



ศึกษาเชิงเปรียบเทียบการสร้างและเผยแพร่สื่อการสอนวิชาสถาปัตยกรรม  
คอมพิวเตอร์ ระหว่างการใช้เทคโนโลยีแบบรวมศูนย์ (แบบเดิม) กับ  
เทคโนโลยีแบบกระจาย (แบบใหม่)

โดย

นางสาวสุวิษฐา บุญรัตน์พันธุ์

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธรรมศาสตร์

ปีการศึกษา 2564

ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

ศึกษาเชิงเปรียบเทียบการสร้างและเผยแพร่สื่อการสอนวิชา  
สถาปัตยกรรมคอมพิวเตอร์ ระหว่างการใช้เทคโนโลยีแบบรวมศูนย์  
(แบบเดิม) กับ เทคโนโลยีแบบกระจาย (แบบใหม่)

โดย

นางสาวสุวิษฐา บุญยรัตพันธุ์

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
วิศวกรรมศาสตรบัณฑิต  
สาขาวิชาวิศวกรรมไฟฟ้า หรือ สาขาวิชาวิศวกรรมคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธรรมศาสตร์  
ปีการศึกษา 2564  
ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

TITLE

By

Miss Suwitchaya Bunyarattaphun

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE OF BACHELOR OF ENGINEERING  
IN ELECTRICAL ENGINEERING / COMPUTER ENGINEERING  
FACULTY OF ENGINEERING  
THAMMASAT UNIVERSITY  
ACADEMIC YEAR 2021  
COPYRIGHT OF THAMMASAT UNIVERSITY

มหาวิทยาลัยธรรมศาสตร์

คณะวิศวกรรมศาสตร์

โครงการ

ของ

นางสาวสุวิษฐา บุญยรัตพันธุ์

เรื่อง

ชื่อเรื่องโครงการ

ได้รับการตรวจสอบและ อนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิศวกรรมศาสตรบัณฑิต

เมื่อวันที่ วันที่ ชื่อเต็มของเดือน พ.ศ. 2565

อาจารย์ที่ปรึกษาโครงการ

---

(รองศาสตราจารย์/ผู้ช่วยศาสตราจารย์/อาจารย์ .....)

หัวหน้าภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

---

(รองศาสตราจารย์ ดร. พิชัย อารีย์)

|                         |  |
|-------------------------|--|
| หัวข้อโครงการ           | ศึกษาเชิงเปรียบเทียบการสร้างและเผยแพร่สื่อการสอนวิชา<br>สถาปัตยกรรมคอมพิวเตอร์ ระหว่างการใช้เทคโนโลยีแบบรวมศูนย์<br>(แบบเดิม) กับ เทคโนโลยีแบบกระจาย (แบบใหม่) |
| ชื่อผู้เขียน            | นางสาวสุวิชญา บุญรัตน์พันธุ์   |
| ชื่อปริญญา              | วิศวกรรมศาสตรบัณฑิต  |
| สาขาวิชา/คณะ            | วิศวกรรมไฟฟ้าและคอมพิวเตอร์<br>คณะวิศวกรรมศาสตร์<br>มหาวิทยาลัยธรรมศาสตร์  |
| อาจารย์ที่ปรึกษาโครงการ | อาจารย์ ดร.ชุมพล บุญมี   |
| ปีการศึกษา              | 2564   |

### บทคัดย่อ

บทคัดย่อภาษาไทยกำหนดให้เป็นฟอนต์ TH Sarabun New ขนาด 14

คำสำคัญ : คำสำคัญ1, 2, 3

|               |                                    |
|---------------|------------------------------------|
| Title         | ชื่อโครงงาน TH Sarabun New ขนาด 16 |
| Author        | Miss Suwitchaya Bunyarattaphun     |
| Degree        | วิศวกรรมศาสตรบัณฑิต                |
| Major         | วิศวกรรมXXXXXX                     |
|               | Faculty of Engineering             |
|               | Thammasat University               |
| Advisor       | Dr. Choompol Boonmee               |
| Academic Year | 2021                               |

### Abstract

Thesis Abstract font TH Sarabun New size 14

Keywords: Keyword1, 2, 3

## กิตติกรรมประกาศ

ขอขอบคุณ

## สารบัญ

|  | หน้า |
|--|------|
| บทคัดย่อ                                       | (1)  |
| Abstract                                       | (2)  |
| กิตติกรรมประกาศ                                | (3)  |
| สารบัญ   | (4)  |
| สารบัญรูป                                      | (6)  |
| สารบัญตาราง                                    | (7)  |
| สัญลักษณ์และคำย่อ                              | (8)  |
| บทที่ 1 บทนำ                                   | 1    |
| 1.1 ที่มาและความสำคัญ                          | 1    |
| 1.2 วัตถุประสงค์                               | 1    |
| 1.3 ขอบเขตการดำเนินงาน                         | 1    |
| 1.4 ขั้นตอนการดำเนินงาน                        | 2    |
| 1.5 ผลที่คาดว่าจะได้รับ                        | 2    |
| 1.6 ตารางการดำเนินงาน                          | 3    |
| บทที่ 2 ทฤษฎี หรืองานที่เกี่ยวข้อง             | 4    |
| 2.1 รายละเอียดวิชาสถาปัตยกรรมคอมพิวเตอร์       | 4    |
| 2.1.1 Computer Abstractions and Technology     | 4    |
| 2.1.2 Instructions: Language of the Computer   | 4    |
| 2.1.3 Arithmetic for Computers                 | 4    |
| 2.1.4 Arithmetic for Computers                 | 5    |
| 2.1.5 and Fast: Exploiting Memory Hierarchy    | 5    |
| 2.1.6 Parallel Processors from Client to Cloud | 5    |
| 2.2 เทคโนโลยีระบบอินเทอร์เน็ตแบบกระจาย IPFS    | 5    |



|  |    |
|--|----|
| 2.2.1 IPSF คืออะไร   | 5  |
| 2.2.2 CID คืออะไร  | 6  |
| 2.2.3 การจัดการไฟล์บน IPFS                                 | 7  |
| 2.2.4 หลักการพื้นฐานในการทำความเข้าใจ IPFS                 | 9  |
| 2.2.5 ข้อดีของ IPSF  | 9  |
| 2.2.6 การเข้าร่วม IPFS                                     | 10 |
| 2.3 เครื่องมือที่ใช้ทำสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์ | 10 |
| 2.3.1 เบลนเดอร์  | 10 |
| 2.3.2 ภาษาที่ใช้ในการทำ Website                            | 11 |
| 2.3.3 Visual Studio Code                                   | 12 |
| 2.3.3 เครื่องมืออื่นๆ                                      | 12 |
| บทที่ 3 การดำเนินงาน/วิจัย                                 | 14 |
| 3.1 การวางแผนการดำเนินงาน                                  | 14 |
| 3.1.1 กลุ่มเป้าหมาย  | 14 |
| 3.1.2 แบบแผนการดำเนินงาน                                   | 15 |
| บรรณานุกรม   | 20 |

## สารบัญรูป

| รูปที่                                   | หน้า |
|--|------|
| 1. unique fingerprint                    | 6    |
| 2. unique fingerprint                    | 7    |
| 3. look up file                          | 7    |
| 4. pin content                           | 7    |
| 5. file's version                        | 7    |
| 6. IPNS                                  | 8    |
| 7. Homepage                              | 16   |
| 8. Homepage                              | 16   |
| 9. Homepage                              | 16   |
| 10. หัวข้อ Introduce                     | 17   |
| 11. หัวข้อ Introduce                     | 17   |
| 12. หัวข้อ What is Computer Architecture | 18   |
| 13. หัวข้อ What is Computer Architecture | 18   |

## สารบัญตาราง

หน้า

ตารางที่ 1 ตารางดำเนินงาน

3

## สัญลักษณ์และคำย่อ

สัญลักษณ์/คำย่อ

คำเต็ม/คำจำกัดความ

EoS

Equation of State

## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญ

ในยุคปัจจุบัน โลกของเราพัฒนาอย่างก้าวกระโดด เพื่อตอบสนองความต้องการความสะดวกสบายในชีวิตประจำวันของมนุษย์ เนื่องจากค่านิยมสมัยใหม่ที่เน้นความสะดวกหรือร้อน และพัฒนาความสามารถได้ด้วยตนเอง ทางผู้จัดทำจึงเล็งเห็นว่าสื่อการสอนที่เข้าถึงได้ง่าย สามารถทำให้ผู้เรียนสืบค้น และฝึกฝนการคิด วิเคราะห์ สังเคราะห์ได้ด้วยตัวเอง ทุกที่ทุกเวลานั้น เป็นสิ่งที่ตอบโจทย์ความต้องการเป็นอย่างมาก

