.

**PROJECT REPORT**

**ON**

NFT GENERATOR

( PRODUCT BASED NFT GENERATOR)

**Of**

**Bachelor of Technology**

**In**

**COMPUTER SCIENCE & ENGINEERING**



**Submitted By ::--**

**KARAN SINGH BISHT**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING ECHELON INSTITUTE OF TECHNOLOGY, FARIDABAD**

**APRIL 2022 – AUGUST 2022**



**CERTIFICATE**

I/We hereby certify that the work which is being presented in the B.Tech. Project Report entitled NFT GENERATOR for the award of the Bachelor of Technology in Computer Science & Engineering and submitted to the Department of Computer Science & Engineering of Echelon Institute of Technology, Faridabad is an authentic record of my own work carried out during a period from January 2022 to May 2022.

The matter presented in this thesis has not been submitted by me for the award of any other degree elsewhere.

Signature of Candidate

KARAN SINGH BISHT ( 19-CSE-045)

**TO WHOM IT MAY CONCERN**

This is to certify that the Project entitled “**NFT GENERATOR”** submitted by “**KARAN SINGH BISHT** ” (19- CSE- 045) Department of Computer Science and Engineering, Echelon Institute of Technology Under YMCA University, Faridabad, for partial fulfillment of the requirements for the degree of Bachelor of Technology in Computer Science & Engineering; is a bonafide record of the work and investigations carried out by him under my supervision and guidance

Signature of HOD

Ms. Shefali Madan

Head of Department

Signature of the Supervisor

Ms. Jagrati Malviya

Assistant Professor



**ACKNOWLEDGEMENTS**

I take this opportunity to thank all those who have helped me in completing the project successfully.

I would like to express my gratitude to **Ms. Jagrati Malviya**, who as my guide/mentor provided me with every possible support and guidance throughout the development of the project. This project would never have been completed without her encouragement and support.

I would also like to show my gratitude to **Ms. Shefali Madan,** Head of Department for providing us with well-trained Team members and giving us all the required resources and a healthy environment for carrying out our project work.

KARAN SINGH BISHT



**ABSTRACT**

NFT (non - fungible token ) Generator it is an offline software based on using **python algorithms** and creating digital artwork with it , this type of artwork is also known as recursive artwork or algorithmic artwork . Which is nowadays a trending topic and after the creation of this artwork we can upload it with the **ERC 721 contract** which is paid or we can use a free service like opensea and upload our created artworks.

This works on simple algorithm of creating random dots and connecting these random dots by which the result is a beautiful artwork , there are a lot of different artwork created like this one famous algorithm is the mandelbrot set and the result is the mandelbulb these kind of **NFTs sells** from somewhere 20,000 INR to 65,000 INR just a single piece of work and with **NFT generator** you can create 100s of random artwork at once .



**TABLE OF CONTENTS**

**I Certificate II Candidate declaration III Acknowledgement IV Abstract**

**Chapter 1: Introduction to Project 7-10**

**Chapter 2: Requirement Analysis 11**

**Chapter 3: Language and Libraries used 12-14**

**Chapter 4: Implementation of components and functions 15-19**

**Chapter 5: Information about Testing 20-25**

**Chapter 6: Conclusion 26-27**

**Chapter 7: References 28-29**

**Chapter 8: Brief Profile 30-31**



**CHAPTER 01**

**INTRODUCTION TO PROJECT**



**Chapter 1**

**INTRODUCTION TO PROJECT**

**WHAT IS NFT GENERATOR ?**

NFT (non - fungible token ) Generator it is an offline software based on using **python algorithms** and creating digital artwork with it , this type of artwork is also known as recursive artwork or algorithmic artwork . Which is nowadays a trending topic and after the creation of this artwork we can upload it with the **ERC 721 contract** which is paid or we can use a free service like opensea and upload our created artworks.

This works on simple algorithm of creating random dots and connecting these random dots by which the result is a beautiful artwork , there are a lot of different artwork created like this one famous algorithm is the mandelbrot set and the result is the mandelbulb these kind of **NFTs sells** from somewhere 20,000 INR to 65,000 INR just a single piece of work and with **NFT generator** you can create 100s of random artwork at once .

