพิ่มสินทรัพย์ที่จะทำการ Tokenize ของสินทรัพย์จริงเป็นในรูปของ NFT และแสดงรายละเอียดทรัพย์สินได้


3.4.10 หน้าเพิ่มสินทรัพย์จริง



รูปที่ 3.27 หน้าเพิ่มสินทรัพย์จริง

จากรูปเป็นหน้าเพิ่มสินทรัพย์จริงเข้าสู่พินัยกรรม ที่เมื่อกดที่เมนูสินทรัพย์ในแถบเมนูด้านบน จะเข้าสู่หน้านี้ โดยที่จะแสดงเลขพินัยกรรม และ รายละเอียดของการเพิ่มสินทรัพย์จริงชื่อสินทรัพย์ , คำอธิบาย , และไฟล์สำหรับยืนยันว่าครอบครองพินัยกรรมนั้นจริง ๆ เช่น โฉนดที่ดิน , บ้าน , รถ

3.4.11 หน้าพินัยกรรมที่ได้รับ

WILL CHAIN

โปรไฟล์ | **พินักรบบของทุน** | พินักรบบที่ได้รับ

OxBS1..66EB

พินักรบบที่ได้รับ

พินักรบบเลขที่ : OxwBF05CB73EA7B816D35d310B4EF296e84BA960f

รับรางวัลใหม่

ชื่อพินักรบบ :	เมื่อถึงสิ้นงวด
ค่าธรรมเนียมเพิ่มเติม :	สนับสนุนค่าใช้จ่ายต่อกลุ่มบ้านจุดOS
ไฟล์คำพินักรบบ :	ดูเอกสารคำพินักรบบ
ผู้ทำพินักรบบ :	OxBS1..66EB
ผู้รับพินักรบบ :	OxBS1..66EB
สินทรัพย์ต่อรางวัลที่ได้รับ :	จำนวนรางวัล สมมติ
สินทรัพย์ต่อสิทธิที่ได้รับ :	BTC 1.00 ETH 2.00

รูปที่ 3.28 พิณัยกรรมที่ได้รับ

จากรูปเป็นหน้าพินัยกรรมที่ได้รับ ที่หลังจากมีการส่งต่อพินัยกรรมเนื่องมาจากการเสียชีวิต ในหน้านี้จะแสดงพินัยกรรมที่ได้รับ โดยที่สามารถกดปุ่ม “ดูพินัยกรรม” ที่จะสามารถดูพินัยกรรมที่เป็นฉบับจริงได้ และในตารางจะมีแสดงรายละเอียดของคนที่ได้รับ สิ้น-
ทรัพย์ที่ได้รับ และสินทรัพย์ดิจิทัลที่ได้รับ และสามารถกดรับสินทรัพย์ทั้งหมดได้