เพื่อบูรณาการกับสิ่งที่เคยได้เรียนรู้มา ผู้จัดทำเล็งเห็นว่าวิชาสถาปัตยกรรมคอมพิวเตอร์ เป็นวิชาบังคับของคณะวิศวกรรมศาสตร์ สาขาคอมพิวเตอร์ ที่นักศึกษาทุกคนจะต้องได้เรียน และยังต้องนำองค์ความรู้ในวิชานี้ไปต่อยอดในวิชาในระดับสูงต่อไป จึงมีความสนใจที่จะทำสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์ หวังเพื่อเป็นประโยชน์แก่ตนเอง และผู้เรียนท่านอื่นในอนาคต ไม่นานก็น้อย

โดยมีความตั้งใจว่าผู้ใช้งานจะต้องเข้าถึงได้ง่าย สะดวก และรวดเร็ว ผ่านระบบอินเทอร์เน็ต ซึ่งระบบที่ใช้ในปัจจุบันเป็นแบบรวมศูนย์ มีข้อเสียคือรวมฐานข้อมูลไว้ในที่เดียว และให้ผู้ใช้งานทุกคนติดต่อขอข้อมูลที่เครื่อง Server เครื่องเดียว อาจทำให้ระบบล่มได้ ดังนั้น ผู้จัดทำจึงต้องการเปรียบเทียบการใช้งานระบบอินเทอร์เน็ตแบบเดิม กับระบบอินเทอร์เน็ตแบบใหม่ เป็นแบบกระจาย อาศัยเครื่องคอมพิวเตอร์พันธมิตรเป็น Server เก็บข้อมูล และเปรียบเทียบว่าระบบแบบใดสามารถตอบโจทย์ความต้องการได้ดีกว่ากัน มากน้อยเพียงใด

#### 1.2 วัตถุประสงค์

1. สามารถสร้างสื่อที่ใช้ศึกษา และทบทวนความรู้วิชาสถาปัตยกรรมคอมพิวเตอร์ และเผยแพร่ผ่านระบบอินเทอร์เน็ตได้
2. เปรียบเทียบการเข้าถึงข้อมูลสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์ ระหว่างเทคโนโลยีแบบรวมศูนย์และแบบกระจาย

#### 1.3 ขอบเขตการดำเนินงาน

1. วางแผนการดำเนินงาน
2. ศึกษาข้อมูลของวิชาสถาปัตยกรรมคอมพิวเตอร์ , IPFS และเครื่องมือที่จะใช้ทำสื่อการสอน
3. ออกแบบการสร้างสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์ และการเปรียบเทียบเทคโนโลยีทั้งสองแบบ
4. สร้างสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์
5. ส่งออกสื่อการสอนไปยังระบบอินเทอร์เน็ตทั้งสองแบบ คือ เทคโนโลยีแบบรวมศูนย์ และแบบกระจาย
6. เปรียบเทียบการเข้าถึงสื่อการสอน ระหว่างการใช้เทคโนโลยีแบบรวมศูนย์ และแบบกระจาย

## 7. สรุปและบันทึกผลการเปรียบเทียบ

### 1.4 ขั้นตอนการดำเนินงาน

1. วางแผนการดำเนินการ
2. รวบรวมข้อมูลที่ต้องใช้ในการศึกษา
3. ศึกษาการใช้งานเทคโนโลยีแบบรวมศูนย์ และแบบกระจาย
4. ศึกษาข้อมูลที่น่ามาใช้ทำสื่อในวิชาสถาปัตยกรรมคอมพิวเตอร์
5. ศึกษาเครื่องมือที่ใช้ทำสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์
6. ออกแบบการสร้างสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์
7. จัดทำสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์
8. ออกแบบลักษณะในการเปรียบเทียบ
9. ศึกษาเครื่องมือที่ใช้เปรียบเทียบ
10. ส่งออกสื่อการสอนโดยใช้เทคโนโลยีทั้งสองแบบ
11. เปรียบเทียบการเข้าถึงข้อมูลสื่อการสอนระหว่างเทคโนโลยีทั้งสองแบบ
12. สรุปผลการเปรียบเทียบที่ได้

### 1.5 ผลที่คาดว่าจะได้รับ

1. สามารถเข้าถึงสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์ได้ ผ่านระบบอินเทอร์เน็ต
2. สามารถเลือกใช้เทคโนโลยีแบบรวมศูนย์ และแบบกระจาย ได้อย่างเหมาะสม

## 1.6 ตารางการดำเนินงาน

| ระยะเวลา<br>ดำเนินงาน   | สิงหาคม | กันยายน | ตุลาคม | พฤศจิกายน | ธันวาคม | มกราคม | กุมภาพันธ์ | มีนาคม |
|---|---------|---------|--------|-----------|---------|--------|------------|--------|
| วางแผนการ<br>ดำเนินการ  | /       |         |        |           |         |        |            |        |
| รวบรวมข้อมูล<br>จำเป็นต้องใช้<br>ในการศึกษา                                     | /       | /       |        |           |         |        |            |        |
| ศึกษาข้อมูลวิชา<br>สถาปัตยกรรม<br>คอมพิวเตอร์                                   | /       | /       |        |           |         |        |            |        |
| ศึกษาเครื่องมือ<br>ที่ใช้ทำสื่อการ<br>สอน                                       | /       | /       |        |           |         |        |            |        |
| ศึกษาการใช้งาน<br>เทคโนโลยีแบบ<br>รวมศูนย์ และ<br>แบบกระจาย                     | /       | /       |        |           |         |        |            |        |
| ออกแบบการ<br>สร้างสื่อการสอน  |         | /       | /      |           |         |        |            |        |
| จัดทำสื่อการ<br>สอน   |         |         | /      | /         | /       | /      | /          |        |
| วางแผนการ<br>เปรียบเทียบ<br>ระหว่าง<br>เทคโนโลยีทั้ง<br>สองแบบ                  |         |         |        |           |         | /      | /          |        |
| ศึกษาเครื่องมือ<br>ที่ใช้เปรียบเทียบ  |         |         |        |           |         | /      | /          |        |
| ส่งออกสื่อการ<br>สอนโดยใช้<br>เทคโนโลยีทั้ง<br>สองแบบ                           |         |         |        |           |         | /      | /          |        |
| เปรียบเทียบการ<br>เข้าถึงข้อมูลสื่อ<br>การสอนระหว่าง<br>เทคโนโลยีทั้ง<br>สองแบบ |         |         |        |           |         | /      | /          |        |
| สรุปผลการ<br>เปรียบเทียบที่ได้  |         |         |        |           |         | /      | /          | /      |

## บทที่ 2

### วัตถุประสงค์

#### 2.1 รายละเอียดวิชาสถาปัตยกรรมคอมพิวเตอร์

RISC-V เป็นสถาปัตยกรรมชุดคำสั่งที่ขยายได้ (ISA) มีแนวโน้มว่าจะเป็นที่ยอมรับอย่างมากในอนาคต ดังนั้นจึงเป็นการดีที่จะนำมามาศึกษาเพื่อพร้อมรับมือกับการทำงานภายใต้ RISC-V

โดยต่อจากนี้จะเป็นรายละเอียดของเนื้อหาวิชาสถาปัตยกรรมคอมพิวเตอร์เพียงคร่าวๆ ที่จะนำมาใช้ทำสื่อการสอน โดยอ้างอิงมาจาก Text Book Computer Organization and Design: The Hardware/Software Interface เขียนโดย David A. Patterson และ John L. Hennessy

โดยจะมีด้วยกัน 6 บท ดังต่อไปนี้

##### 2.1.1 Computer Abstractions and Technology

อธิบายให้เข้าใจเกี่ยวกับประสิทธิภาพของคอมพิวเตอร์นั้นพิจารณาจากปัจจัยหลักสามประการ ได้แก่ การนับคำสั่ง เวลารอบสัญญาณนาฬิกา และรอบสัญญาณนาฬิกาต่อคำสั่ง (CPI) และยังมีเนื้อหาเกี่ยวกับประวัติ ทฤษฎี และการพัฒนาคอมพิวเตอร์ และอุปกรณ์ต่างๆ ตั้งแต่ยุคแรกเริ่มจนมาเป็นคอมพิวเตอร์ในปัจจุบัน มีการสอดแทรกคำศัพท์ที่จำเป็นต้องรู้ สอดแทรกเอาไว้ เพื่อให้พื้นฐานนั้นแน่นขึ้น และนำไปใช้ต่อยอดในอนาคต

##### 2.1.2 Instructions: Language of the Computer

คำในภาษาคอมพิวเตอร์เรียกว่าคำสั่ง (Instructions) และกลุ่มคำจะเรียกว่าชุดคำสั่ง (Instructions set) ในบทนี้จะกล่าวถึงคำสั่งและชุดคำสั่งคอมพิวเตอร์ในรูปแบบต่างๆ ทั้งแบบที่มนุษย์ใช้เขียน และแบบที่คอมพิวเตอร์ใช้อ่าน ชุดคำสั่งที่เลือกคือ RISC-V ซึ่งเดิมพัฒนาขึ้นที่ UC Berkeley เริ่มในปี 2010 และ เปรียบเทียบกับชุดคำสั่งยอดนิยมอีกสองชุดอย่างคือ MIPS และ Intel x86 และกล่าวถึงฮาร์ดแวร์และความสัมพันธ์ระหว่างภาษาโปรแกรมระดับสูงกับภาษาดั้งเดิม