Algorithmic art, also known as computer-generated art, is a subset of generative art (generated by an autonomous system) and is related to systems art (influenced by systems theory). Fractal art is an example of algorithmic art.

For an image of reasonable size, even the simplest algorithms require too much calculation for manual execution to be practical, and they are thus executed on either a single computer or on a cluster of computers. The final output is typically displayed on a computer monitor, printed with a raster-type printer, or drawn using a plotter. Variability can be introduced by using pseudo-random numbers. There is no consensus as to whether the product of an algorithm that operates on an existing image (or on any input other than pseudo-random numbers) can still be considered computer-generated art, as opposed to computer-assisted art.

***Role of the algorithm***

Letter Field by Judson Rosebush, 1978. Calcomp plotter computer output with liquid inks on rag paper, 15.25 x 21 inches. This image was created using an early version of what became Digital Effects' Vision software, in APL and Fortran on an IBM 370/158. A database of the Souvenir font; random number generation, a statistical basis to determine letter size, color, and position; and a hidden line algorithm combine to produce this scan line raster image, output to a plotter.

From one point of view, for a work of art to be considered algorithmic art, its creation must include a process based on an algorithm devised by the artist. Here, an algorithm is simply a detailed recipe for the design and possibly execution of an artwork, which may include computer code, functions, expressions, or other input which ultimately determines the form the art will take. This input may be mathematical, computational, or generative in nature. Inasmuch as algorithms tend to be deterministic, meaning that their repeated execution would always result in the production of identical artworks, some external factor is usually introduced. This can either be a random number generator of some sort, or an external body of data (which can range from recorded heartbeats to frames of a movie.) Some artists also work with organically based gestural input which is then modified by an algorithm. By this definition, fractals made by a fractal program are not art, as humans are not involved. However, defined differently, algorithmic art can be seen to include fractal art, as well as other varieties such as those using genetic algorithms. The artist Kerry Mitchell stated in his 1999 Fractal Art Manifesto.

*Fractal Art is not..Computer(ized) Art, in the sense that the computer does all the work. The work is executed on a computer, but only at the direction of the artist. Turn a computer on and leave it alone for an hour. When you come back, no art will have been generated.*

***Role of NFT’s***

NFT means non-fungible tokens (NFTs), which are generally created using the same type of programming used for cryptocurrencies. In simple terms these cryptographic assets are based on blockchain technology. They cannot be exchanged or traded equivalently like other cryptographic assets.

A non-fungible token is a financial security consisting of digital data stored in a blockchain, a form of distributed ledger. The ownership of an NFT is recorded in the blockchain, and can be transferred by the owner, allowing NFTs to be sold and traded.

Reflecting the growing interest in the nonfungible token (NFT) space among today's investors, new research predicts that the existing $3 billion market size will reach $13.6 billion by the end of 2027.



**CHAPTER 02**

**REQUIREMENT ANALYSIS**



**Chapter 2**

**REQUIREMENT ANALYSIS**

**Software and Hardware Requirements:**

A Software requirements specification (SRS) document describes the intended purpose,requirements, and nature of software/application/project to be developed.

1) SUBLIME - to write the code

2) PYTHON Programming Language

3) OpenAI to prompt and name then

4) ERC 721 smart contract

5) Open sea - polygon matic contract

6) REACT to show images

7) Pillow for images

8) OS :- Windows/ MacOs / Linux

9) A good CPU (preferably 6 cores)

10) Good amount of RAM (minimum 2 GB)

11) knowledge of basic programming for problem solving



**CHAPTER 03**

**LANGUAGE AND LIBRARIES USED**



**PYTHON:**

**What is PYTHON?**

**Chapter 03**

**Language and Libraries used**

Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn’t specialized for any specific problems. This versatility, along with its beginner-friendliness, has made it one of the most-used programming languages today. A survey conducted by industry analyst firm RedMonk found that it was the second-most popular programming language among developers in 2021

Python is commonly used for developing websites and software, task automation, data analysis, and data visualization. Since it’s relatively easy to learn, Python has been adopted by many non-programmers such as accountants and scientists, for a variety of everyday tasks, like organizing finances.