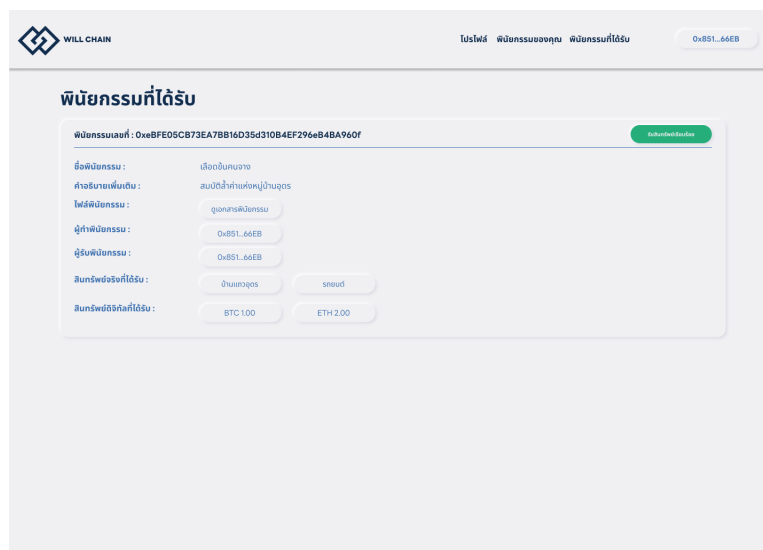
3.4.12 หน้่ายื่นยันรับพินัยกรรม

The screenshot shows the Will Chain website interface. At the top, there is a logo and the text "WILL CHAIN". On the right, it says "โปรไฟล์ | พินัยกรรมของคุณ | พินัยกรรมที่ได้รับ" and a button "0x851...66EB". The main heading is "พินัยกรรมที่ได้รับ". Below this, a table lists transaction details with columns for "พินัยกรรมเลขที่" and "ข้อมูลการโอน". A modal window is open in the center, displaying the text "ยืนยันการรับสินทรัพย์ที่คุณได้จาก 0x8515B0c0c0c0c846c0c0c0c1c69Ccd66EB หรือไม่?" with "ใช่แล้ว" and "ไม่ใช่" buttons.

รูปที่ 3.29 หน้ายื่นรับพินัยกรรม

จากรูปเป็นหน้ายื่นยื่นการรับพินัยกรรมโดยจะทำการเลือกสินทรัพย์ที่ได้รับในแต่ละสินทรัพย์ที่อยู่ในพินัยกรรม

3.4.13 หน้ารับพินัยกรรมเสร็จสิ้น



รูปที่ 3.30 หน้ารับพินัยกรรมเสร็จสิ้น

จากรูปเป็นหน้ารับพินัยกรรมเสร็จสิ้น โดยจะเป็นหน้าที่รับสินทรัพย์และรับพินัยกรรมเสร็จสิ้น

3.5 ออกแบบการทดสอบ

ทดสอบด้วยการจำลองการใช้งานผ่าน platform โดยมี function ที่จะทดสอบดังนี้

3.5.1 Function connect MetaMask wallet สำหรับการเชื่อมต่อ wallet ของผู้ใช้งานเข้ากับ MetaMask เพื่อเตรียมพร้อมต่อการทดสอบ function อื่น ๆ

3.5.2 Function เกี่ยวกับการจัดการพินัยกรรมรวมถึงการเพิ่มผู้รับผลประโยชน์และสินทรัพย์

3.5.3 Function เกี่ยวกับการจัดการสินทรัพย์ที่ผู้ใช้ทำการลงทะเบียนไว้ในระบบ

3.5.4 Function การส่งพินัยกรรมและสินทรัพย์ไปให้ผู้รับมรดก

บทที่ 4 ผลการดำเนินงาน

4.1 Site map

4.1.1 หน้าหลัก

4.1.2 หน้าโปรไฟล์

- ลงทะเบียนด้วยรหัสเลขบัตรประชาชน

4.1.3 หน้าพินัยกรรมที่บันทึกของคุณ

- ทำการ Approve พินัยกรรมของคุณ
- ยกเลิกการทำพินัยกรรม

4.1.4 หน้าบันทึกพินัยกรรม

- อัปเดตพินัยกรรม

4.1.5 หน้าจัดการพินัยกรรม

- เพิ่มสินทรัพย์
- เพิ่มผู้รับพินัยกรรม

4.1.6 พินัยกรรมที่ได้

- รับสินทรัพย์

4.2 Token ที่ใช้ใน Will-Chain

4.2.1 Will Token

Will Token คือ NFT ที่ทำหน้าที่แทนพินัยกรรมซึ่งทำงานบนระบบ Ethereum โดยใช้มาตรฐาน ERC-721 ซึ่ง Will Token จะเป็นตัวที่ต้องใช้ในการที่เราจะสร้างรับสินทรัพย์ที่มีอยู่ในพินัยกรรมของ Will Contract

4.2.2 ETH

ETH คือ เหรียญดิจิทัลที่ใช้เทคโนโลยีบล็อกเชนทำงานอยู่เบื้องหลัง เพื่อใช้จ่ายสำหรับค่าธรรมเนียม สำหรับทุก Smart Contract ที่ผู้ใช้งานต้องการจะขึ้นบน Ethereum chain

4.2.3 Any Token Support ERC20

Any Token Support ERC20 คือเหรียญที่ใช้มาตรฐาน ERC-20 ที่ซึ่งใช้เพื่อสำหรับฝากเหรียญดิจิทัลเข้าสู่ระบบของเรา

4.2.4 Real Token

Real Token คือ NFT ที่ทำหน้าที่แทนสินทรัพย์จริงซึ่งทำงานบนระบบ Ethereum โดยใช้มาตรฐาน ERC-721

