

A stylized graphic featuring a central teal-colored Earth showing the continents of North and South America. It is surrounded by two concentric teal circular orbits. Four teal circles of varying sizes are positioned on these orbits, representing planets or moons. The background is a dark navy blue, filled with numerous white stars of different sizes and shapes, some appearing as simple dots and others as multi-pointed stars.

Super-Earth Real Estate

SMU FinTech Team 3

Introduction

- Auction land on the new super-Earth planet discovered recently and make available for purchase via Ethereum token.
- Register and update the star land registry dApp, pin an star land File to IPFS via Pinata.
- All property contracts are encrypted and tokenized on Solidity Remix IDE

Our Team



Allyssa
Carmin



Vicky
Lee



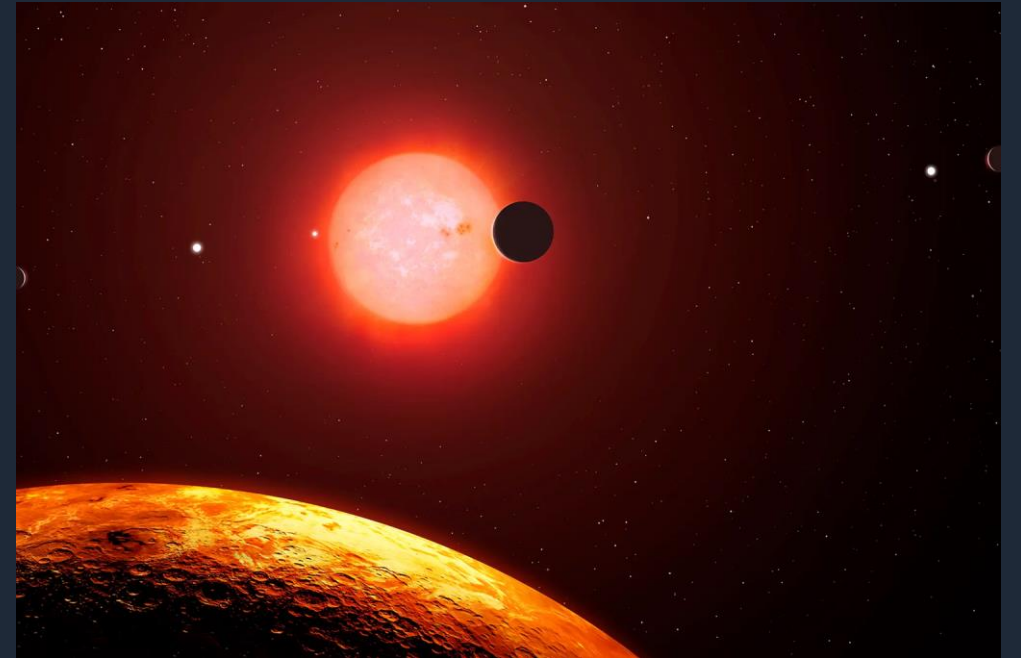
Julian
Loudon



Noman
Zubairi

What is LP 890-9c?

- Also known as SPECULOOS-2 or TOI-4306, is a high proper motion red dwarf star located 105 light-years away from the Solar System in the constellation of Eridanus.
- It is the second-coolest star found to host a planetary system, after TRAPPIST-1.



Current Property Offerings

Whether you'll eventually visit the Super-Earth as a tourist, or to do scientific research, or to mine valuable rare-earth elements, the new Super Earth which is 3 times the size of earth offers a wealth of fascinating locations with high value mineral & land rights.

Top Listings:

- Diamond Land “Girls Best Friend”
- Oil & Gas Land Rights
- Waterloo “Infinite water spring rights”
- Gold Heaven
- Ti22 “Titanium Overload”

Book your Journey to Super-Earth Today!

- Not certain which region to select? Click on the region name to access a detailed description and location information. It's that easy to buy Super-Earth property!
- WE DELIVER EVERYWHERE AROUND THE WORLD!
- All property prices are stated in Ethereum per acre based on current exchange rates.