##### 2.1.3 Arithmetic for Computers

เป้าหมายของบทนี้ คือการแสดงให้เห็นถึงตัวเลขที่ยากขึ้น เช่น เศษส่วนและจำนวนจริงอื่นๆ รวมถึงอัลกอริทึมเลขคณิต ฮาร์ดแวร์ที่ติดตามอัลกอริทึมเหล่านี้ และความหมายของทั้งหมดนี้ของชุดคำสั่ง และแสดงวิธีใช้เพื่อทำให้โปรแกรมทางเลขคณิตทำงานเร็วมากยิ่งขึ้น



#### 2.1.4 The Processor

อธิบายการทำงานในหน่วยประมวลผล ในการใช้งาน RISC-V ขั้นพื้นฐาน จะตรวจสอบการใช้งานที่มีชุดย่อย ของชุดคำสั่ง RISC-V เป็นหลัก ไม่ว่าจะเป็น คำสั่งในการอ้างอิง หน่วยความจำ คำสั่งทางคณิตศาสตร์และตรรกศาสตร์ คำสั่งเกี่ยวกับ Branch

#### 2.1.5 Large and Fast: Exploiting Memory Hierarchy

แสดงและอธิบายการเข้าถึงข้อมูล การดึง และการเก็บข้อมูลในหน่วยความจำ โดยเสนอวิธีเลือกและใช้เทคโนโลยี ว่าการทำงานในแบบของเรา เหมาะที่จะใช้หน่วยความจำในรูปแบบใด ถ้าเป็นหน่วยความจำที่เล็ก ก็จะมีราคาแพงกว่าหน่วยความจำที่ใหญ่ เช่น SRAM กับ Magnetic disk

#### 2.1.6 Parallel Processors from Client to Cloud

บทนี้จะอธิบายถึงหน่วยประมวลผลแบบขนาน ความยากในการสร้างโปรแกรม Hardware Multithreading , Multicore และการแชร์หน่วยความจำแบบ Multiprocessors รวมไปถึงการประยุกต์ใช้ด้านการประมวลผลขั้นสูง อย่างการประมวลผลการฝึกอีกด้วย

### 2.2 เทคโนโลยีระบบอินเทอร์เน็ตแบบกระจาย IPFS

#### 2.2.1 IPFS คืออะไร

“IPFS is a distributed system for storing and accessing files, websites, applications, and data.”

ประโยคข้างต้น เป็นประโยคที่ทางเว็บของ IPFS ใช้ในการอธิบายถึง IPFS อย่างสั้นๆ ในประโยคเดียว ซึ่งหมายความว่า IPFS นั้นเป็นระบบแบบกระจาย สำหรับเก็บและเข้าถึงไฟล์, เว็บไซต์, แอปพลิเคชัน และข้อมูล

IPFS หรือ Interplanetary File System เป็นเว็บแบบกระจายตัว แบบ peer-to-peer ที่เชื่อมต่ออุปกรณ์คอมพิวเตอร์แต่ละเครื่อง ซึ่งทำหน้าที่เป็น node ทั้งหมดเข้าสู่ระบบไฟล์เดียวกัน ทำงานคล้ายกับ Blockchain มีการเข้าถึงแตกต่างจาก HTTP โดยจะใช้ค่า Hash เรียกว่า

content identifier (CID) ที่ได้จากข้อมูลของเราในการเข้าถึงข้อมูล สามารถเรียกผ่าน Browser ได้ปกติ

ยกตัวอย่างให้เห็นอย่างชัดเจน ในการเรียกใช้อินเตอร์เน็ตผ่าน HTTP แบบที่ใช้งานกันในปัจจุบัน จะทำการเรียกผ่าน URL บน Browser เช่น <https://en.wikipedia.org/wiki/Aardvark> (ชื่อโปรโตคอลที่ใช้://ชื่อเครื่องคอมพิวเตอร์ และชื่อเครือข่ายย่อย.ประเภทของเว็บไซต์ที่ใช้/ไต่เรียกทอรี่/ชื่อไฟล์และนามสกุล) แต่ระบบ IPFS นั้นจะมีรูปแบบที่แตกต่างออกไป โดยจะทำการเรียกผ่าน [ipfs.io/ipfs/\(CID ของไฟล์ที่ Hash มาแล้ว\)](https://ipfs.io/ipfs/QmXoybizjW3WknFiJnKLwHCnL72vedxjQkDDP1mXWo6uco/wiki/Aardvark.html) เช่น

<https://ipfs.io/ipfs/QmXoybizjW3WknFiJnKLwHCnL72vedxjQkDDP1mXWo6uco/wiki/Aardvark.html> เห็นได้ว่า QmXoybizjW3WknFiJnKLwHCnL72vedxjQkDDP1mXWo6uco คือ CID

## 2.2.2 CID คืออะไร

Content identifier หรือ CID เป็นตัวระบุที่อยู่เนื้อหาที่อธิบายตนเอง ไม่ได้ระบุตำแหน่งที่จัดเก็บเนื้อหา แต่รูปแบบของมันเป็นชนิดของที่อยู่ที่ยื่นอยู่กับเนื้อหา จำนวนอักขระของ CID ขึ้นอยู่กับ cryptographic hash มากกว่าขนาดเนื้อหาของมันเอง เนื่องจากเนื้อหาส่วนใหญ่ใน IPFS ถูก Hash โดยใช้ sha2-256 ทำให้ CID ส่วนใหญ่ที่พบจะมีขนาดเท่ากัน (256 บิตซึ่งเท่ากับ 32 ไบต์) ทำให้ง่ายต่อการจัดการ โดยเฉพาะอย่างยิ่งเมื่อต้องจัดการกับเนื้อหาหลายส่วน

ตัวอย่างเช่น หากเราเก็บรูปภาพของ aardvark บนเครือข่าย IPFS CID ของมันจะมีลักษณะดังนี้ : QmcRD4wkPPi6dig81r5sLj9Zm1gDCL4zgpEj9CfuRrGbzf



(รูปที่ 1: unique fingerprint. หรือ CID ที่ได้จากการ hash ไฟล์ [1] )

ขั้นตอนแรกในการสร้าง CID คือการแปลงข้อมูลอินพุตโดยใช้อัลกอริทึมการเข้ารหัสที่จับคู่อินพุตของขนาดที่กำหนดเอง (ขนาดของข้อมูลหรือไฟล์) กับเอาต์พุตที่มีขนาดคงที่ การแปลงนี้เรียกว่า cryptographic hash digest หรือว่า simply hash.

อัลกอริทึมการเข้ารหัสที่ใช้ต้องสร้าง hash ที่มีลักษณะดังต่อไปนี้:

- Deterministic (กำหนดได้) : อินพุตเดียวกัน ควรสร้าง hash แบบเดียวกันเสมอ
- Uncorrelated (ไม่สัมพันธ์กัน) : การเปลี่ยนแปลงเล็กน้อยในอินพุต ควรสร้าง hash ที่ต่างไปจากเดิมอย่างสิ้นเชิง

- One-way (ทางเดียว) : เมื่อ hash แล้ว ไม่สามารถนำ hash นั้นไปทำย้อนกลับได้
- Unique (ไม่ซ้ำกัน) : ไม่สามารถสร้าง hash เดียวกัน จากไฟล์ที่ต่างกันได้

การ Hash ไม่ได้มีลักษณะเฉพาะเจาะจงสำหรับ IPFS และมีอัลกอริทึมการ Hash จำนวนมากเช่น sha2-256, blake2b, sha3-256 และ sha3-512, sha1 และ md5 ที่ไม่ปลอดภัยอีกต่อไป

### 2.2.3 การจัดการไฟล์บน IPFS



รูปที่ 2: unique fingerprint.  
หรือ CID [2]



รูปที่ 3: look up file.  
การติดต่อ nodes ในระบบ [2]



รูปที่ 4: pin content.  
ปักหมุดเนื้อหาที่สนใจ [2]



รูปที่ 5: file's version.  
ค่า hash จะแตกต่างกัน [2]

เมื่อเพิ่มไฟล์ลงในระบบ IPFS

ไฟล์จะถูกแบ่งออกเป็นชิ้นเล็กๆ ทำการ has แบบเข้ารหัส และได้รับ fingerprint แบบไม่ซ้ำ ที่เรียกว่า content identifier (CID) โดย CID นี้ทำหน้าที่เป็นตัวบ่งชี้การของไฟล์ ตามที่เป็นอยู่ ณ เวลานั้น

