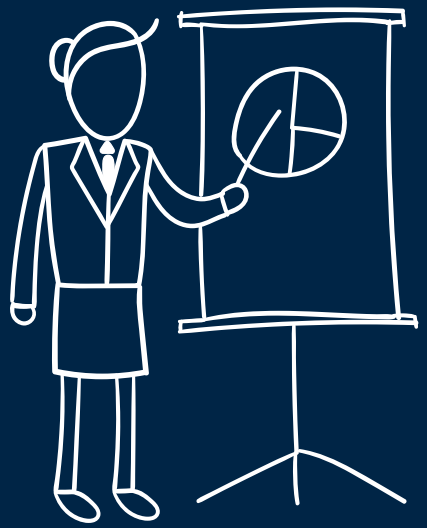
# 火柴人滑雪遊戲



經管四 107570047 吳東翰
資財碩一 110AB8047 張兆宜
資財三甲 108AB0015 詹少鋐
資財三甲 108AB0037 楊詔棋
資財三甲 108AB0044 劉禮惟
臺灣海洋大學 00957118 蔡翔宇

1

遊戲架構

2

網頁

3

合約程式碼

4

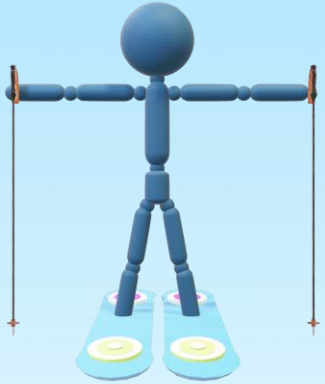
遊戲展示

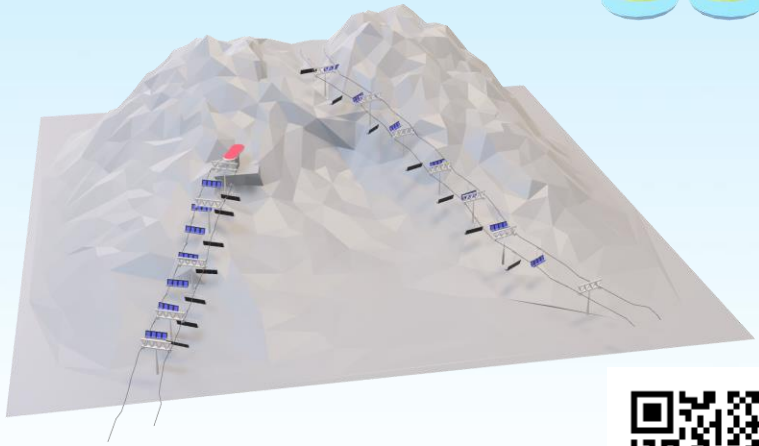

# 遊戲架構

# 滑雪

Ski

滑雪是指利用滑雪板在雪地滑行的一種活動，最初是為了便於在冬季的雪地中出行，後來逐漸演變成一種冬季運動項目。滑雪和滑冰、滑水有相似之處，但是滑的表面不同。滑雪的英文名詞「ski」本是一個挪威語詞彙，源於古挪威語「skid」，意為小段木頭。