Encrypted and Secure (NFT) Contracts

Your property ownership contract is encrypted and tokenized on the Solidity Smart Chain, and is accessible only through your secure, private portal – an exclusive feature available only to our clients!



Super-Earth property Prime Real Estate

- This artist conception shows a young, hypothetical planet around a cool star. A soupy mix of potentially life-forming chemicals can be seen pooling around the base of the jagged rocks.



Super-Earth property Prime Real Estate

- This artist's concept allows us to imagine what it would be like to stand on the surface of the exoplanet TRAPPIST-1F.



Demo - Back-End (Smart Contract)

```
pragma solidity ^0.5.17;

import "https://github.com/OpenZeppelin/openzeppelin-contracts/blob/release-v2.5.0/contracts/token/ERC721/ERC721Full.sol";

contract Certificate is ERC721Full {
    constructor() public ERC721Full("LandToken", "LAND") {}

    function purchaseLand(address buyer, string memory tokenURI)
        public
        returns (uint256)
    {
        uint256 newLandId = totalSupply();
        _mint(buyer, newLandId);
        _setTokenURI(newLandId, tokenURI);

        return newLandId;
    }
}
```

Demo - Front-End (User Interface)

```
# Imports
import json
import os
from pathlib import Path
import streamlit as st
from dataclasses import dataclass
from typing import Any, List
from web3 import Web3
w3 = Web3(Web3.HTTPProvider('HTTP://127.0.0.1:7545'))

from wallet import generate_account, get_balance, buy_land

# Cache the contract on load
@st.cache(allow_output_mutation=True)
# Define the load_contract function
def load_contract():

    # Load ABI
    with open(Path('contracts\compiled\land_abi.json')) as f:
        land_abi = json.load(f)

    # Set the contract address (this is the address of the deployed contract)
    contract_address = os.getenv("SMART_CONTRACT_ADDRESS")

    # Get the contract
    contract = w3.eth.contract(
        address=contract_address,
        abi=land_abi
    )

    # Return the contract from the function
    return contract

# Load the contract
contract = load_contract()

# Database of lands available for sale
land_database = {
    "Land 1": ["Land 1", .25, "0xaBa6f43b2Bae43Ff7C697aB5A03847796d4A6afB", "images\Land1.jpg"],
    "Land 2": ["Land 2", .15, "0x45F7B55bB08f8665Dc8E294D412dCC238fb7F00E", "images\Land2.jpg"],
    "Land 3": ["Land 3", .43, "0xD1EC080bE92EC2175ea60E64FD835f44ef7Ba3d8", "images\Land3.jpg"],
    "Land 4": ["Land 4", .17, "0xa7808b6a7d45fc8f46F149Ec4B98fEc697eF0e61", "images\Land4.jpg"],