**ERC 721:**

**What is ERC 721 (SMART CONTRACT)?**

Firstly, ERC-721 is a type of standard — a template or format that other developers agree to follow. Following the same standards makes writing code easier, more predictable, and reusable. These standards are completely voluntary, but following a widely used standard means compatibility with a wide variety of applications including exchanges, dapps, and wallets.ERC-721 is a token standard on Ethereum for non-fungible tokens (NFTs). Fungible means interchangeable and replaceable; Bitcoin is fungible because any Bitcoin can replace any other Bitcoin. Each NFT, on the other hand, is completely unique. One NFT cannot replace another.The initial ERC-721 specification was proposed by Dieter Shirley as an Ethereum Improvement Proposal (EIP), which is a process for introducing new standards to Ethereum.

Anyone can submit an EIP, but it goes through a process of review and revision before it is accepted by the community. Once accepted, the EIP moves to an Ethereum Request for Comments (ERC), which is a standard process for Ethereum applications. The official authors of the ERC-721 standard are William Entriken, Dieter Shirley, Jacob Evans, and Nastassia Sachs.

**PILLOW:**

**What is PILLOW?**

Python Imaging Library (expansion of PIL) is the de facto image processing package for Python language. It incorporates lightweight image processing tools that aids in editing, creating and saving images. Support for Python Imaging Library got discontinued in 2011, but a project named pillow forked the original PIL project and added Python3.x support to it. Pillow was announced as a replacement for PIL for future usage. Pillow supports a large number of image file formats including BMP, PNG, JPEG, and TIFF. The library encourages adding support for newer formats in the library by creating new file decoders.

This module is not preloaded with Python. So to install it execute the following command in the command-line:

– pip install pillow



**CHAPTER 4**

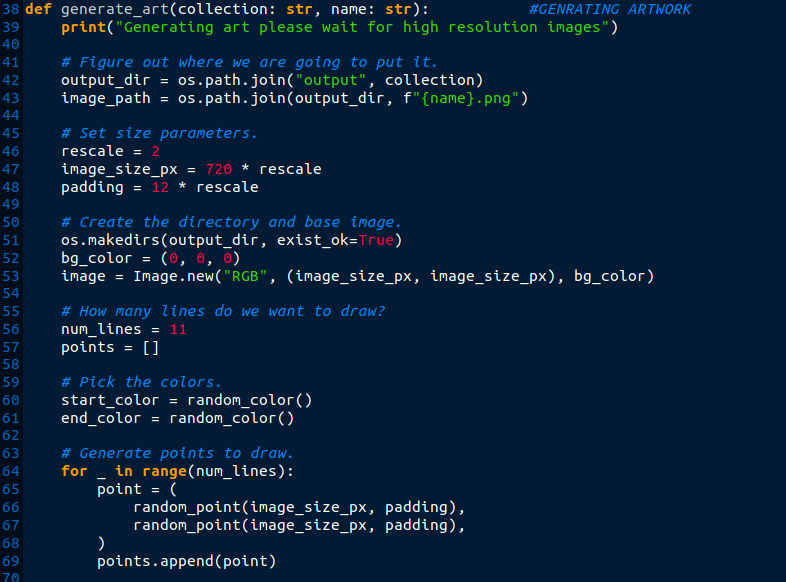
**IMPLEMENTATION OF**

**COMPONENTS And**

**FUNCTIONS**