เมื่อ node อื่นค้นหาไฟล์นั้นๆ พวกมันจะถาม peer nodes ที่จัดเก็บเนื้อหาที่อ้างอิงโดย CID ของไฟล์ เมื่อพวกมันดูหรือดาวน์โหลดไฟล์ node จะบันทึกสำเนาไฟล์ลงเครื่องของตนเอง และกลายเป็นผู้ให้บริการเนื้อหานั้นๆ กับผู้ใช้รายอื่นจนกว่าบันทึกจะถูกล้าง

node สามารถปักหมุดเนื้อหาเพื่อเก็บไฟล์ไว้ตลอดไป หรือทั้งเนื้อหาที่ไม่ได้ใช้งานมาระยะหนึ่งเพื่อประหยัดพื้นที่ ซึ่งหมายความว่าแต่ละ node ในเครือข่ายจะจัดเก็บเฉพาะเนื้อหาที่สนใจ บวกกับข้อมูลการจัดทำดัชนีบางอย่างที่ช่วยค้นหาว่า node ไດจัดเก็บสิ่งใด

หากเพิ่มไฟล์ version ใหม่ลงใน IPFS hash จะแตกต่างกัน ดังนั้นจึงได้รับ CID ใหม่ ซึ่งหมายความว่าไฟล์ที่จัดเก็บบน IPFS นั้นทนทานต่อการปลอมแปลงและการเซ็นเซอร์ การเปลี่ยนแปลงใดๆ ในไฟล์จะไม่เขียนทับไฟล์ต้นฉบับ และสามารถนำชิ้นส่วนทั่วไปในไฟล์กลับมาใช้ใหม่ได้ เพื่อลดค่าใช้จ่ายในการจัดเก็บ



รูป. 6. IPNS.

การค้นหาไฟล์ version ล่าสุด [2]

Node ไม่จำเป็นต้องจำ CID จำนวนมาก เพราะ IPFS สามารถค้นหาไฟล์ version ล่าสุดได้โดยใช้ระบบ IPNS decentralized naming system และ DNSLink สามารถใช้เพื่อจับคู่ CID กับชื่อ DNS ที่มนุษย์สามารถอ่านได้

## 2.2.4 หลักการพื้นฐานในการทำความเข้าใจ IPFS

### 2.2.4.1 การระบุตัวตนที่ไม่ซ้ำผ่านการระบุที่อยู่เนื้อหา

โดยใช้หลักการของ Content identifier หรือ CID ผ่านการ hash เป็นให้ ID ของไฟล์ที่ออกมาไม่ซ้ำกัน ทำให้ง่ายต่อการระบุตัวตน และเรียกดูไฟล์

### 2.2.4.2 การเชื่อมโยงเนื้อหาผ่าน Directed acyclic graphs (DAGs)

IPFS และระบบแบบกระจายอื่น ๆ ใช้ประโยชน์จากโครงสร้างข้อมูลที่เรียกว่า Directed acyclic graphs (DAGs) โดยเฉพาะอย่างยิ่ง พวกเขาใช้ Merkle DAG ซึ่งเป็น DAG ที่แต่ละ node มีตัวระบุที่ไม่ซ้ำกันซึ่งเป็น Hash ของเนื้อหาของ node หมายถึง แนวคิด CID ที่เรากล่าวถึงในส่วนก่อนหน้านี้

การระบุวัตถุข้อมูลด้วยค่าของ Hash คือการกำหนดที่อยู่เนื้อหา IPFS ใช้ Merkle DAG ที่ได้รับการปรับให้เหมาะสมสำหรับการแสดง directory และไฟล์ แต่คุณ สามารถจัดโครงสร้าง Merkle DAG ได้หลายวิธี ในการสร้างการนำเสนอ Merkle DAG สำหรับเนื้อหาของคุณ IPFS มักจะแยกเนื้อหาออกเป็นบล็อกๆ ก่อน การแบ่งไฟล์ ออกเป็นบล็อกหมายความว่าส่วนต่างๆ ของไฟล์สามารถมาจากแหล่งต่างๆ และได้รับการ ตรวจสอบสิทธิ์อย่างรวดเร็ว

สรุปคือ IPFS ต้องการให้มอบ CID ให้กับเนื้อหาและเชื่อมโยงเนื้อหานั้นเข้าด้วยกันใน Merkle DAG

### 2.2.4.3 การค้นพบเนื้อหาผ่าน Distributed hash tables (DHTs)

เพื่อค้นหาว่า peers ใดเป็นเจ้าของเนื้อหาที่กำลังติดตาม IPFS จะใช้ตาราง Hash แบบกระจายหรือ DHT ตาราง Hash เป็นฐานข้อมูลของคีย์ถึงค่าของมัน ตาราง Hash แบบกระจายคือตารางที่แบ่งตารางตาม peers ทั้งหมดในเครือข่ายแบบกระจาย หากต้องการค้นหาเนื้อหาจำเป็นต้องถาม peers เหล่านี้

Project libp2p เป็นส่วนหนึ่งของระบบ IPFS ที่ให้ DHT และจัดการ peers ที่เชื่อมต่อและพูดคุยกัน เมื่อรู้แล้วว่าเนื้อหาที่ต้องการอยู่ที่ไหน จะใช้ DHT อีกครั้งเพื่อค้นหาตำแหน่งปัจจุบันของ peers เหล่านั้น (การกำหนดเส้นทาง) ดังนั้น เพื่อเข้าถึงเนื้อหา ต้องใช้ libp2p เพื่อสืบค้น DHT แล้วสองครั้ง

เมื่อค้นพบเนื้อหา และพบตำแหน่งปัจจุบันของเนื้อหานั้นแล้ว ตอนนี้ต้องเชื่อมต่อกับเนื้อหานั้นและรับมัน (แลกเปลี่ยนเนื้อหา) ในการขบถลอกจากและส่งบล็อกไปยัง peers อื่น ๆ ปัจจุบัน IPFS ใช้โมดูลที่เรียกว่า BitSwap ช่วยให้เราสามารถเชื่อมต่อกับ peers อื่นๆ หรือ peers ที่มีเนื้อหาที่คุณต้องการ ส่งรายการที่ต้องการ (รายการบล็อกทั้งหมดที่สนใจ) และให้ peers นั้นส่งบล็อกทั้งหมดที่ร้องขอ เมื่อบล็อกเหล่านั้นมาถึง จะสามารถตรวจสอบได้โดย Hash เนื้อหาเพื่อรับ CID และเปรียบเทียบกับ CID ที่ขอไป CID เหล่านี้ยังอนุญาตให้คุณจัดบล็อกที่ซ้ำกันหากจำเป็น

มีโปรโตคอลการจำลองเนื้อหาอื่น ๆ ที่อยู่ระหว่างการสนทนาเช่นกัน ที่ได้รับการพัฒนามากที่สุดคือ Graphsync ยังมีข้อเสนอที่อยู่ระหว่างการสนทนาเพื่อขยายโปรโตคอล BitSwap เพื่อเพิ่มฟังก์ชันการทำงานเกี่ยวกับคำขอและการตอบกลับ

การจับคู่คีย์-ค่ามีสามประเภทที่แมปโดยใช้ DHT ได้แก่

#### 1. Provider records

จับคู่ตัวระบุข้อมูล (เช่น มัลติแฮช) กับเพียร์ที่โฆษณาว่าพวกเขามีเนื้อหานั้น และยินดีที่จะให้ข้อมูลแก่คุณ

- IPFS เพื่อค้นหาเนื้อหา
- IPNS บน PubSub เพื่อค้นหาสมาชิกคนอื่น ๆ ของหัวข้อ pubsub

#### 2. IPNS records

จับคู่คีย์ IPNS (เช่น แฮชของคีย์สาธารณะ) กับ records IPNS (เช่น ตัวชี้ที่ลงชื่อและกำหนดเวอร์ชันกับเส้นทาง เช่น /ipfs/bafyxyz...)

- IPNS

#### 3. Peer records

จับคู่ peerID กับชุดของที่อยู่หลายแห่งที่อาจเข้าถึงเพียร์นั้นได้

- IPFS เมื่อเรารู้จักเพียร์ที่มีเนื้อหา แต่ไม่ทราบที่อยู่ของเพียร์
- การเชื่อมต่อด้วยตนเอง (เช่น ipfs swarm connect /p2p/Qmxyz..)