4.3 Test Plan

4.3.1 Validation Testing

ตรวจสอบว่า Software ตรงตาม Requirement หรือไม่
บันทึกข้อผิดพลาดพร้อมกับการแก้ไข

4.3.2 Verification Testing

ตรวจสอบว่า Software ออกแบบได้ตรงตาม UX/UI ที่ออกแบบไว้หรือไม่

ตรวจสอบว่า Software ออกแบบได้ตรงตาม Architecture หรือไม่

ตรวจสอบว่า Software ออกแบบได้ตรงตาม UML design หรือไม่

4.4 Software Testing

System Testing จะใช้การทำ Unit Testing โดยทำการแยกการทดสอบเป็น function ต่าง ๆ ดังนี้

4.4.1 Test Case

Page	Test case	Result
Profile	ใส่รหัสบัตรประชาชนถูกต้อง	
	ใส่รหัสบัตรประชาชนที่ไม่มีในระบบ	
	ใส่รหัสบัตรประชาชนไม่ครบ/เกิน	
	เลือก menu สินทรัพย์	
	เลือก menu พินัยกรรม	
	เลือก menu พินัยกรรมที่ได้รับ	
สินทรัพย์ของคุณ	เลือกเพิ่มสินทรัพย์	
	เลือกลบสินทรัพย์	
	เลือกดูข้อมูลสินทรัพย์	
บันทึกสินทรัพย์	กรอกข้อมูลปกติแล้วกดบันทึก	
	ไม่กรอกชื่อสินทรัพย์	
	ไม่เลือกข้อมูลเพิ่มเติม	
	ไม่อัปเดตไฟล์สินทรัพย์	
	เพิ่มจำนวนสินทรัพย์	
	กรอกข้อมูลสินทรัพย์ไม่ครบทุกหน้า	
รายชื่อคนรับมรดก	เลือกเพิ่มรายชื่อ	
	เลือกลบรายชื่อ	
พินัยกรรมของคุณ	เลือกเพิ่มพินัยกรรม	
	เลือกลบพินัยกรรม	
	เลือกดูข้อมูลพินัยกรรม	
บันทึกพินัยกรรม	กรอกข้อมูลปกติแล้วกดบันทึก	
	ไม่กรอกชื่อพินัยกรรม	
	ไม่อัปเดตไฟล์พินัยกรรม	
	ไม่เลือกผู้ได้รับมรดก	
	ไม่เลือกชนิดของสินทรัพย์ที่ได้รับ	
	เลือกชนิดของสินทรัพย์ไม่ตรงกับสินทรัพย์ที่ได้รับ	
	มอบสินทรัพย์มากกว่าที่มีอยู่	
หน้าพินัยกรรมที่ได้รับ	เลือกดูพินัยกรรม	
	กดรับสินทรัพย์ก่อนเวลา	
	กดรับสินทรัพย์เมื่อถึงเวลา	

หลังจากผ่าน System testing แล้วจะนำไปให้ User ทำการทดสอบว่าตอบสนองความต้องการหรือไม่ โดยจะนำไปให้บุคคลที่มีสินทรัพย์ดิจิทัลที่มีความต้องการที่จะทำพินัยกรรมทดลองใช้ Will Chain ของเรา

- ความง่ายในการใช้งาน
- ความน่าเชื่อถือของ Software
- ความเสถียร ของ Software

4.4.2 Test Smart Contract

4.4.2.1 การทดสอบ Smart Contract ด้วย Hardhat Testing การทดสอบด้วย Hardhat Testing ผู้จัดทำได้ออกแบบทดสอบของ Smart Contract ด้วยแบ่งเป็น Flow ในการเรียกใช้งานในแต่ละ Test Case เพื่อให้ครอบคลุมทุกเงื่อนไขในแต่ละฟังก์ชัน

```

Network Info
=====
> HardhatEVM: v2.14.0
> network: hardhat

Test USDT Contract
✓ function mint (262ms)

Test Will
✓ function getRealTokenAddress (565ms)
✓ function depositRealToken real token should assign to owner
✓ function getNameTokenID
✓ function getDescRealToken

Test WillFactory
✓ function set WillToken address connect to willFactory contract (216ms)
✓ function set RealToken address connect to willFactory contract
✓ Create will should mint will token to msg.sender (83ms)
✓ function registerID
✓ function claim will should beneficiary receive will token from owner (151ms)
✓ test require createWill not set will token address
✓ test require claimWill beneficiary not set value (131ms)
✓ Function delete will delete out of mapping (91ms)

21 passing (2s)

File          % Stmts   % Branch   % Funcs   % Lines   Uncovered Lines
-----
contracts\    92.05     81.82     86.05     92.39
  RealToken.sol 72.73     100      57.14     72.73      35,51,60
  USDT.sol      100      100      100      100
  Will.sol      100      85.71     100      100
  WillFactory.sol 96.3      75      100     96.55      63
  WillToken.sol 70      100      50      70      29,45,54
contracts\interfaces\ 100      100      100      100
  IERC20MetaData.sol 100      100      100      100
  IERC6127.sol 100      100      100      100
-----
All files      92.05     81.82     86.05     92.39