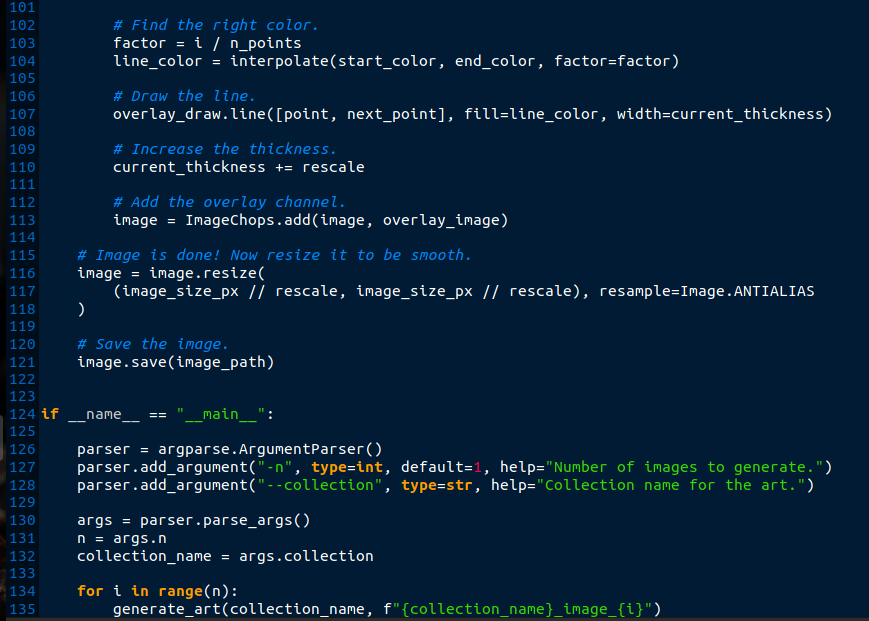


**CODE**

**￼**



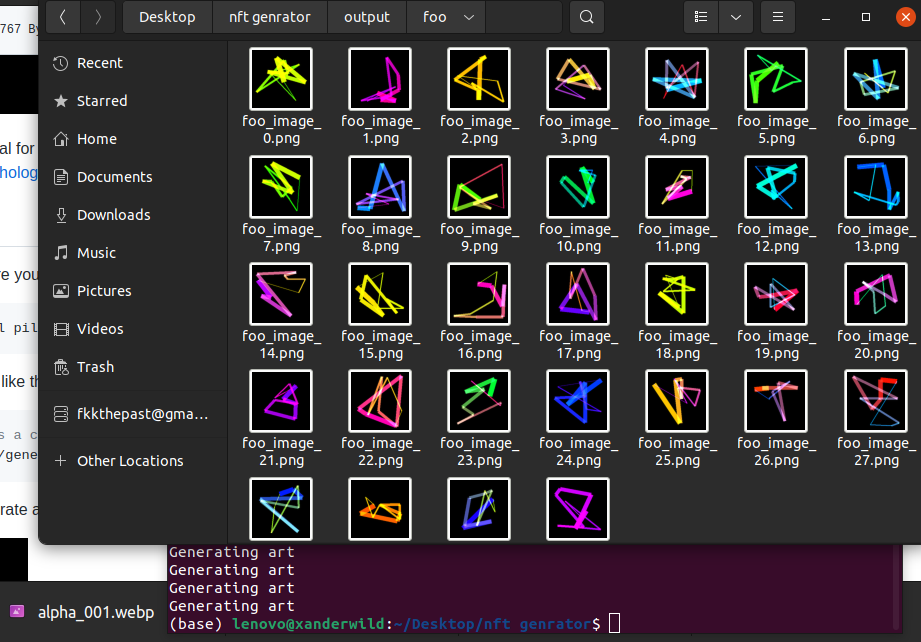




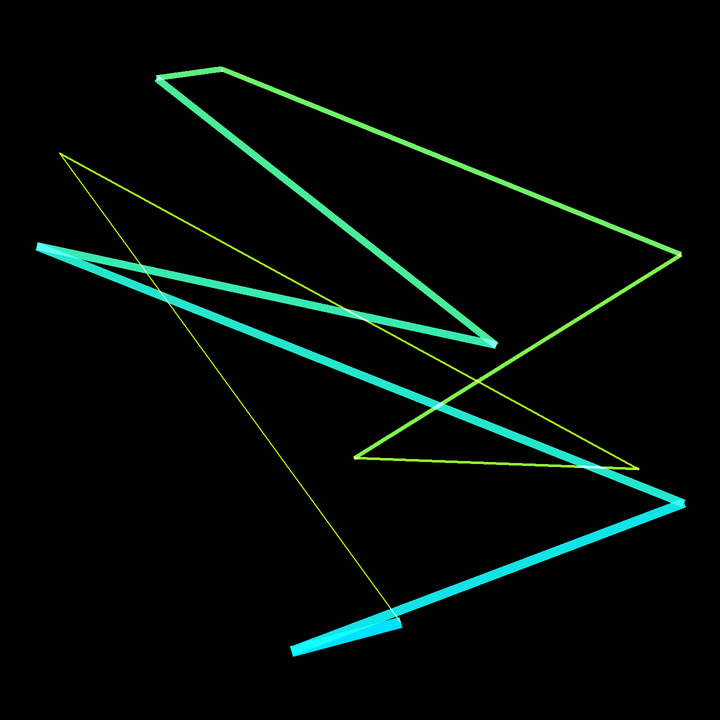