ประเภทของ records เหล่านี้มีความหมายต่างกันเล็กน้อย แต่ทั้งหมดได้รับการอัปเดตโดยใช้โปรโตคอล DHT เดียวกัน ใน IPFS จะใช้ Kademlia

#### 2.2.4.4 Peer ID และ Multi Address

เมื่อเริ่มใช้งาน IPFS ระบบจะทำการระบุ Peer ID ขึ้นมาใช้แทน โดยจะเป็นค่า Hash ขนาด 52 ตัวอักษร

Peer ID จะใช้งานแทน IP Address version 4 และ 6 ที่อาจจะมีมากกว่า 1 Address ต่อ 1 ผู้ใช้งาน เพื่อความสะดวกรวดเร็ว Peer ID 1 ค่า จะสามารถใช้แทน IP Address ได้หลาย Address หรือที่เรียกว่า Multi Address ตัวอย่างเช่น

Peer ID ของ Cloudflare คือ

QmcfgsJsMtx6qJb74akCw1M24X1zFwgGo11h1cuhwQjtJP

สามารถ Map ค่า IP Address ได้ 2 Address คือ

/ip6/2606:4700:60::6/tcp/4009 (IP v6) และ /ip4/172.65.0.13/tcp/4009 (IP v4)

เมื่อทำการเรียกใช้งาน จะเรียกโดยใช้รูปแบบ

/ip6/2606:4700:60::6/tcp/4009/p2p/

QmcfgsJsMtx6qJb74akCw1M24X1zFwgGo11h1cuhwQjtJP

#### 2.2.5 ข้อดีของ IPFS

การกระจายความเสี่ยง เนื่องจากข้อมูลของเราไม่ได้ถูกเก็บเอาไว้ที่คอมพิวเตอร์เพียงเครื่องเดียว แต่กระจายไปอยู่ที่คอมพิวเตอร์ที่เป็น node ทั่วโลก ทำให้ข้อมูลของเรานั้นแทบจะไม่มีทางสูญหายได้เลย เพราะหาก node ที่ใกล้ที่สุดพัง หรือว่าปิดใช้งาน เรายังสามารถเข้าถึงข้อมูลจาก node อื่นๆ ที่ยังเปิดใช้งานได้อยู่

ทำให้เซ็นเซอร์เนื้อหายากขึ้น เนื่องจากไฟล์บน IPFS อาจมาจากหลายที่ จึงทำให้ไม่สามารถระบุต้นทางของเนื้อหาที่แนบได้ ทำให้การเซ็นเซอร์ ฟิเตอร์ หรือปิดกั้นการเข้าถึงข้อมูลนั้นเป็นไปได้ยาก เพราะเมื่อทำการปิดกั้นการเข้าถึงข้อมูลจาก node ใด node หนึ่ง ผู้ใช้งานก็ยังสามารถเข้าถึงข้อมูลนั้นได้จาก node อื่นๆ อยู่ดี

สามารถเพิ่มความเร็วของเว็บ หากผู้ใช้งานอยู่ห่างไกลจากผู้อัปโหลด ไม่จำเป็นต้องกังวลว่าจะเข้าถึงเว็บไซต์หรือข้อมูลได้ช้า เพราะผู้ใช้งานสามารถดึงข้อมูลจากคอมพิวเตอร์ที่เป็น node บริเวณใกล้เคียงแทนได้ ยังมี node อยู่ใกล้มากเท่าใด การเข้าถึงก็จะรวดเร็วมากยิ่งขึ้นเท่านั้น สิ่งนี้จะเป็นประโยชน์อย่างมาก หากบริเวณใกล้เคียงมี node มาก แต่ความเร็วอินเทอร์เน็ตน้อย

นอกจากนี้ถ้าหากผู้ใช้งานเปิดให้คอมพิวเตอร์ของเราให้เป็น node ของระบบ IPFS แล้ว เมื่อทำถูกต้องตามขั้นตอนที่ระบุเอาไว้ ผู้ใช้งานสามารถได้รับเงินดิจิทัลหรือเรียกว่า Filecoin จากการเป็น node ได้อีกด้วย

## 2.2.6 การเข้าร่วม IPFS

ถึงแม้ว่า IPFS จะมีเทคโนโลยีที่ซับซ้อน และดูสะดวกต่อการเข้าถึงข้อมูล แต่ระบบนี้จะทำงานได้อย่างมีประสิทธิภาพเมื่อมีผู้ใช้งานเปิดให้คอมพิวเตอร์ของตนเองเป็น node มากยิ่งขึ้น เมื่อผู้ใช้งานแชร์ไฟล์ของตัวเองผ่านระบบ IPFS แล้วปิดเครื่อง ไฟล์นั้นจะไม่สามารถเข้าถึงได้ ต้องทำให้แน่ใจว่ามี node อื่นเข้ามาเรียกดูไฟล์ของเราแล้ว ไฟล์นั้นๆ จึงจะคงอยู่ในระบบต่อไปแม้เราจะปิดเครื่องคอมพิวเตอร์ก็ตาม เนื่องจากจะมีการบันทึกสำเนาของไฟล์เอาไว้ใน node อื่นเพียงแค่ชั่วคราวเท่านั้น เพราะโดยค่าเริ่มต้น คอมพิวเตอร์ที่เป็น node จะทำการแชร์ไฟล์กับผู้อื่นในระยะเวลาที่จำกัด แต่สามารถทำให้ไฟล์ใช้งานได้อย่างถาวรมากขึ้นด้วยการปักหมุด ซึ่งจะบันทึกสำเนาลงในคอมพิวเตอร์ และทำให้พร้อมใช้งานบนระบบ IPFS จนกว่าจะตัดสินใจเลิกตรึง

เว็บไซต์ที่เปิดให้บริการแบบไม่เสียเงินนั้นมีมากมาย ยกตัวอย่างเช่น pinata ที่เปิดให้เราฝากข้อมูลในระบบ IPFS และทำการ Hash ข้อมูลนั้น (CID) ผ่าน API ให้เรียบร้อยพร้อมใช้งาน โดยพื้นที่ที่เปิดให้บริการฟรี มีขนาด 1 GB และต้องสมัครสมาชิก(ไม่เสียเงิน)

## 2.3 เครื่องมือที่ใช้ทำสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์

### 2.3.1 เบลนเดอร์

เบลนเดอร์ เป็นซอฟต์แวร์เสรี สำหรับงานคอมพิวเตอร์กราฟิกสามมิติ และต่างๆ อีกมากมาย สามารถใช้สร้าง โมเดลสองมิติ สามมิติ หรือสามารถตัดต่อวิดีโอ รองรับภาษาไพทอน สำหรับเขียนสคริป ถูกพัฒนาขึ้นเพื่อใช้ภายในสตูดิโอแอนิเมชัน NeoGeo และ Not a Number Technologies (NaN) ในประเทศฮอลแลนด์ โดย Ton Roosendaal ทำงานได้หลายระบบ เช่น Windows, Linux, OS X, BSD

ตัวอย่างสื่อที่ใช้ เบลนเดอร์

1. Elephants Dream/Project Orange : ภาพยนตร์สั้น
2. Plumiferos : ภาพยนตร์

### 2.3.2 ภาษาที่ใช้ในการทำ Website

#### 2.3.2.1 ภาษา HTML

HTML (Hypertext Markup Language) พัฒนาโดย ทิม เบอร์เนิร์ส-ลี แห่งศูนย์ปฏิบัติการวิจัยทาง อนุภาคฟิสิกส์ของยุโรป(CERN) แห่งกรุงเจนีวา ประเทศสวิตเซอร์แลนด์ เป็นภาษาคอมพิวเตอร์ในรูปแบบหนึ่งที่ใช้สำหรับสร้างหน้าเว็บ โดยมีนามสกุล .html มี Web Browser เป็นตัวแปลงภาษา HTML เป็นรูปแบบหน้าเว็บ ลักษณะของไฟล์ HTML ประกอบไปด้วย แท็ก (Tag) ต่างๆ ที่เป็นคำ สั่งของ HTML ซึ่งแท็กจะอยู่ภายในเครื่องหมาย < และ >

### 2.3.2.2 ภาษา PHP

ผู้สร้างภาษา PHP มีชื่อว่า รัสมีส เลอร์เตอร์ฟ PHP มาจากคำว่า PHP Hypertext Preprocessor เป็นภาษาสคริปต์ในการใช้งานจริงมักใช้งานร่วมกับภาษา HTML โดยบันทึกไฟล์เป็นนามสกุล .php ทาง Web Browser จะประมวลผลภาษา PHP ออกมาก่อน จากนั้นค่อยประมวลผลภาษา HTML ที่ถูกใช้งานร่วมกัน แต่ถ้าเราเขียนภาษา PHP ลงไปในไฟล์ที่มีนามสกุล .html ทาง Web Browser จะไม่สามารถประมวลผล PHP ขึ้นบนหน้าเว็บได้

### 2.3.2.3 ภาษา CSS

ภาษา CSS หรือ Cascading Style Sheets เป็นภาษาที่ใช้พัฒนาลักษณะรูปแบบ ใส่พื้นหลัง หรือเพิ่มกรอบข้อความ ของหน้าเว็บ เพื่อเพิ่มความสวยงามให้หน้าเว็บ เริ่มพัฒนาในปี ค.ศ. 1994 ถูกกำหนดมาตรฐานโดย W3C

มีรูปแบบการใส่ CSS ใน HTML ถึง 3 แบบ ได้แก่

1. แบบ Inline Styles คือการใส่คำสั่งไว้ในส่วน tag เปิด
2. แบบ Internal CSS Style คือการใส่คำสั่งไว้ในส่วน head ของ html
3. แบบ External Style Sheet คือการสร้างไฟล์ .css เอาไว้นอก HTML แล้วทำการ Link เข้ามาในส่วนของ head ใน HTML