    "Land 5": ["Land 5", .20, "0x91d409a2910e352970965C0c64AE4dc6886E0bb6", "images\Land5.jpg"]
}

lands = ["Land 1", "Land 2", "Land 3", "Land 4", "Land 5"]
```

```
def get_land():
    db_list = list(land_database.values())

    for number in range(len(lands)):
        st.image(db_list[number][3], width=400)
        st.text(" \n")
        st.write("Name of land: ", db_list[number][0])
        st.write("Price of land: ", db_list[number][1])
        st.text(" \n")

# Streamlit code
st.markdown("# Buy land on Super Earth!")
st.text(" \n")

get_land()

accounts = w3.eth.accounts
account = accounts[0]

st.sidebar.markdown("## Purchase land on Super Earth with Ether!")

st.sidebar.write(account)

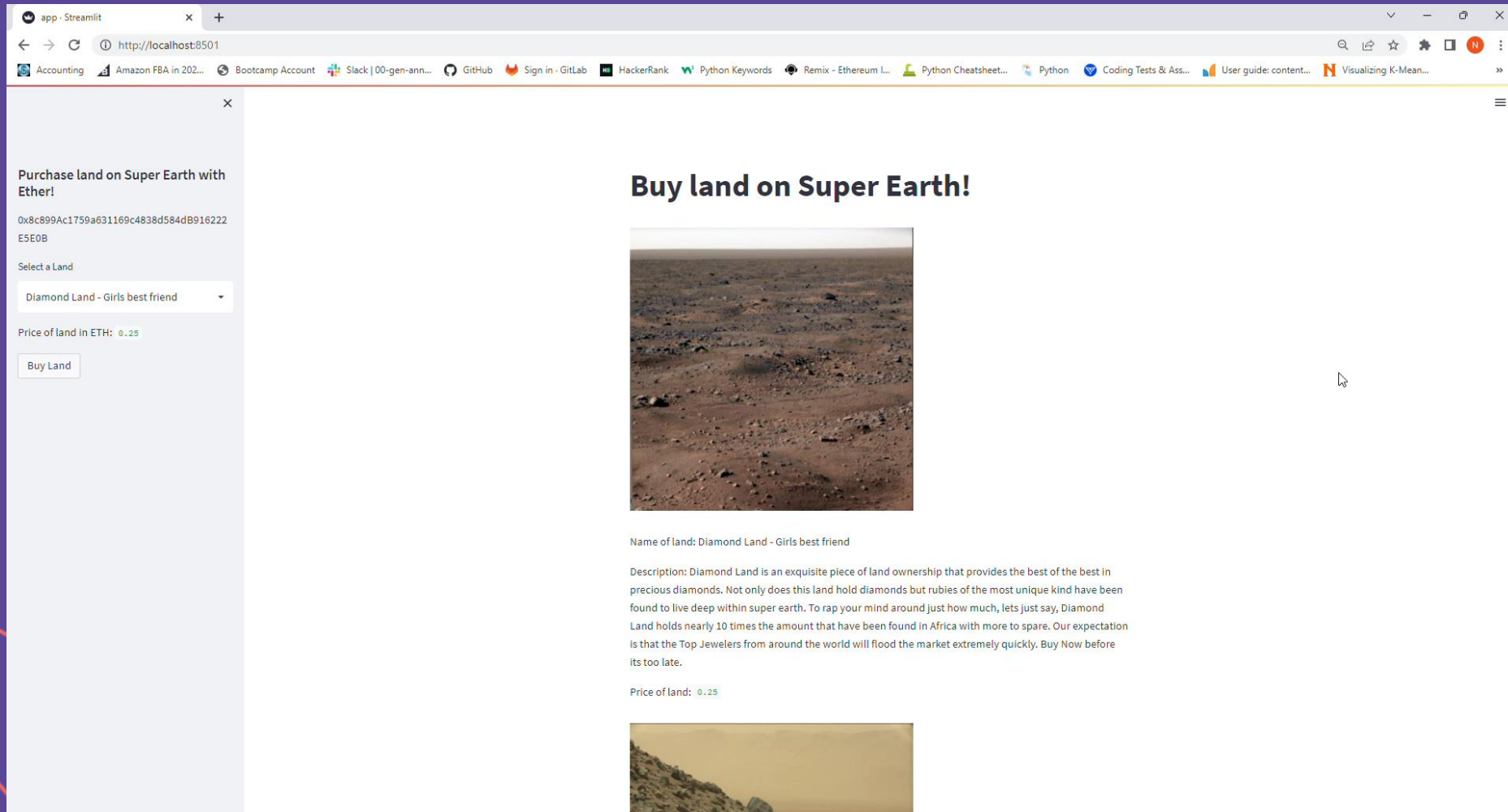
land = st.sidebar.selectbox('Select a Land', lands)

price = land_database[land][1]

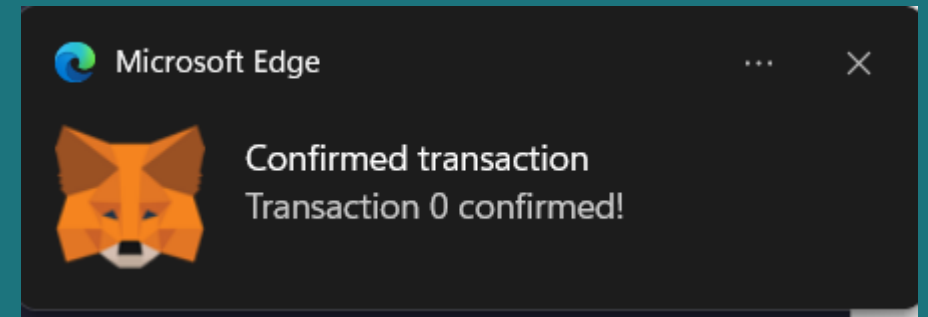
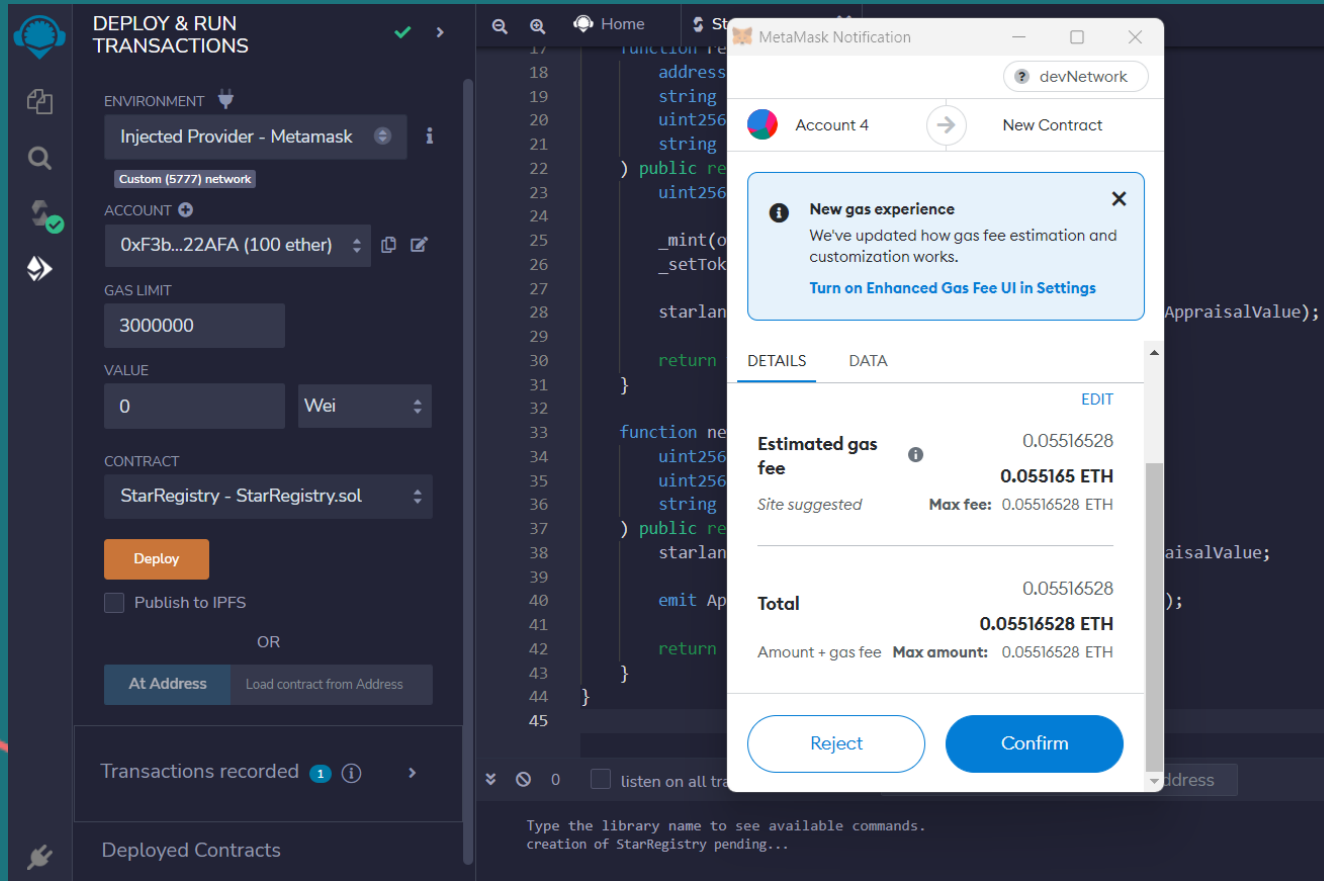
st.sidebar.write("Price of land in ETH:", price)

buyer_account = land_database[land][2]
new_land = land
if st.sidebar.button("Buy Land"):
    contract.functions.purchaseLand(buyer_account, new_land).transact({'from': buyer_account, 'gas': 1000000})
    st.balloons()
    st.text(" \n")
    st.text(" \n")
    st.sidebar.write("Congratulations! You now own land on Super Earth!")
```