**OUTPUT**

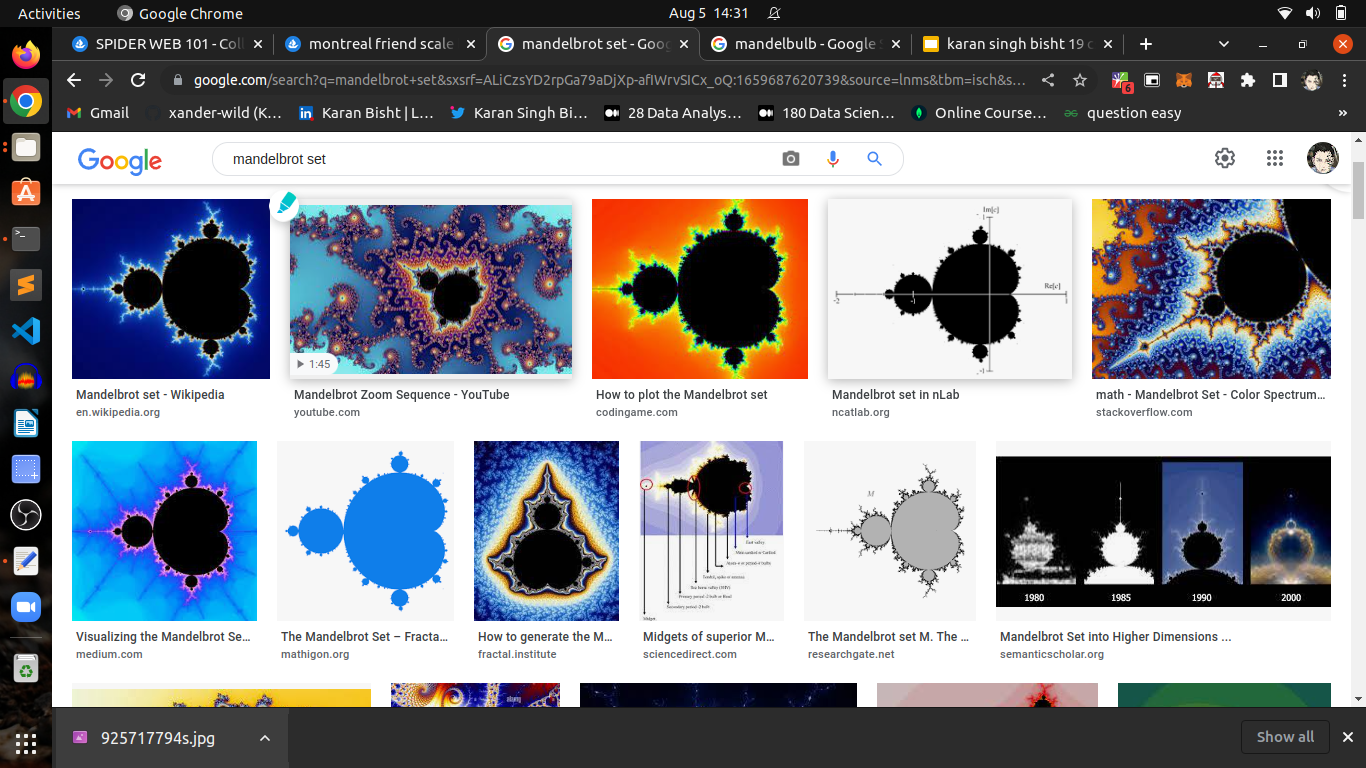
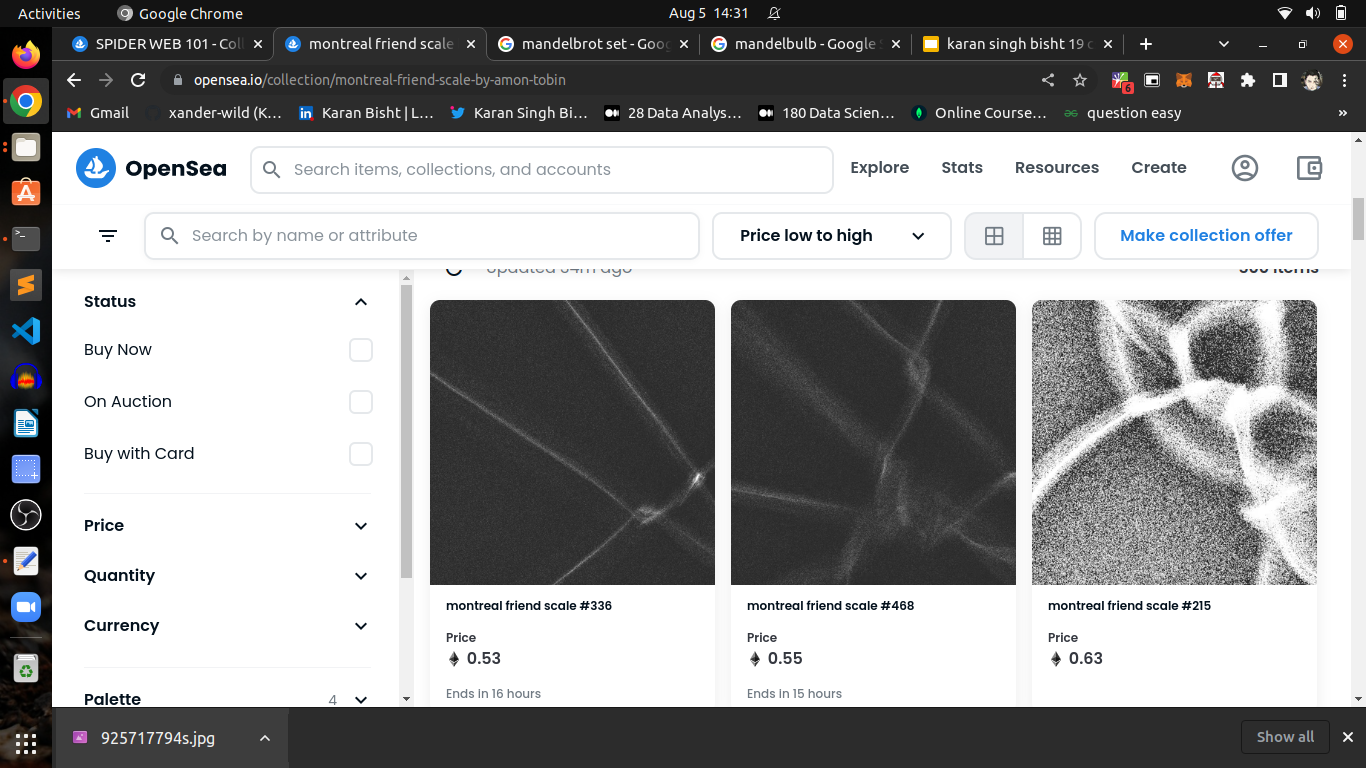
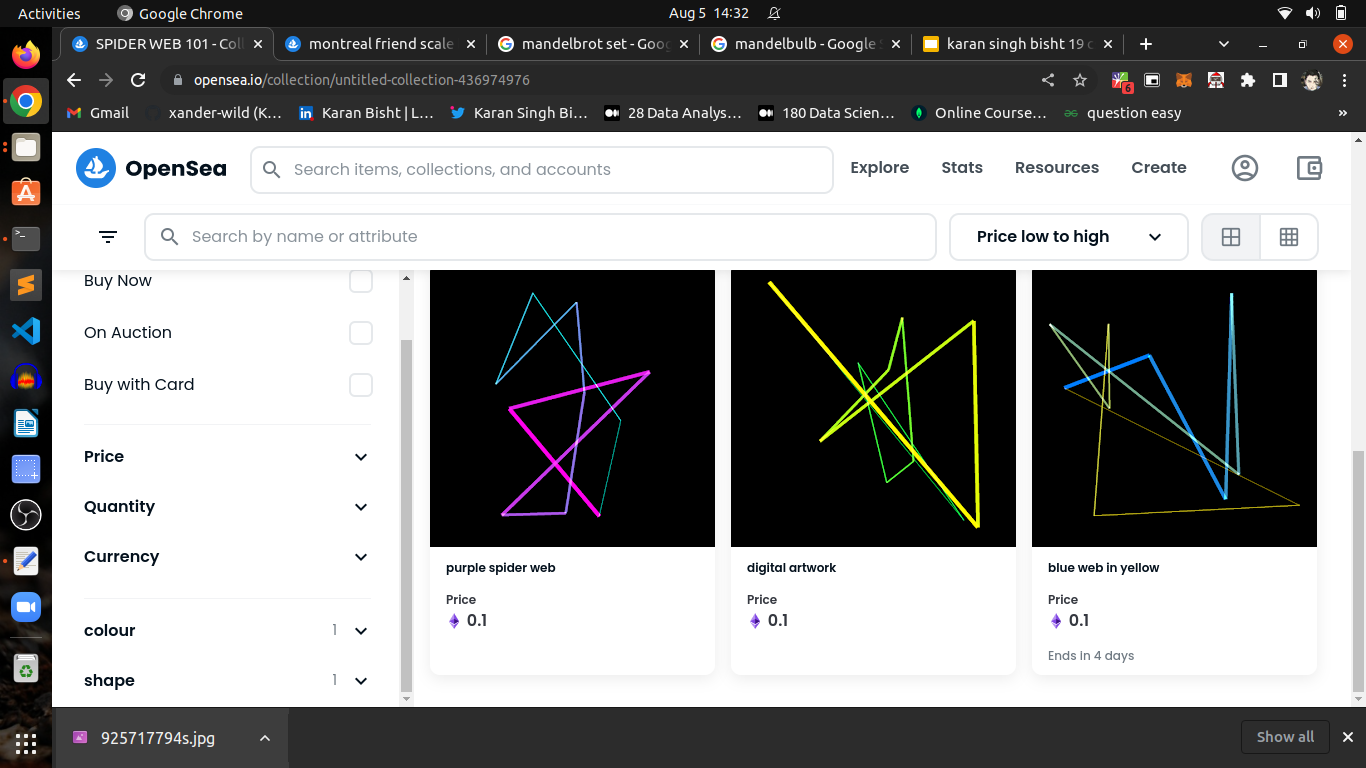
OUTPUT FOLDER SCREEN SHOT