### 2.3.2.4 ภาษา Javascript

ถูกออกแบบและสร้างโดย Brendan Eich สำหรับเป็นภาษาสคริปต์ทำงานบน Web Browser Navigator ที่เป็นผลิตภัณฑ์ของบริษัท Netscape มีรูปแบบการเขียนภาษาที่ใช้คล้ายคลึงกับ ภาษา C

ภาษา Javascript ทำให้หน้าเว็บที่ในตอนแรกนั้นเป็นแบบ Static สามารถตอบโต้กับผู้ใช้ได้โดยไม่ต้องรีเฟรชหน้าใหม่ (Dynamic) เพิ่มลูกเล่น และความสะดวกให้กับการใช้งานเว็บไซต์เป็นอย่างมาก

โดยการใช้งานนั้นใช้ร่วมกับภาษา HTML ทำให้หน้าเว็บมีความเคลื่อนไหว อาจจะใช้ตรวจสอบความถูกต้องของข้อมูลของผู้ใช้งานกรอก หรือใช้เพื่อให้เกิดความเปลี่ยนแปลงบนหน้าเว็บ

## 2.3.3 Visual Studio Code

VS Code หรือ Visual Studio Code จากบริษัทไมโครซอฟต์ เป็นโปรแกรมประเภท Editor ใช้ในการแก้ไขโค้ดที่มีขนาดเล็ก แต่มีประสิทธิภาพสูง เป็น OpenSource โปรแกรมจึงสามารถนำมาใช้งานได้โดยไม่เสียค่าใช้จ่าย เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานหลายแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows , macOS และ Linux รองรับหลายภาษาทั้ง JavaScript,



TypeScript และ Node.js ในตัว และสามารถเชื่อมต่อกับ Git ได้ง่าย สามารถนำมาใช้งานได้ง่ายไม่ซับซ้อน มีเครื่องมือและส่วนขยายต่าง ๆ ให้เลือกใช้มากมาย รองรับการใช้งานภาษาอื่น ๆ ทั้ง ภาษา C++ , C# , Java , Python , PHP หรือ Go สามารถปรับเปลี่ยน Themes ได้ มีส่วน Debugger และ Commands

### 2.3.4 เครื่องมืออื่นๆ

#### 2.3.4.1 โปรแกรมตัดต่อ

ในด้านของตัววิดีโอ ผู้จัดทำเลือกใช้งานโปรแกรมเบรนเดอร์ ในฟังก์ชันการตัดต่อวิดีโอ และ โปรแกรม VN Video Editor ซึ่งใช้ในอุปกรณ์เคลื่อนที่ได้อย่างสะดวกสบาย

ด้านรูปภาพประกอบ เลือกใช้โปรแกรม Procreate และ Adobe Photoshop CS6 ซึ่งเป็นโปรแกรมที่นิยมในปัจจุบัน เมื่อมีปัญหาในการใช้งาน ผู้จัดทำสามารถแก้ไขได้ด้วยตนเอง

#### 2.3.4.2 เว็บไซต์จำเป็นต่อการฝากข้อมูล

1. YouTube ใช้ในการฝากวิดีโอสื่อการสอน โดย เป็นเว็บไซต์ที่ให้ผู้ใช้งานสามารถอัปโหลดและแลกเปลี่ยนคลิปวิดีโอผ่านทางเว็บไซต์ ก่อตั้งเมื่อ 15 กุมภาพันธ์ พ.ศ. 2548 โดย แซต เฮอร์ลีย์, สตีฟ เซง และ ยาวีต คาริม อดีตพนักงานบริษัทเพย์พาล ในปัจจุบันยูทูปมีพนักงาน 67 คน และมีสำนักงานอยู่ที่ ซานบรูโนในมลรัฐแคลิฟอร์เนีย ปัจจุบันเป็นส่วนหนึ่งของกูเกิล

2. Pinata ใช้ในการฝากข้อมูลแบบ IPFS โดย Web Pinata นั้นง่ายต่อการอัปโหลดและจัดการไฟล์บน IPFS Interface ผู้ใช้ที่เป็นมิตรของเรารวมกับ IPFS API ทำให้ และมีพื้นที่ให้แบบไม่เสียค่าใช้จ่ายถึง 1GB

3. IPFS Desktop เป็น Interface ที่สามารถใช้ในการ Upload ข้อมูลเข้าสู่ระบบของ IPFS ได้เช่นเดียวกัน โดยสามารถ Download ได้ผ่านเว็บไซต์ ipfs.io

#### 2.3.4.3 emulsiV

เป็นโปรแกรมออนไลน์ที่จำลองการทำงานของชุดคำสั่งใน RISC-V

### บทที่ 3

#### การดำเนินงาน/วิธีวิจัย

โครงการนี้มีจุดมุ่งหมายสองประการ หนึ่ง คือการจัดทำสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์ เพื่อเป็นตัวช่วยในการศึกษาของนักศึกษาคณะวิศวกรรมศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์ ที่จำเป็นต้องใช้องค์ความรู้ในการเรียนต่อขั้นสูง และการทำงานในอนาคต สอง ศึกษาเชิงเปรียบเทียบระบบอินเทอร์เน็ตแบบรวมศูนย์ และแบบกระจาย ผ่านการอัปโหลดสื่อการสอนขึ้นไปบนระบบอินเทอร์เน็ตทั้งสอง และเปรียบเทียบส่วนดีส่วนเสียที่เกิดขึ้น เพื่อให้สามารถเลือกใช้ระบบอินเทอร์เน็ตที่เหมาะสม และมีประสิทธิภาพต่อสื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์มากที่สุด

#### 3.1 การวางแผนการดำเนินงาน

##### 3.1.1 กลุ่มเป้าหมาย

กลุ่มเป้าหมายของโครงการนี้มีด้วยกันทั้งหมด 2 กลุ่มใหญ่ ดังนี้

##### 3.1.1.1 กลุ่มเป้าหมายภายในคณะวิศวกรรมศาสตร์

จากกลุ่มผู้เรียนรายวิชา CN210 สถาปัตยกรรมคอมพิวเตอร์ ที่เป็นรายวิชาบังคับของภาควิชา สามารถประมาณได้คร่าวๆ ดังนี้

1. นักศึกษาในสาขาวิศวกรรมคอมพิวเตอร์ที่ลงเรียนวิชา CN210 ประมาณ 50 คน
2. นักศึกษาในสาขาวิศวกรรมคอมพิวเตอร์ที่ต้องการสืบค้นความรู้ ประมาณ 150 คน
3. อาจารย์ผู้สอน 1 คน

##### 3.1.1.2 กลุ่มเป้าหมายนอกคณะวิศวกรรมศาสตร์

นอกจากนักศึกษาและอาจารย์ของคณะวิศวกรรมศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์ ที่ใช้โครงข่ายอินเทอร์เน็ตภายในคณะ ทางผู้จัดทำเล็งเห็นว่าอาจมีผู้ที่สนใจในเนื้อหาวิชานี้เพิ่มจากภายนอกคณะ ไม่ว่าจะเป็นทั้งในมหาวิทยาลัย หรือ นอกมหาวิทยาลัย

### 3.1.2 แบบแผนการดำเนินงาน

สามารถแบ่งขั้นตอนในการดำเนินงานได้ดังนี้

#### 3.1.2.1 วางแผนการดำเนินงานและรวบรวมข้อมูล

ตลอดระยะเวลาในการทำโครงการ เป็นเวลาประมาณ 8 เดือน ได้มีการวางแผน และแบ่งสัดส่วนการทำงานดังตารางที่ 1.6 ตารางการดำเนินงาน

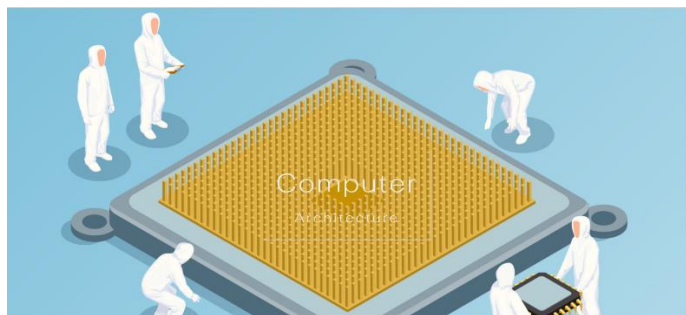
โดยแบ่งออกเป็น 2 ช่วง คือช่วงภาคเรียนที่ 1 ตั้งแต่เดือนสิงหาคม ถึง เดือนธันวาคม พ.ศ. 2564 ทางผู้จัดทำได้เริ่มต้นวางแผนและพูดคุยกับอาจารย์ที่ปรึกษา ตลอดจนรวบรวม และศึกษาข้อมูลที่เกี่ยวข้องกับโครงการทั้งหมด จากนั้น นำเอาองค์ความรู้ดังกล่าวไปใช้ในการออกแบบการสร้างสื่อการสอน และจัดทำขึ้นมา