彩蛋

玩法從螢幕上方移動到下面
堅持久一點 可以拿到代幣跟NFT

全部會出現的滑雪人滑雪板 顏色


# 網頁

CONNECT WALLET

# SkiFree Web3 Ver.

**Connect Wallet to Start**

Edited by Horden Chan

玩遊戲前要connect wallet

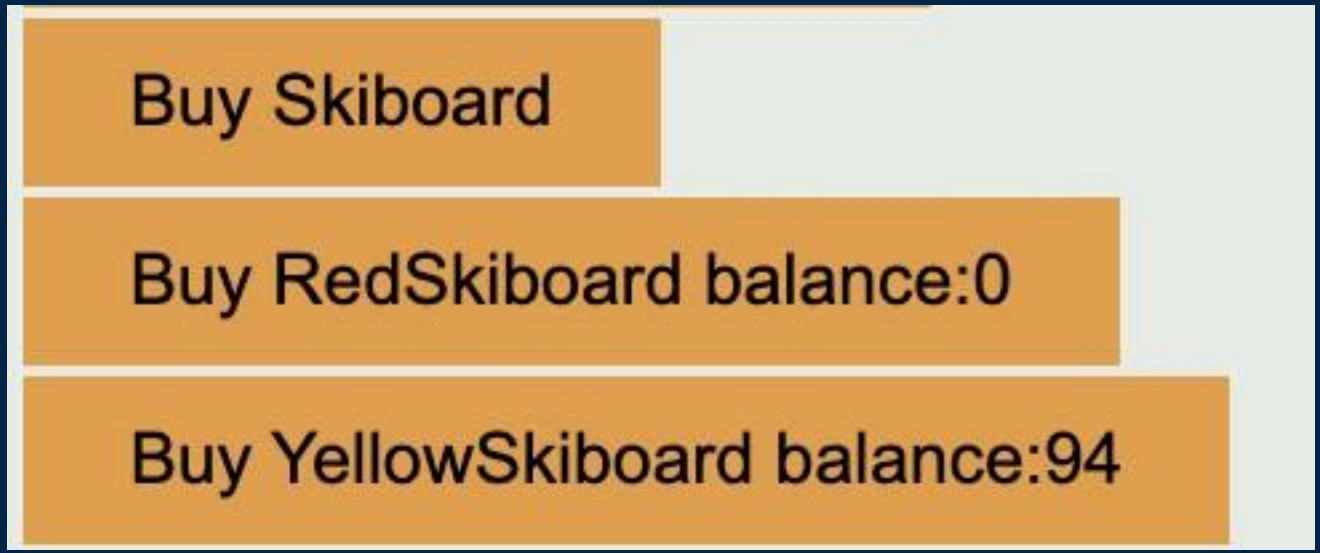
Buy Token balance:11

Buy Skiboard


#0


#1

可以買滑雪板
買Token


Buy Skiboard
Buy RedSkiboard balance:0
Buy YellowSkiboard balance:94

滑雪板數量

Buy RedSkiboard balance:0

Buy YellowSkiboard balance:94


No skiboard


Yellow skiboard

選擇你要的滑雪板開局

Buy Token balance:22

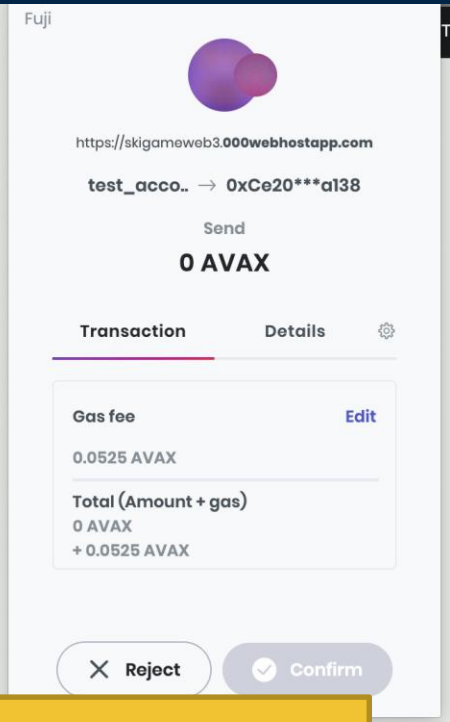
Buy Skiboard

No skiboard

Yellow skiboard

# SkiFree Web3 Ver.

Edited by Horden Chan




開始遊戲要付Token和滑雪板

# SkiFree Web3 Ver.


Fuji

https://skigameweb3.000webhostapp.com

test_acco.. → 0xCe20***a138

Send

0 AVAX

Transaction Details

Gas fee Edit

0.0525 AVAX

Total (Amount + gas)
0 AVAX
+ 0.0525 AVAX

遊戲結束得到Token

獎勵對照

0.01乙太 100 滑雪板

0.01乙太10token

200分一token

SkiFree Web3 Ver.

不同滑雪板會有不同的加成
黃色1.25倍Token
藍色沒用滑雪板

角色顏色(速度):
黃色>紅色>藍色

SkiFree Web3 Ver.


# 合約程式碼

```
contract ERC20 {
    //uint256 private totalSupply;
    address public owner;
    event Transfer(address indexed _from, address indexed _to, uint256 _value);//Transfer event , active it when the token is being transfer
    event Approval(address indexed _owner, address indexed _spender, uint256 _value);//Approval event , active it when succesfully execute "approve" method
    uint256 constant private MAX_UINT256 = 2**256 - 1;
    mapping (address => uint256) private balances;//建立一個address映射到uint256類別balances,顯示該address帳戶餘額
    mapping (address => mapping (address => uint256)) private allowed;//建立一個address映射到address,uint256類別allowed,顯示該帳戶允許哪個帳戶操作他多少金額
    string private _name;
    string private _symbol;
    uint8 private _decimal=0;//小數位為0
    uint256 private totalSupplyAmount=10000;//10000 tokens in total
```

```
constructor(string memory name_, string memory symbol_) public{//constructor
    _name = name_;
    _symbol = symbol_;
    owner=msg.sender;
    balances[owner]=totalSupplyAmount;//將總共的10000沒tokens都給initial owner
    emit Transfer(address(0), owner, 10000);
}
```

```
function buyToken(address sender)public payable
{
    require( msg.value == 0.01 ether, "0.01 ETH");
    erc20.ownerTransfer(sender,10);
}
function getToken(address sender,uint256 score)public
{
    require((PlayTimes[msg.sender][block.timestamp/86400])<=10);//reruire當日遊玩次數超過10次的
    uint256 prize=score/200;//1000分換一代幣
    erc20.ownerTransfer(sender,prize);
}
```