Demo - Streamlit



Star Land Registry & Appraisal System



You can now view your Streamlit app in your browser.

Local URL: `http://localhost:8501`
Network URL: `http://10.0.0.84:8501`

Demo - Streamlit

Star Land Registry Appraisal System

Choose an account to get started

Select Account

0xA95A963F40635Dfc5f4471752806971E61ed9C06

Register New Star Land

Enter the name of the star land

Enter the initial appraisal amount

Upload Star Land



Drag and drop file here

Limit 200MB per file • JPG, JPEG, PNG

Browse files

Register Star Land

Appraise Star Land

Choose a Star Land Token ID

No options to select.

Enter the new appraisal amount

Enter details for the Appraisal Report

Appraise Star Land

Get the appraisal report history

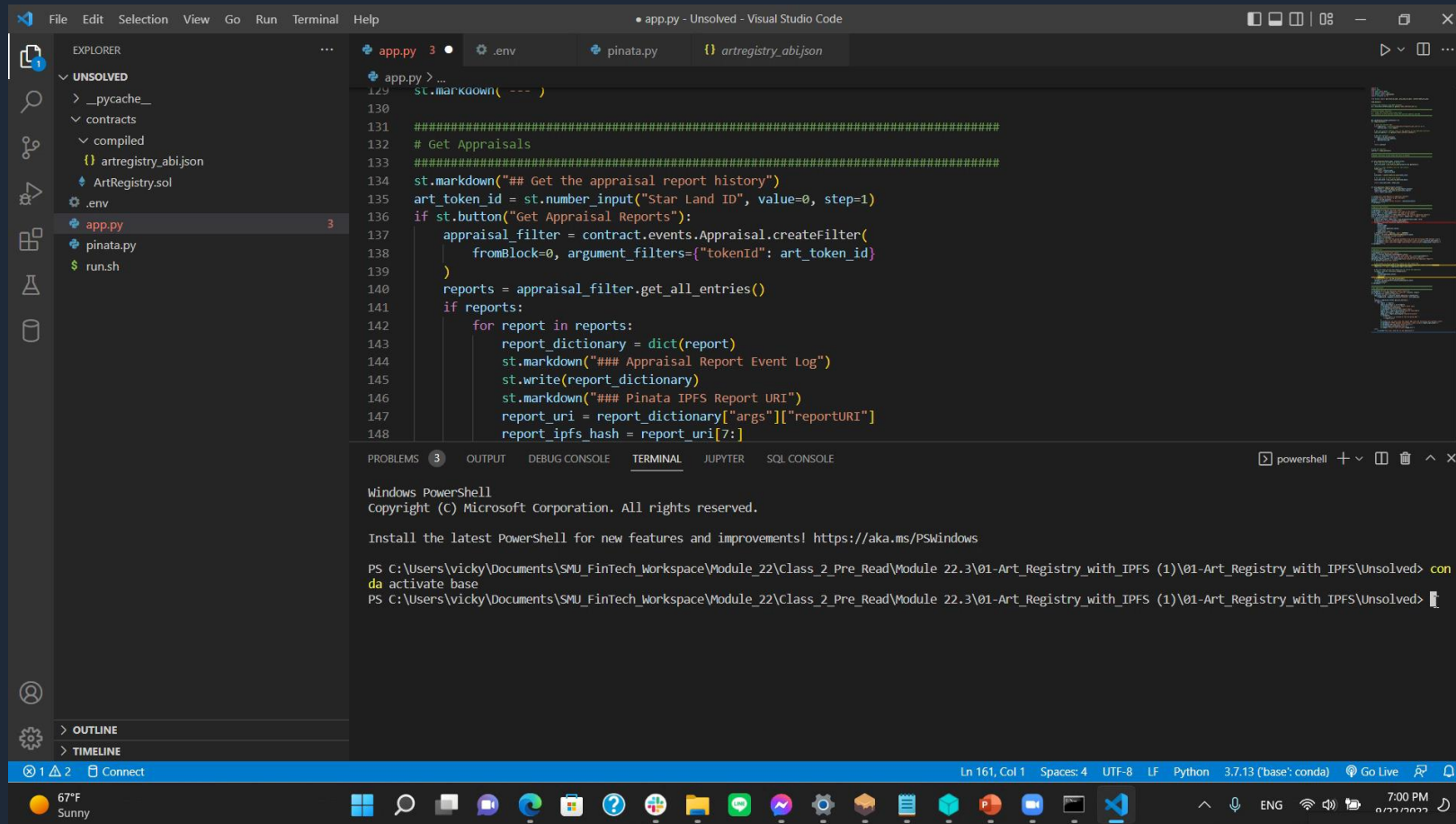
Star Land ID

0

- +

Get Appraisal Reports

Demo



The screenshot displays the Visual Studio Code interface. The Explorer sidebar on the left shows a project structure with folders like `__pycache__`, `contracts`, and `compiled`, and files like `artregistry_abi.json`, `ArtRegistry.sol`, `.env`, `app.py`, `pinata.py`, and `run.sh`. The `app.py` file is open in the editor, showing Python code that interacts with a blockchain contract. The code includes comments and function calls like `st.markdown`, `st.number_input`, `contract.events.Appraisal.createFilter`, and `appraisal_filter.get_all_entries`. The Terminal window at the bottom shows a Windows PowerShell prompt with the command `conda activate base` and the output `PS C:\Users\vicky\Documents\SMU_FinTech_Workspace\Module_22\Class_2_Pre_Read\Module_22.3\01-Art_Registry_with_IPFS (1)\01-Art_Registry_with_IPFS\Unsolvd> conda activate base`. The status bar at the bottom indicates the current file is `Ln 161, Col 1`, the encoding is `UTF-8`, and the language is `Python`.

