**區塊鏈技術與應用**

**Blockchain Techniques and Applications**

**Homework 01**

**Guest Lecture: Smart Contract**

|  |  |
| --- | --- |
| 班　級： | 資工碩一 |
| 學　號： | 113598043 |
| 姓　名： | 張育丞 |

March 30, 2025

**Week 01**

Date: March 11, 2025

**Task & Check Points**

|  |  |
| --- | --- |
| **CP** | **Task** |
| ☑ | 準備 MetaMask Wallet。 |
| ☑ | 加入 Holesky & Sepolia。 |
| ☑ | 透過 Faucet 領取 Holesky & Sepolia的錢。 |
| ☑ | 製作NFT的組圖（至少16張），並上傳至IPFS。 |

* **準備 MetaMask Wallet**

1. 在Chrome Web Store搜索「MetaMask」。
2. 點擊「加到Chrome」。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 網頁, 網站 的圖片  AI 產生的內容可能不正確。 |

1. 點擊「新增擴充功能」。

|  |
| --- |
|  |

1. 檢查「擴充功能」清單，是否出現MetaMask，並釘選於佇列。

|  |
| --- |
|  |

1. 將「使用條款」進行，以及進行「產生新的錢包」。

|  |
| --- |
|  |

1. 將「推播資訊」打勾，並且進行「我同意」。

|  |
| --- |
|  |

1. 進行建立密碼，接著將「聲明資訊」進行打勾，接著「產生新的錢包」。

|  |
| --- |
|  |

1. 進行「保護我的錢包」的按鈕，接著Windows 10和11會跳出在Google Chrome進行保護政策。

|  |
| --- |
|  |

1. 接著會出現12個「助憶詞」，可以透過截圖，將其順序與內容進行存取，使用複製到剪貼簿雖然方便，但是順序有時會跑掉。

|  |
| --- |
|  |

1. 紀錄完畢後，進行確認。
2. 輸入挖空的「助憶詞」，進行驗證。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 圖表, 設計 的圖片  AI 產生的內容可能不正確。 |

1. 輸入密碼即可進入錢包。

|  |
| --- |
|  |

* **加入Holesky & Sepolia**

1. 進入錢包後，先進行Sepolia的新增，因此需將列表最下面的「顯示測試網路」進行啟動，接著Sepolia就能出現。

|  |
| --- |
|  |

1. Holesky的新增需要使用「加入客製化網路」，因此點擊其按鈕，接著看到許多欄位。

|  |
| --- |
|  |

1. 對應欄位將依序輸入，相關內容已於下圖展示，將不另贅述。

|  |
| --- |
|  |

1. 完成第3步驟，將會看見有Holesky的錢包。

|  |
| --- |
|  |

* **透過 Faucet 領取 Holesky & Sepolia的錢**

1. 將前往Google Cloud Web3 – Ethereum Faucet這個網頁進行領錢的手續。

|  |
| --- |
|  |

1. 在選單中將分別針對Holesky和Sepolia進行選擇，並填入Wallet Address，接著完成提領，這過程將會需要數秒鐘的等待。

|  |
| --- |
|  |
|  |

1. 當領取完畢，並再次點擊領取時，將會出現需要一天後，才能再次領取的警告。

|  |  |
| --- | --- |
|  |  |

* **製作NFT的組圖（16張），並上傳至IPFS**

1. 先完成Background、Flag、Body、Anumal等圖片的蒐集，並將尺寸為1920 x 1920，並對於Background外的其他圖片需要去除背景，避免產生時遭到圖層覆蓋。
2. 由於4個Layer，我對於每個類別規劃將做出6張圖，因此將會獲得6x6x6x6=1296張的隨機產生圖。

|  |  |  |
| --- | --- | --- |
| Layer | Count | Image |
| Background | 6 | 一張含有 鮮豔, Rectangle, 正方形, 視窗 的圖片  AI 產生的內容可能不正確。 |
| Animal | 6 | 一張含有 螢幕擷取畫面, 黑色 的圖片  AI 產生的內容可能不正確。 |
| Body | 6 | 一張含有 螢幕擷取畫面 的圖片  AI 產生的內容可能不正確。 |
| Flag | 6 | 一張含有 螢幕擷取畫面, 黑色 的圖片  AI 產生的內容可能不正確。 |