```

รูปที่ 4.1 ผล Test Coverage ของ Smart Contract

จากรูป เป็นการออกแบบ Test Case ให้ครอบคลุมทุกฟังก์ชันโดยจากภาพจะมีค่า Test Coverage โดย

%Stmts เป็นตัวชี้วัดวัดเปอร์เซ็นต์ของคำสั่งรหัสที่ถูกดำเนินการในระหว่างการทดสอบ ระบุจำนวนบรรทัดของโค้ดที่ถูกเรียกใช้ อย่างน้อยหนึ่งครั้ง

%Branch เป็นตัวชี้วัดวัดเปอร์เซ็นต์ของ code ที่ได้รับระหว่างการทดสอบ ตรวจสอบโดยเป็นการตรวจสอบเงื่อนไขของตัว Smart Contract

%Funcs เป็นตัวชี้วัดวัดเปอร์เซ็นต์ของฟังก์ชันที่ถูกเรียกใช้ในระหว่างการทดสอบ จะตรวจสอบว่าแต่ละฟังก์ชันได้รับการดำเนินการ อย่างน้อยหนึ่งครั้งหรือไม่

%Lines เป็นตัวชี้วัดวัดเปอร์เซ็นต์ของความครอบคลุมของบรรทัดของ code โดยจะวัดเปอร์เซ็นต์ของบรรทัดโค้ดที่ดำเนินการระหว่างการทดสอบ

4.5 การ Deploy Smart Contract ไปที่ Ethereum Chain

การ Deploy Smart Contract ไปที่ Ethereum Chain โดยเป็น Test network เพื่อความรวดเร็วในการ Deploy Smart Contract และไม่มีค่าธรรมเนียมในการทำธุรกรรมต่าง ๆ ในการทำงานของระบบ โดยใช้ Hardhat ในการทำการ Deploy Smart Contract โดยมีขั้นตอนดังนี้

1. ต้องทำการ Deploy Contract WillFactory ซึ่งเป็น Contract ที่ใช้สำหรับการจัดการพินัยกรรมทั้งหมดและจัดการการสืบทอดพินัยกรรมของพินัยกรรมทั้งหมด
2. ทำการ Deploy Contract Will Token เพื่อที่จะทำการ Tokenize ของพินัยกรรมและเป็น NFT มาตรฐาน ERC-721 โดย Will Token ไม่สามารถถ่ายโอนให้คนอื่นได้ด้วยตัวเอง แต่จะสามารถถ่ายโอนได้ ผ่าน Contract WillFactory เท่านั้น
3. ทำการ Deploy Contract Real Token เพื่อที่จะทำการ Tokenize ตัวสินทรัพย์จริงและสินทรัพย์นี้จะทำการสร้าง NFT มาตรฐาน ERC-721 โดย Real Token จะไม่สามารถถ่ายโอนให้คนอื่นได้ด้วยตัวเอง แต่จะสามารถถ่ายโอนได้ ผ่าน Contract Will เท่านั้น
4. ทำการ Deploy Contract USDT เพื่อที่จะทำการจำลองสินทรัพย์ดิจิทัลประเภทเหรียญในระบบ โดยจะเป็นมาตรฐาน ERC-20 ใช้สำหรับทดสอบการฝากด้วย Token ที่นอกเหนือจาก ETH
5. ทำการ Deploy Contract Will เป็นการ Contract ที่สร้างขึ้นมาจาก Will Factory โดย Will Contract จะทำหน้าที่คอยเก็บพินัยกรรมของเจ้าของพินัยกรรมและจัดการพินัยกรรมสินทรัพย์ในระบบทั้งหมด