買代幣或用分數換代幣

```
function startGame(address sender,uint256 skiboardId) public returns(bool success)//資產滑雪板，nft人物，滑雪板
{

    require((PlayTimes[msg.sender][block.timestamp/86400])<10);//reruire當日遊玩次數超過10次的
    PlayTimes[msg.sender][block.timestamp/86400]+=1;

    erc20.TransferToOwner(sender,1);//燒掉滑雪板，確認他有nft
    if(skiboardId!=0)
    {
        require(erc1155.balanceOf(sender,skiboardId)>0);
        erc1155.TransferToOwner(sender,skiboardId,1);//burn the skiboard
    }
    //TODO
}
function buySkiBoard(address sender,uint256 skiboardId)public payable returns(bool success)
{
    require( msg.value == 0.01 ether, "0.01 ETH");
    erc1155.ownerTransfer(sender,skiboardId,100);
}
```

燒滑雪板
買滑雪板

```
function erc20name()public view returns(string memory)
{
    return erc20.name();
}
function erc20symbol()public view returns(string memory)//the function which returns tokens symbol
{
    return erc20.symbol();
}
function erc20decimals()public view returns(uint8)//the function which returns the decimal (0)
{
    return erc20.decimals();
}
function erc20totalSupply()public view returns(uint256)
{
    return erc20.totalSupply();
}
function erc20transfer(address _to, uint256 _value) public returns (bool success)
{
    return erc20.transfer(_to,_value);
}
function erc20ownerTransfer(address _to,uint256 _value)public returns(bool success)
{
    return erc20.ownerTransfer(_to,_value);
}
//function erc20transferFrom(address _from, address _to, uint256 _value) public returns (bool success);
function erc20balanceOf(address _owner) public view returns (uint256 balance)
{
    return erc20.balanceOf(_owner);
}
```

ERC20

```
function erc1155ownerTransfer(address _to, uint256 _id, uint256 _value) external
{
    erc1155.ownerTransfer( _to, _id,_value);
}
function erc1155TransferFrom(address _from, address _to, uint256 _id, uint256 _value) external
{
    erc1155.TransferFrom(_from,_to,_id,_value);
}
/*function erc1155safeBatchTransferFrom(address _from, address _to, uint256[] calldata _ids, uint256[] calldata _values) external
{
    bytes temp=0x21;
    erc1155.safeBatchTransferFrom(_from,_to,_ids, _values,temp);
}*/
function erc1155balanceOf(address _owner, uint256 _id) external view returns (uint256)
{
    return erc1155.balanceOf(_owner,_id);
}
function erc1155balanceOfBatch(address[] calldata _owners, uint256[] calldata _ids) external view returns (uint256[] memory)
{
    return erc1155.balanceOfBatch(_owners,_ids);
}
function erc1155create(uint256 _initialSupply, string calldata _uri) external returns(uint256 _id)
{
    return erc1155.create(_initialSupply,_uri);
}
/*function erc1155mint(uint256 _id, address[] calldata _to, uint256[] calldata _quantities) external
{
    erc1155.mint(_id,_to,_quantities);
}*/
function erc1155setURI(string calldata _uri, uint256 _id) external
{
    erc1155.setURI(_uri,_id);
}
//function erc1155setApprovalForAll(address _operator, bool _approved) external
//function erc1155isApprovedForAll(address _owner, address _operator) external view returns (bool);
//ERC1155 ends
```

ERC1155

# 分工表

## 吳東翰
火柴人3D
美術
PPT一部分

## 詹少鋐
網頁遊戲撰寫及設計
合約串接
部分合約修改

## 張兆宜
錢包連接
找到滑雪遊戲程式碼
滑雪板顏色

## 蔡翔宇
Erc20
erc1155gamestatus
合約程式碼

## 楊詔棋
ERC721
合約設計撰寫

## 劉禮惟
PPT一部分
報告

4

# 遊戲展示

# 合約地址&原始碼

ERC721合約(Avax fuji)

0x55664eff32125F9Cbe148cAD3325f230608D950a

ERC20 ERC1155 遊戲狀態整合合約(Avax fuji)

0xCe20521056143a84DFBbBA2dB2Ca8e524893a138

原始碼

https://github.com/horden316/Skifree-web3