1. 網路搜尋「Mintrables」，並註冊。

|  |
| --- |
|  |

1. 首先建立一個「我的專案」。

|  |
| --- |
|  |

1. 接著對專案名稱命名，並設定尺寸為圖片所用的1920x1920。

|  |
| --- |
|  |

1. 在專案中「新增群組（+Group）」。

|  |
| --- |
|  |

1. 在群組中依據圖片的類別Background、Flag、Body、Anumal等，新增圖層（Layer），其順序相當重要，需要避免Background蓋到其他圖層。

|  |
| --- |
|  |

1. 對於每一個圖層（Layer）進行圖片新增。

|  |
| --- |
|  |

1. 接著進行圖片產生。
2. 圖片產生時預設為10000，單因排列組合僅有1296，故不能大於1296，但避免後續使用的FileBase等限制，故僅產生18張，其編號將由0至17進行編號。
3. 產生完畢後，將會在電腦本地端收到與專案名稱一樣的壓縮檔。
4. 壓縮檔進行解壓縮後，將會獲得與壓縮檔同名的資料夾。
5. 在該資料夾下，會有metadata和images兩個資料夾。分別儲存0 ~ 17共18張的資料及圖片。

|  |
| --- |
|  |

1. 在Chrome Web Store搜索「IPFS 同伴」，並完成安裝。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片  AI 產生的內容可能不正確。 |

1. 點擊「新增擴充功能」。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 字型 的圖片  AI 產生的內容可能不正確。 |

1. 將會看到歡迎頁。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 設計 的圖片  AI 產生的內容可能不正確。 |

1. 在「擴充功能」清單中也能找到該擴充套件，點擊後將能看到以下畫面。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 字型, 標誌 的圖片  AI 產生的內容可能不正確。 |

1. 搜索與前往FileBase的網站，在該網站中提供5GB及1000個檔案的額度使用，並且無須綁定任何信用卡，可以註冊後立即使用。
2. 進行FileBase的註冊。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 字型, 網頁 的圖片  AI 產生的內容可能不正確。 |

1. 註冊完畢將進行Bucket的建立，其名稱可以自行命名，Storage Network將為預設的IPFS (Always public)。

|  |
| --- |
|  |

1. 進入建立完成的Bucket後進行上傳，上傳內容為第13步驟的images的資料夾。
2. 上傳完畢後，每張圖片皆有IPFS網址，將其與metadata中的圖片之images的連結進行更換。
3. 進行上傳metadata資料夾。

|  |
| --- |
|  |

1. 接著可以檢查看接著可以檢查看看metadata是否有產生資料夾的IPFS。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| [0.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/0.png) | [1.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/1.png) | [2.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/2.png) | [3.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/3.png) | [4.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/4.png) | [5.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/5.png) |
|  |  |  |  |  |  |
| [6.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/6.png) | [7.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/7.png) | [8.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/8.png) | [9.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/9.png) | [10.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/10.png) | [11.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/11.png) |
|  |  |  |  |  |  |
| [12.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/12.png) | [13.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/13.png) | [14.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/14.png) | [15.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/15.png) | [16.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/16.png) | [17.png](https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU/17.png) |

* IPFS Gateway URL

<https://shallow-salmon-marmot.myfilebase.com/ipfs/QmULZjDJi6U8BZCrUVXJyAoZAPSUTXAEGevY673fcPACaU>

**Week 02**

Date: March 18, 2025

**Task & Check Points**

|  |  |
| --- | --- |
| **CP** | **Task** |
| ☑ | 以ERC-20佈建Sepolia的鏈。 |
| ☑ | 使用合約各項功能(須包含多個使用者)。 |
| ☑ | 額外功能 Buy & Burn。 |
| ☑ | OnlyOwner相關功能測試，請給出不是Owner的失敗測試。 |

* 以ERC-20佈建Sepolia的鏈

1. 進入「Remix - Ethereum IDE」網頁。

|  |
| --- |
|  |

1. 點擊「建立新檔案」，並命名為「YCToken.sol」，這名稱可自行命名。

|  |
| --- |
|  |

1. 對於「YCTOken.sol」完成程式撰寫。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片  AI 產生的內容可能不正確。 |

1. 勾選「Auto Complie」，並點擊「Complie YCToken.sol」。

|  |
| --- |
|  |

1. 會發現在File Explorer多出許多檔案。
2. 接著到「Deploy & Run Transctions」的分頁中，進行連線與Deploy。

|  |
| --- |
|  |

1. 經過和交易與核准後，將會得到以下畫面。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片  AI 產生的內容可能不正確。 |

* View Sepolia Transaction

<https://sepolia.etherscan.io/tx/0x2f68a11cff02b4980ff0b13e28a724102b7e48ea9642a69eb95342c5636731e5>