ผลลัพธ์ของการ Deploy Smart Contract จะได้ Contract Address ดังนี้

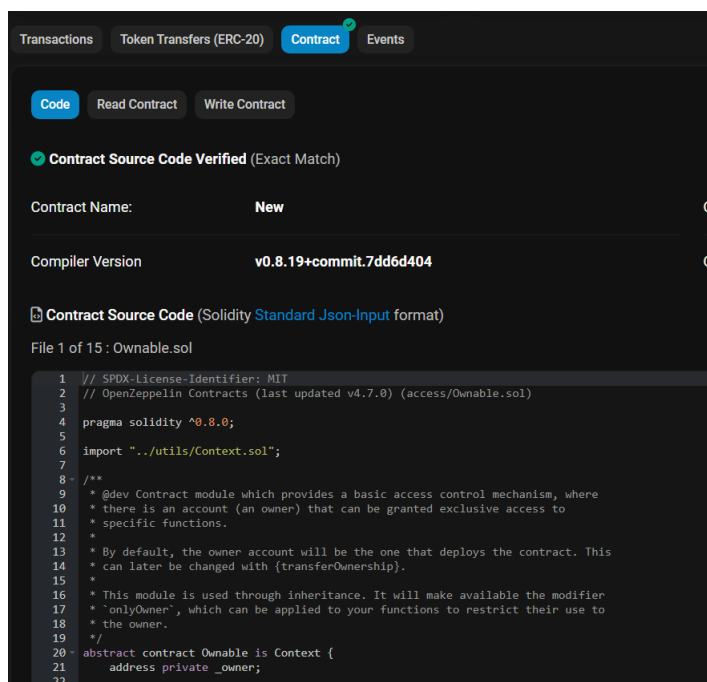
willFactory deployed to: 0x54BeBcd2469AAE5E4417f4c6d01d2C8Eb31331cC

willToken deployed to: 0x9f2c24a6aB735C9ECFbF2e8d1A36CD8F54D2248F

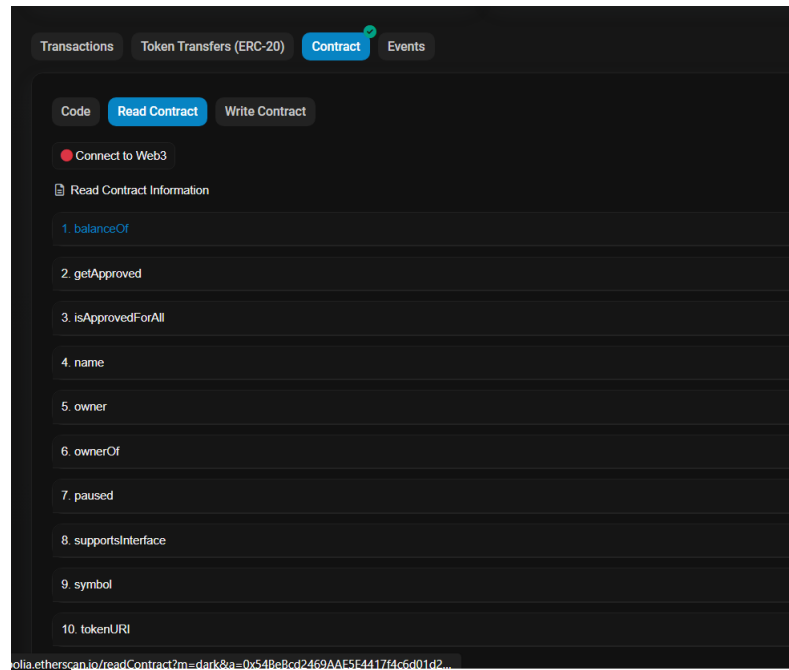
USDt deployed to: 0xe25ddff621198069bA7fe5A18f3D94C1f6F60496

