ERC-3664 的實現與 DApp-NFT 電子寵物

專題編號：111-CSIE-S026
執行期限：110 年第 1 學期至 111 年第 1 學期
指導教授：劉傳銘
專題參與人員： 108590019 劉彥麟
108590043 李永祺
108590046 蔡禎宸
108820005 洪德易
108820043 洪子翔

一、摘要

此專題主要測試全新的技術，並運用以太坊、智能合約、NFT 等技術實現電子寵物 NFT，而使用 React 搭建 APP 頁面，玩家能夠餵養、販售、裝飾配件在自己的寵物 NFT 上面，此 NFT 最大的特點在於其中的配件可以另外添加，任何人或團體只要經過許可都可能可以為寵物 NFT 設計一個全新的配件。

關鍵詞：Solidity、Smart Contract、Ethereum、NFT、Dapp、Moralis

二、緣由與目的

我們認為 NFT 最大的優勢在於「擁有」，意指他人不可以輕易奪走，透過密碼學、去中心化架構達到此特性，這種特性使得 NFT 可能可以作為一種數位類型的資產，但是現在 NFT 應用僅限在炫富、會員卡、藝術品等功能，有點過於侷限對此技術應用的想像，我們認為 NFT 可以作為未來數位世界中描述物品的最小單位，意指在網路世界中帶有類似於現實物品屬性的東西都可以作為 NFT，然而若以這個觀點下去思考的話，那麼 NFT 也應該與現實物品帶有相同的特性，其中之一便是「組合」，一個物品的組成是由多個配件組合而成，例如手機是由螢幕、晶片、鏡頭等等配件組合而成的，那麼 NFT 應該也要可以達到類似的功能，我們便以此作為基礎，發想出互動屬性更強更適合做應用的 NFT，此專案的 NFT 與傳統 NFT 最大的不同在於，可以藉由智能合約更改其資訊，也可以讓 NFT 去包含其他的 NFT，使其成為一個帶有複合屬性的 NFT，讓其互動性更加的豐富，可更改的資訊包含等級、身上的配件等，且配件都是獨立的 NFT，可再之後進行抽換跟更新。

三、開發工具

(一) 合約

智能合約部分使用 Hardhat[1] 的 Ethereum 測試網 Goerli 上開發，Hardhat 包含功能有測試、部署、連接等等，合約是由 Solidity 進行撰寫，而 unit test 跟 migrate 則是使用 Javascipt 撰寫。

(二)網頁

APP 部分採用 React 開發，使用 JavaScripts 撰寫程式，遊戲後端資料使用 Moralis[2]，Moralis 是一個新興的 web3 整合型服務，包含了鏈上資料的 API 跟資料庫，類似於 Firebase 的功能，使得製作一 Dapp 可以更加簡單。

四、使用技術方法

(一) ERC-721[3]

為市面上 NFT 主流使用協議，由 OpenZeppelin 提供

(二) ERC-3664[4]

針對 ERC721 進行擴充功能的協議，主要用於進行部件的拆分跟合併，由 DRepublic Labs 提供，提供了 NFT 的 attribute 相關操作，attach, increase, separate 等函式。

(三) ERC2981[5]

此為用來設定 NFT 交易版稅的設定，有支援 2981 的交易所會自動調用當中的

function，來得知交易應該抽多少金額並且發送給誰。

(四) 治理合約架構

此為自創的模式，想法是希望 NFT 能作為一個 IP，由一發行商發行 NFT，並允許其他合作夥伴來加入，並用智能合約來確定規則跟治理方式，可將其看為一個粗略版的 DAO。目前當中搭載的功能只有，設定每個會員可得交易手續費的幾%，並可以依照前面的設定去分配治理合約中所含有的金額。

(五) Chainlink VRF V2

此為 Chainlink 項目方製作，主要為產生一個較為公平且不容易被預測的隨機數。

(六) APP 與智能合約的互動

玩家在遊玩遊戲時，當需要調用合約時就會透過 Moralis 的 API，設定對應的合約地址以及合約的 method，將資料傳送到以太坊虛擬機(EVM)進行區塊鏈的資料交換，再經由後端跟資料庫進行遊戲資訊的管理，最後顯示在 APP 上。

(七) 遊戲資料存放

為了實現透過調用智能合約上相同的資料，建造多元的遊戲環境,我們只將遊戲內較重要的資料儲存在區塊鏈上，剩餘的資訊則採取中心化伺服器，透過 Moralis API 進行傳遞。

五、問題及解決辦法

(一) 資料存放問題

若是將所有資料都放在 Metadata 跟合約中，鏈上將會負擔過重，會導致 Gas 過高以及讀取過慢的問題，因此我們決定將部分資料存放在資料庫當中，例如飽食度、親密度、寵物暱稱、部分圖片，藉由犧牲部分去中心化來加快讀取上的速度以及減低 Gas 的消耗，改善用戶的體驗。

(二) 拆分後的圖片問題

NFT 一旦可以進行部件拆分後，顯示圖片也必須跟著變動，然而存放 NFT 資料 Metadata 在鑄造後就相當難改動，而且組裝後 NFT 的圖片沒辦法是先存放進去，目前只能透過即時組圖的方式去做更動，且合約上 tokenURI()也必須去做更新，因此，只能由合約中的管理員做代理去操作，避免用戶隨意設定 token 的 URI，但是這麼做的問題在於去中心化的程度降低，因此如何拿捏安全跟中心化之間的平衡，還仍需再設計。

(三) 治理合約分配問題

現在的設計模式只能讓所有會員按照先前設定的比例去分配合約中的 token，但是這就造成可能會有設定比例跟貢獻不成正比的狀況，而後續也沒辦法自行更改。因此，能否有更好的治理模式可以吸引更多人願意貢獻自己的心力，讓整個 NFT 生態更加活躍，我們仍需思考。

六、結論

現在的 NFT 以及區塊鏈是相當不成熟的產業，當中有非常多的問題需要去解決，隨著人們的生活更多是在網路上，那藉由去中心化跟密碼學來去保障自己在網路上的資產或資料的這點，可能會顯得越來越重要，現在我們認為在保護上已經做到了不錯，但如何在用戶體驗、應用場景、處理效能跟去中心化之間拿捏平衡，以及一些必要的治理跟完全的控管、統治如何劃分清楚，我想只有我們繼續往前才能將這些問題釐清。

參考文獻

[1] Hardhat Documentations: https://hardhat.org/docs

[2] Moralis: https://docs.moralis.io/

[3]OpenZeppelin ERC-721 Docs: https://docs.openzeppelin.com/contracts/4.x/erc721

[4] DRepublic Labs ERC-3664 Source code: https://github.com/DRepublic-io/EIP-3664

[5] EIP-2981: NFT Royalty Standard: https://eips.ethereum.org/EIPS/eip-2981