1. 對「YCToken.sol」進行Flatten。
2. 在產生的「YCToken\_flattened.sol」會缺少「// SPDX-License-Identifier: MIT」，需要協助補上。
3. 到Solidity Compiler 分頁設定「Auto Complie」和「Context (YCToken\_flattened.sol)」，並進行Complie。
4. 進行設定為「Injected Provider - MetaMask」與「YCToken – YCToken\_flattened.sol」後進行Deploy和確認。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片  AI 產生的內容可能不正確。 |

* View Sepolia Transaction

<https://sepolia.etherscan.io/tx/0xb835b6c7421141e019f6a7446e202744f00bd44260bcd591c8099e25911303ac>

1. 透過交易後Address網頁找到Contract，並進入Verify and Public 超連結，進行驗證與公開。
2. 驗證時，需填寫Address和Solidity (Single file)，並選擇v0.8.26+commit 8a97fa7a和MIT License (MIT)，接著貼上「YCToken\_flattened.sol」裡面的code進行驗證，接著就會公開你的原始碼。

* View Contract Source code

<https://sepolia.etherscan.io/address/0xa6d4503a8a1b2f8a43214475172bc34fac80b643#code>

1. 連接MetaMask Wallet。

|  |
| --- |
|  |

1. 在網頁進行blanceOF，在此需填入address。
2. 返回Deploy & Run Transactions，下方也要進行blanceOF。
3. 進行Import Tokens之前需要到網頁複製。
4. 進行Import Tokens的時候會找不到選單，可遵循下圖找到該項目。

|  |
| --- |
|  |

1. 接著只要貼上address，並點擊下一步即可，成果如下。

|  |
| --- |
|  |

* 使用合約各項功能(須包含多個使用者)

1. 傳送Token的環節，僅需點擊發送。

|  |
| --- |
|  |

1. 設定好金額與接收對象，就能進行轉帳。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體 的圖片  AI 產生的內容可能不正確。 |

* 額外功能 Buy & Burn

1. 在Buy的部分前期與上個小結相同，僅將程式碼改為下圖。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片  AI 產生的內容可能不正確。 |

* View Contract Source code

<https://sepolia.etherscan.io/address/0x9ffe9583727864a7e6bf795e594d3265b5f7b742#code>

1. Buy Token與上述相同，不再贅述。

|  |
| --- |
|  |

* View Token YCToken

<https://sepolia.etherscan.io/token/0x9ffe9583727864a7e6bf795e594d3265b5f7b742>

1. Burn Token與上述相同，不再贅述。

|  |
| --- |
|  |

* View Token YCToken

<https://sepolia.etherscan.io/token/0x9ffe9583727864a7e6bf795e594d3265b5f7b742>

* OnlyOwner相關功能測試，請給出不是Owner的失敗測試。

1. Openzeppelin v4.9.0將程式碼修改為下圖樣式。

|  |
| --- |
|  |

1. 經過如上設定後驗證交易。

|  |
| --- |
|  |

* View Contract Source code

<https://sepolia.etherscan.io/address/0x6502cdd191cf926fbbd78374c7d0fa24ce50a626#code>

* View Transaction

<https://sepolia.etherscan.io/token/0x6502cdd191cf926fbbd78374c7d0fa24ce50a626>

1. 切換到另一帳號，進行Write Contract，由於不是Onwer所以失敗，如下圖。

|  |
| --- |
|  |

* View Transaction Details

<https://sepolia.etherscan.io/tx/0x9ee955a1ca5a3a61ca1e3cd7a097c8137801eb4980d4c823028a1ecd2201e622>

1. Openzeppelin v5.1.0

|  |
| --- |
|  |

* View Contract Source code

<https://sepolia.etherscan.io/address/0x50d6841ad9602e1f50796e031ecb9544bfe8542b#code>

* View Token YCToken

<https://sepolia.etherscan.io/token/0x50d6841ad9602e1f50796e031ecb9544bfe8542b>

1. 切換到另一帳號，進行Write Contract，一樣交易失敗。

|  |
| --- |
|  |
|  |

**Week 03**

Date: March 25, 2025

**Task & Check Points**

|  |  |
| --- | --- |
| **CP** | **Task** |
| ☑ | ERC-721。 |

**Extra Points (Bonus)**

|  |  |
| --- | --- |
| **CP** | **Task** |
| ☑ | ERC-1155 |
| ☑ | ERC 721A |

* ERC-721

1. 建立「YCNFT.sol」，如下圖之程式碼。

|  |
| --- |
|  |

1. 進行Flatten。
2. 進行Complie。
3. 進行Deploy & Run Transactions。
4. Deploy a contract確認。