RealToken deployed to: 0x7d7Ce3e5Be7Be44FC38F9A73046D2d79735d552f

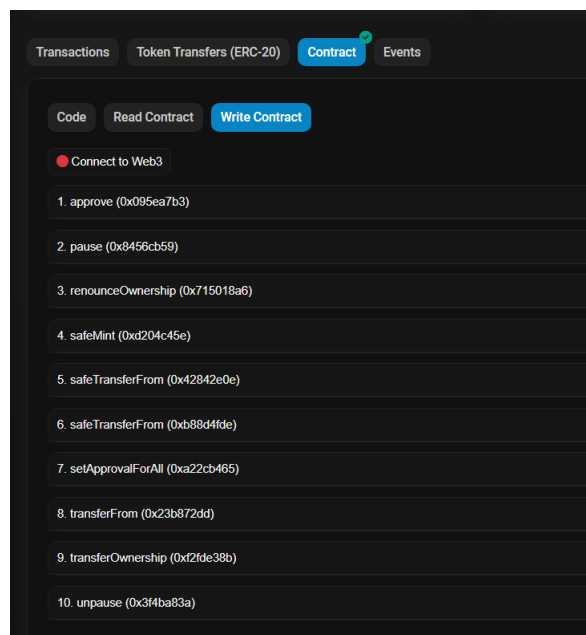
โดย Will Contract จะถูกสร้างด้วย WillFactory และ Smart Contract ทั้งหมดจะถูก verify เพื่อความถูกต้องและความปลอดภัยด้วย Hardhat verify โดย Smart Contract ที่ verify แล้วจะแสดงผลได้ที่หน้า Etherscan ที่เป็น block explorer และสามารถไต่ฟังก์ชันใน Contract ได้ผ่าน หัวข้อ Contract



รูปที่ 4.2 แสดงผล code หลัง verified contract



รูปที่ 4.3 แสดงฟังก์ชันสำหรับที่แสดงผล หลัง verified contract



รูปที่ 4.4 แสดงฟังก์ชันที่มีการเขียนลงระบบ ethereum chain หลัง verified contract

บทที่ 5 สรุปผล

5.1 สรุปปัญหาที่พบและวิธีการแก้ไข

5.1.1 Design System

ปัญหาที่พบ 1 : Design ที่ทำมาในช่วงแรกไม่สามารถ implement ได้เนื่องจากขาดความรู้ความเข้าใจในตัว Tech Stack ของระบบทำให้มีการปรับแก้ไขในส่วน Design System ใหม่ และ ส่งผลให้ทุกส่วนในด้านการทำงานของระบบต้องมีการ design ใหม่ การแก้ไข : คือทำการ Design ระบบใหม่ระหว่างช่วงที่ทำการ Develop ไปพร้อม ๆ กัน

5.1.2 Process Management

ปัญหาที่พบ 1 : จากเหตุผลของ Design System ส่งผลให้กระทบการวางแผนและประเมินเวลาในแต่ละวีค การแก้ไข : เพิ่ม workload ในแต่ละวีคและเพิ่ม goal ในแต่ละวีคว่าอะไรต้องเสร็จก่อนเป็นลำดับ

5.1.3 Blockchain

ปัญหาที่พบ 1 : ไม่มี Token ของสินทรัพย์ดิจิทัลให้ทดสอบกับตัว Contract การแก้ไข : เลือกใช้วิธีการสร้าง Contract USDT ซึ่งเป็นเหรียญที่ใช้ interface ตัวเดียวกันกับของที่มีอยู่ใน blockchain

5.1.4 Smart Contract

ปัญหาที่พบ 1 : ไม่สามารถทำการส่งพินัยกรรมให้ผู้รับพินัยกรรมที่มีในระบบมากกว่าสองคนได้ใน พินัยกรรมเดียว การแก้ไข : ทำการสร้างพินัยกรรมที่เป็น Will Contract สำหรับจัดการ 1 พินัยกรรมต่อ 1 คน และถ้ามีหลายคน ก็ใช้พินัยกรรมเดิมในสร้าง Will Contract ขึ้นมาใหม่

ปัญหาที่พบ 2 : มีปัญหาเรื่องของการกำหนดเงื่อนไขว่าใครสามารถทำอะไรบ้างในแต่ละคอนแทค การแก้ไข : ทำการใช้ interface AccesControl เป็นตัวช่วยในการจัดการ Role และ เพิ่ม role ของ smart contract