ช่วงภาคเรียนที่ 2 ตั้งแต่เดือนมกราคม ถึง เดือนเมษายน พ.ศ. 2565 ทางผู้จัดทำจะดำเนินการปรับปรุงและแก้ไขสื่อการสอนให้แล้วเสร็จ แล้ววางแผนขั้นตอนการเปรียบเทียบระบบอินเตอร์เน็ตทั้งสองแบบ คือ แบบรวมศูนย์ และแบบกระจาย เพื่อเตรียมเครื่องมือให้พร้อมสำหรับการเก็บรวบรวมผลลัพธ์ที่ได้จากการเปรียบเทียบ จากนั้น นำผลที่ได้รับมาวิเคราะห์ และปรับปรุงแก้ไขวิธีการการใช้ดำเนินการเปรียบเทียบให้สมบูรณ์ที่สุด แล้วจึงสรุปผลว่าระบบอินเตอร์เน็ตแบบใดนั้นเหมาะกับเว็บไซต์สื่อการสอนวิชาสถาปัตยกรรมคอมพิวเตอร์มากที่สุด

#### 3.1.2.2 การออกแบบ และจัดทำสื่อการสอน

จากวัตถุประสงค์ของโครงการ และความตั้งใจของผู้จัดทำ ที่ต้องการจัดทำสื่อการสอน วิชาสถาปัตยกรรมคอมพิวเตอร์ ที่นักศึกษาสามารถเข้าถึงได้ง่าย จึงได้จัดทำเว็บไซต์สื่อการสอน ขึ้นมาเพื่อตอบวัตถุประสงค์ในข้อนี้

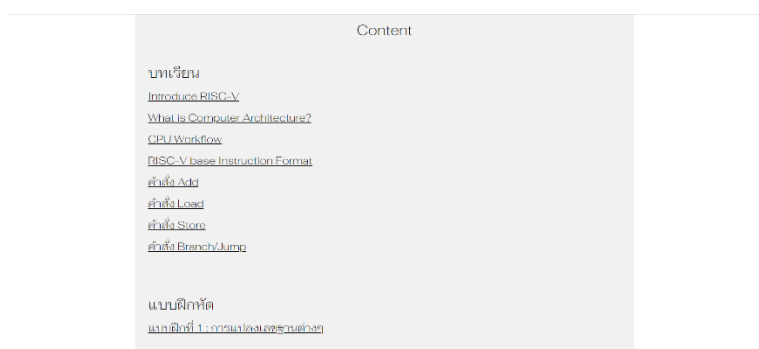
นอกจากจะสามารถเข้าถึงเนื้อหาได้ง่าย รูปแบบข้อมูลที่นำเสนอออกมาก็จำเป็นที่จะต้องน่าอ่าน และดึงดูดใจผู้ใช้อย่าง ทางผู้จัดทำได้เลือกใช้สื่อหลายหลายรูปแบบ ไม่ว่าจะเป็นรูปภาพนิ่ง รูปภาพเคลื่อนไหว รูปภาพสามมิติ คลิปวิดีโอ ตลอดจนสไลด์สรุปความรู้ มานำเสนอให้น่าสนใจมากยิ่งขึ้น



(รูปที่ 7 : Homepage – Header ของทุกหน้าเว็บไซต์สื่อ  
การสอน วิชา Computer Architecture)



(รูปที่ 8: Homepage – แนะนำเว็บไซต์สื่อการสอน  
วิชา Computer Architecture )

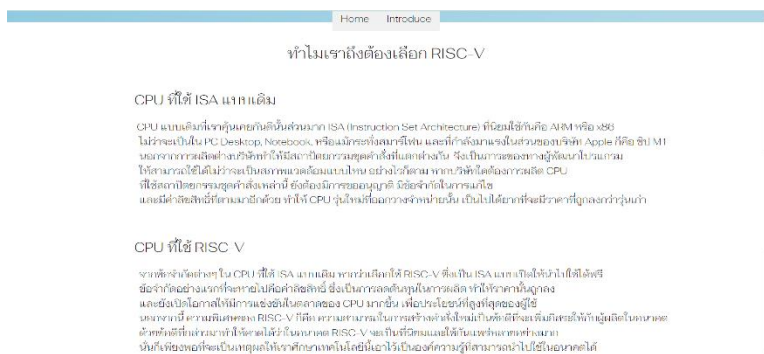


(รูปที่ 9: Homepage – หัวข้อที่มีในเว็บไซต์สื่อการสอน  
วิชา Computer Architecture )

เนื้อหาในเว็บไซต์นั้นจะทำการแบ่งหัวข้อให้ผู้เข้าใจได้ง่าย สามารถแบ่งได้เป็น 2 ส่วน คือ

ส่วนของบทเรียน

1. Introduce RISC-V เป็นหน้าที่อธิบายว่าทำไมเราถึงต้องเรียนรู้เกี่ยวกับ RISC-V และมีความสำคัญอย่างไร



(รูปที่ 10: หัวข้อ Introduce – เนื้อหาแนะนำว่าทำไมถึงต้องเรียนรู้เกี่ยวกับ RISC-V )

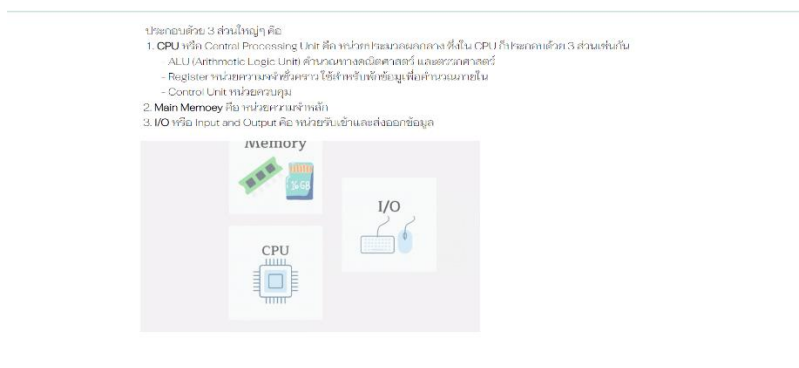


(รูปที่ 11: หัวข้อ Introduce – สไลด์สรุปเกี่ยวกับการแนะนำ RISC-V )

2. What is Computer Architecture? อธิบายเกี่ยวกับสถาปัตยกรรมคอมพิวเตอร์ว่าประกอบด้วยอะไรบ้าง และประวัติความเป็นมาของคอมพิวเตอร์



(รูปที่ 12: หัวข้อ What is Computer Architecture - ส่วนของเนื้อหาที่อธิบายเกี่ยวกับสถาปัตยกรรมคอมพิวเตอร์ )



(รูปที่ 13: หัวข้อ What is Computer Architecture - ส่วนของเนื้อหาภาพประกอบ)

3. CPU Workflow อธิบายขั้นตอนการประมวลผลของ CPU
4. คำสั่งใน RISC-V อธิบายชุดคำสั่งพื้นฐานของ RISC-V ในภาษา assembly รวมไปถึง base format ของคำสั่งใน RISC-V

### คำสั่งใน RISC-V

หน้าตาของคำสั่งใน RISC-V ในภาษา assembly

| RISC-V operands              |   |  |  |  |
|------------------------------|---|--|--|--|
| Name                         | Example   | Comments   |  |  |
| 32 registers                 | x0-x31  | Fast locations for data. In RISC-V, data must be in registers to perform arithmetic. Register x0 always equals 0.  |  |  |
| 2 <sup>64</sup> memory words | Memory[0], Memory[8], ..., Memory[18,446,744,073,709,551,608] | Accessed only by data transfer instructions. RISC-V uses byte addresses, so sequential doubleword accesses differ by 8. Memory holds data structures, arrays, and spilled registers. |  |  |

| RISC-V assembly language |                  |                 |                               |                                    |
|--------------------------|------------------|-----------------|-------------------------------|------------------------------------|
| Category                 | Instruction      | Example         | Meaning                       | Comments                           |
| Arithmetic               | Add              | add x5, x6, x7  | $x5 = x6 + x7$                | Three register operands; add       |
|                          | Subtract         | sub x5, x6, x7  | $x5 = x6 - x7$                | Three register operands; subtract  |
|                          | Add immediate    | addi x5, x6, 20 | $x5 = x6 + 20$                | Used to add constants              |
|                          | Load doubleword  | ld x5, 40(x6)   | $x5 = \text{Memory}[x6 + 40]$ | Doubleword from memory to register |
|                          | Store doubleword | sd x5, 40(x6)   | $\text{Memory}[x6 + 40] = x5$ | Doubleword from register to memory |
|                          | Load word        | lw x5, 40(x6)   | $x5 = \text{Memory}[x6 + 40]$ | Word from memory to register       |