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片  AI 產生的內容可能不正確。 |

1. 進行驗證與公開交易原始碼。

* View Constact Source Code

<https://sepolia.etherscan.io/address/0x2186e828141a53e9fd15b85efba3fddb38f7d7dc#code>

1. Query tokenURI

|  |
| --- |
|  |

1. 使用OpenSea testnet 查看。

|  |
| --- |
|  |

* OpenSea URL

<https://testnets.opensea.io/assets/sepolia/0x2186E828141A53E9fD15b85EFBa3fDdB38F7D7DC/2>

* ERC-1155

1. 先將程式碼改為以下圖片的形式。

|  |
| --- |
|  |

1. 其他與先前的設定方式相同。

* View Contract Source Code

<https://sepolia.etherscan.io/address/0xd2f7f0f1bd7beb8bd5ba69d4727e0e26b1a4781c#code>

|  |
| --- |
|  |

* OpenSea URL

<https://testnets.opensea.io/assets/sepolia/0xd2f7f0f1bd7beb8bd5ba69d4727e0e26b1a4781c/0>

* ERC-721A

1. 前期須將「YC721A.sol」程式碼修改為下圖的形式。

|  |
| --- |
|  |

1. 進行Flatten。
2. 進行Complie。
3. Deploy & Run Transactoin的部分依據過往形式選擇，但Deploy需要展開選單並且填寫Name和Symbol，如下圖。

|  |
| --- |
|  |

1. 進行Transact。
2. 進行確認。

|  |
| --- |
|  |

* View Contract Source Code

<https://sepolia.etherscan.io/address/0x5ffd210ef39b610444d3b6a9c72470c79acc0ff4#code>

1. 閱讀Contract。

|  |
| --- |
|  |

1. 進行鑄造。

|  |
| --- |
|  |

* View Transaction Details (10 tokens)

<https://sepolia.etherscan.io/tx/0x7b82517da5bf252954554ffd3d7396c44c21ec5318953ad2515c0556db08538f>

* View Transaction (10 tokens)

<https://sepolia.etherscan.io/address/0x2ca3c3f00d2e8e65fdbff1a5120ae6cca39dc593>

* View Transaction Details (1000 tokens)

<https://sepolia.etherscan.io/tx/0xbf927fcb15a011ef4df62450d1e98d1205ceffc964cfbb1a12d7e13609336c4e>

1. 進行空投。

* View Transaction (20 tokens)

<https://sepolia.etherscan.io/tx/0x2523227cfc10dfc63baece5f7cdf7c236da8f81c327b5b473dad1b86f0c5bc61>

* OpenSea URL

<https://testnets.opensea.io/assets/sepolia/0x5ffd210ef39b610444d3b6a9c72470c79acc0ff4/1>

|  |
| --- |
|  |

**Summary & Thoughts**

其實我對於區塊鏈以及數位貨幣的交易就非常感興趣，因過去有相關的私鏈的開發經驗，所以更加的滿心期待，在陳博士的引導中，將許多概念重新建構，對於錢包及冷錢包等是當今滿常見的技術了，在幣圈相當盛行，對於有幸能夠學到，我感到非常的開心。對於整體過程，漸進式且化繁為簡的教學，讓在課堂中無法順利完成的部分，我們也能回家一一品嘗學習，中間也遇到不小插曲，例如在家做的過程，因Google Chrome會同步環境中的擴充套件，例如MetaMask和IPFS，有時雖然在家裡製作，但是會因為同步的過程，導致學校電腦啟動Google Chrome時就被封鎖，不過這也成功讓我學習到如何在學校進行危機自救，這個過程真的值得一提，但就先不贅述，底下就附上相關的佐證，進行留念，這也是研究所修課的奇妙記憶。

|  |
| --- |
| image |
|  |
| image |
| image |

**Reference**

1. MetaMask - Chrome Web Store, <https://chromewebstore.google.com/detail/metamask/nkbihfbeogaeaoehlefnkodbefgpgknn>
2. IPFS, <https://ipfs.tech/>
3. IPFS - Chrome Web Store, <https://chromewebstore.google.com/detail/ipfs-同伴/nibjojkomfdiaoajekhjakgkdhaomnch>
4. Google Cloud Web3 - Ethereum Faucet, <https://cloud.google.com/application/web3/faucet/ethereum>
5. FileBase, <https://filebase.com>
6. Remix - Ethereum IDE, <https://remix.ethereum.org/>
7. OpenSea, <https://testnets.opensea.io/>