ปัญหาที่พบ 3 : มีปัญหาเรื่องการการ estimate gas ไม่ได้ ตอนที่ทำการ Transfer จาก Will Contract ไปที่กระเป๋าของผู้รับพินัยกรรม การแก้ไข : ทำการ approve token ของเจ้าของพินัยกรรมนั้นก่อนทำการส่งไปที่ Contract

5.1.5 Frontend

ปัญหาที่พบ 1 : ทำการเชื่อมต่อ metamask ได้ยาก การแก้ไข : ทำการใช้ตัว thirdparty ในการเชื่อมต่อกับ metamask wallet

หนังสืออ้างอิง

1. International Business Machines Corporation, "What is Blockchain Technology? ," Available at www.ibm.com/topics/what-is-blockchain, [Online; accessed 5-September-2022].
2. Ethereum, 2022, "ERC-20 TOKEN STANDARD," Available at ethereum.org/en/developers/docs/standards/tokens/erc-20/, [Online; accessed 5-September-2022].
3. Ethereum, 2022, "ERC-721 TOKEN STANDARD," Available at ethereum.org/th/developers/docs/standards/tokens/erc-721/, [Online; accessed 5-September-2022].
4. SCB10X, 2022, "Tokenization ในโลก Blockchain คืออะไร?," Available at scb10x.com/blog/tokenization-blockchain, [Online; accessed 6-September-2022].
5. Ethereum, 2022, "What is Ethereum?," Available at ethereum.org/en/what-is-ethereum/, [Online; accessed 5-September-2022].
6. Thanyavuth Akarasomcheep, 2018, "Scrum คืออะไร เริ่มใช้งานอย่างไร?," Available at medium.com/fastwork-engineering/scrum-คืออะไร-เริ่มใช้งานอย่างไร-2483e761a47e, [Online; accessed 5-September-2022].
7. Kulawat Pom Wongsaroj, 2019, "จะเลือก Scrum หรือ Kanban ดี?," Available at kulawat.medium.com/จะเลือก-scrum-หรือ-kanban-ดี-f7c0743f8a45, [Online; accessed 5-September-2022].
8. International Business Machines Corporation, "What are smart contracts on blockchain?," Available at www.ibm.com/topics/smart-contracts, [Online; accessed 5-September-2022].
9. RAKESH SHARMA, 2022, "Non-Fungible Token (NFT): What It Means and How It Works," Available at www.investopedia.com/non-fungible-tokens-nft-5115211, [Online; accessed 5-September-2022].
10. ADAM BARONE, 2022, "What Is an Asset? Definition, Types, and Examples," Available at www.investopedia.com/terms/a/asset.asp, [Online; accessed 5-September-2022].
11. Siam Commercial Bank, "What are digital assets?," Available at www.scb.co.th/en/personal-banking/stories/digital-asset.html, [Online; accessed 5-September-2022].
12. Siam Commercial Bank, "พินัยกรรม ทำอย่างไร?," Available at www.scb.co.th/th/personal-banking/stories/tips-for-you/testament.html, [Online; accessed 5-September-2022].
13. RudreshVeerkhare, "CryptoWill," Available at github.com/RudreshVeerkhare/CryptoWill, [Online; accessed 29-October-2022].
14. GitHub, "GitHub," Available at github.com, [Online; accessed 5-September-2022].
15. MetaMask, "MetaMask," Available at <https://metamask.io/>, [Online; accessed 5-September-2022].
16. NestJS, "NestJS," Available at docs.nestjs.com, [Online; accessed 5-September-2022].
17. NextJS, "Next.js by Vercel - The React Framework," Available at nextjs.org/docs, [Online; accessed 5-September-2022].
18. Solidity, "Solidity," Available at docs.soliditylang.org, [Online; accessed 5-September-2022].
19. TypeScript, "TypeScript," Available at www.typescriptlang.org, [Online; accessed 5-September-2022].
20. Web3.js Developer Team, "web3.js - Ethereum JavaScript API," Available at web3js.readthedocs.io/en/v1.8.1/, [Online; accessed 5-September-2022].
21. Hardhat, "Documentation - Hardhat," Available at hardhat.org/docs, [Online; accessed 5-September-2022].
22. Pinata, "Documentation - Pinata," Available at docs.pinata.cloud, [Online; accessed 5-September-2022].
23. InterPlanetary File System, "Documentation - IPFS," Available at docs.ipfs.tech, [Online; accessed 5-September-2022].