(รูปที่ 14: หัวข้อ คำสั่งใน RISC-V -

ส่วนของเนื้อหาภาพประกอบ )

|                      |                                      |                  |                                   |  |
|----------------------|--------------------------------------|------------------|-----------------------------------|--|
| Conditional branch   | Branch if less, unsigned             | bltu x5, x6, 100 | if (x5 < x6) go to PC+100         | PC-relative branch if registers less, unsigned             |
|                      | Branch if greater or equal, unsigned | bgeu x5, x6, 100 | if (x5 >= x6) go to PC+100        | PC-relative branch if registers greater or equal, unsigned |
| Unconditional branch | Jump and link                        | jal x1, 100      | $x1 = \text{PC}+4$ ; go to PC+100 | PC-relative procedure call                                 |
|                      | Jump and link register               | jalr x1, 100(x5) | $x1 = \text{PC}+4$ ; go to x5+100 | Procedure return; indirect call                            |

ศึกษาเพิ่มเติมในส่วน operands

- มี Register 32 ตัว (x0-x31) โดย Register ตัวที่ 0 (x0) นั้น เป็นตัวพิเศษเพราะค่าจะเป็น 0 เสมอ ไม่สามารถเปลี่ยนได้
- มี 2<sup>64</sup> memory words RISC-V ใช้ไบต์แอดเดรส ดังนั้นการเข้าถึงแบบไบต์แอดเดรสตามลำดับจึงต้องทำงานกัน 8 (Memory[0], Memory[8],...,Memory[18,446,744,073,709,551,608])

ตัวอย่างคำสั่งใน RISC-V

add a, b, c // รวมถึงบวก b เข้ากับ c แล้วเก็บไว้ใน a จากนั้นจะลงการทำงาน

โดยทั่วไปแล้วจะทำงานเป็นชุดคำสั่ง คือ มีการทำงานมากกว่า 1 คำสั่งขึ้นไป ยกตัวอย่างเช่น

add a, b, c // รวมถึงบวก b เข้ากับ c แล้วเก็บไว้ใน a

add a, a, d // รวมถึงบวก b, c เข้ากับ a ในคำสั่งก่อนหน้านี้ และ d เข้าด้วยกัน แล้วเก็บไว้ใน a (a เก็บค่าใหม่)

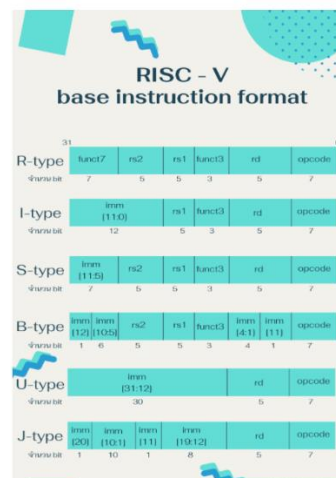
add a, a, e // รวมถึงบวก b, c, d เข้ากับ a ในคำสั่งก่อนหน้านี้ และ e เข้าด้วยกัน แล้วเก็บไว้ใน a (a เก็บค่าใหม่)

เมื่อทำงาน 3 คำสั่งแล้ว จากนั้นจะจบการทำงาน 1 ชุดคำสั่ง

(รูปที่ 15: หัวข้อ คำสั่งใน RISC-V -

ส่วนของการสรุปเนื้อหาเกี่ยวกับคำสั่งในภาษา assembly )

Base Instruction Format



(รูปที่ 16: หัวข้อ คำสั่งใน RISC-V -

Base Instruction Format ใน RISC-V)

5. คำสั่ง Add อธิบายขั้นตอนของคำสั่ง Add
6. คำสั่ง Load อธิบายขั้นตอนของคำสั่ง Load
7. คำสั่ง Store อธิบายขั้นตอนของคำสั่ง Store
8. คำสั่ง Branch/Jump อธิบายขั้นตอนของคำสั่ง Branch/Jump

ส่วนของแบบทดสอบ

1. แบบฝึกที่ 1 : การแปลงเลขฐานต่างๆ

ซึ่งแบบทดสอบนั้นเป็นแบบตัวเลือก เมื่อทำการส่งคำตอบแล้วจะมีการรวมคะแนนให้ และมีเฉลยบอกอยู่ด้านล่าง ทำให้ง่ายต่อการตรวจคำตอบ และสามารถทำความเข้าใจในข้อที่ทำได้ทันที

### 3.1.2.2 ปัญหาที่พบในการดำเนินงาน

ตลอดระยะเวลาในการดำเนินงานในภาคเรียนที่ 1 นั้น พบปัญหาอยู่บ้าง แต่เนื่องจากยังเป็นช่วงเริ่มต้น จึงค่อนข้างมีความราบรื่นพอสมควร

ปัญหาเบื้องต้นที่พบในการดำเนินงาน

1. ปัญหาด้านภาษา จากหนังสือและเว็บไซต์ที่ใช้อ้างอิงข้อมูลนั้นยังไม่มีแปลเป็นภาษาไทย ทำให้ต้องศึกษาจาก Textbook และ เว็บไซต์ของต่างประเทศ ซึ่งเมื่อนำมาสรุปเองแล้วอาจจะยังสื่อสารได้ไม่ดีเท่าที่ควรจะเป็น
2. ปัญหาด้านภาษาที่ใช้เขียนเว็บไซต์ บาง Syntax ที่ต้องการใช้ แต่เมื่อใช้จริงแล้วมีปัญหา เช่น ใช้ iframe ในการแสดงสไลด์ แต่เมื่อลองใช้จริงแล้วไม่แสดงผล



## บรรณานุกรม

### หนังสือและบทความในหนังสือ

1. David A. Patterson and John L. Hennessy, *Computer Organization and Design THE HARDWARE/SOFTWARE INTERFACE*, Cambridge, MA, United States: Morgan Kaufmann, 2017.

### สื่ออิเล็กทรอนิกส์

2. Wikipedia contributors. "RISC-V." Wikipedia, The Free Encyclopedia. <https://en.wikipedia.org/w/index.php?title=RISC-V&oldid=1054414570>. (accessed September 12, 2021).
3. Wikipedia contributors. "InterPlanetary File System." Wikipedia, The Free Encyclopedia. [https://en.wikipedia.org/wiki/InterPlanetary\\_File\\_System](https://en.wikipedia.org/wiki/InterPlanetary_File_System). (accessed September 12, 2021).
4. Wikipedia contributors. "YouTube." Wikipedia, <https://en.wikipedia.org/wiki/YouTube>. (accessed September 12, 2021).
5. Wikipedia contributors. "Blender (software)." Wikipedia, The Free Encyclopedia. [https://en.wikipedia.org/wiki/Blender\\_\(software\)](https://en.wikipedia.org/wiki/Blender_(software)). (accessed September 12, 2021).
6. IPFS. "How IPFS works." IPFS. <https://ipfs.io/> (accessed November 10, 2021).
7. IPFS. "What is IPFS?." IPFS Docs. <https://docs.ipfs.io/concepts/what-is-ipfs/> (accessed November 10, 2021).
8. IPFS. "How IPFS works." IPFS Docs. <https://docs.ipfs.io/concepts/how-ipfs-works/> (accessed November 10, 2021).
9. Artiya. "IPFS คืออะไร." Artiya. <https://artiya4u.medium.com/what-is-ipfs-2d54241a7b4b> (accessed November 10, 2021).
10. Pinata. "MADE FOR BUILDERS Build the next unicorn." Pinata. <https://www.pinata.cloud/> (accessed November 10, 2021).

11. klumtee10.ภาษาสำหรับการสร้างเว็บไซต์.” klumtee10.  
<https://sites.google.com/site/klumtee10/home/phasa-sahrab-kar-srang-web>  
 (accessed November 10, 2021).
12. MarcusCode. “ภาษา JavaScript.” MarcusCode. <http://marcuscode.com/lang/javascript>  
 (accessed November 10, 2021).
13. Unknown. “สถาปัตยกรรมคอมพิวเตอร์ (Computer Architecture).”  
[wuttipongthoing.blogspot.com](http://wuttipongthoing.blogspot.com). <http://wuttipongthoing.blogspot.com/2014/08/computer-architecture.html> (accessed November 10, 2021).
14. แมตต์มิลส์. “RISC-V ข้อมูลจำเพาะและคุณสมบัติของ Freer ISA.” ITIGIC เทคนิควิธีการเคล็ดลับและเทคนิค. <https://itigic.com/th/risc-v-specifications-and-features-of-freer-isa/> (accessed November 10, 2021).
15. Microsemi. “RISC-V CPUs.” Microchip Microsemi Product Portfolio.  
<https://www.microsemi.com/product-directory/mi-v-risc-v-ecosystem/4406-risc-v-cpus>  
 (accessed November 10, 2021).

## รูปภาพ

[1] *cryptographic hash digest*. ipfs.io. (n.d.)

[2] *How IPFS works*. ipfs.io. (n.d.)