```
129 st.markdown( --- )
130
131 #####
132 # Get Appraisals
133 #####
134 st.markdown("## Get the appraisal report history")
135 art_token_id = st.number_input("Star Land ID", value=0, step=1)
136 if st.button("Get Appraisal Reports"):
137     appraisal_filter = contract.events.Appraisal.createFilter(
138         fromBlock=0, argument_filters={"tokenId": art_token_id}
139     )
140     reports = appraisal_filter.get_all_entries()
141     if reports:
142         for report in reports:
143             report_dictionary = dict(report)
144             st.markdown("### Appraisal Report Event Log")
145             st.write(report_dictionary)
146             st.markdown("### Pinata IPFS Report URI")
147             report_uri = report_dictionary["args"]["reportURI"]
148             report_ipfs_hash = report_uri[7:]
```

Windows PowerShell
Copyright (c) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

PS C:\Users\vicky\Documents\SMU_FinTech_Workspace\Module_22\Class_2_Pre_Read\Module_22.3\01-Art_Registry_with_IPFS (1)\01-Art_Registry_with_IPFS\Unsolvd> conda activate base
PS C:\Users\vicky\Documents\SMU_FinTech_Workspace\Module_22\Class_2_Pre_Read\Module_22.3\01-Art_Registry_with_IPFS (1)\01-Art_Registry_with_IPFS\Unsolvd>

67°F Sunny 7:00 PM

Product Roadmap



MILESTONE

Front End –
User Interface

MILESTONE

Adding Auction
Functionality

MILESTONE

Expanding Property
Offerings as we
explore more Super-
Earth

MILESTONE

Tickets to the Future!

■ Questions?

Ready to get started?

Choose your landing site on the Super-Earth!