ONE OF THE OUTPUT SCREENSHOT



PRODUCT ONN OPEN SEA - FOR SELLING AND SIMILAR PRODUCT



**CHAPTER 5**

**INFORMATION ABOUT TESTING STRATEGY**



**Chapter 5**

**INFORMATION ABOUT TESTING STRATEGY**

The testing of software means measuring or accessing the software to determine the quality. Testing Is a measuring instrument for software quality with the unit of measurement being the number of defects found during testing.

Testing activities also help to achieve software quality. Testing is essential in the development of any system software. Testing is essential in the development of any software system.Testing is in order to assess what the system actually does and how well expected to spend approximately 40% of development cost and time in testing in order to achieve reasonable quality levels.

**Levels of Testing**

❖ Unit Testing

❖ Component Integration Testing

❖ System Testing

❖ Acceptance Testing

**Unit Testing**

Unit testing comprises the set of tests performed usually by the programmers prior to the integration of the unit into a large Program. This is the lowest level of testing and is done by the programmer (Who develops it) who can test it in great detail. The function is done in isolation. This Is where the most detailed investigation of internal working of the individual unit is carried out.

**Component Integration Testing**

When two or more tested components are combined into a larger structure, the testing process should look for errors in two ways:-

❖ In the interface between the components



**System Testing**

❖ The functions, which can be performed by the new group

After integration testing is completed, the entire system is tested as a whole. The functional specifications or requirements specification are used to derive the test case. At this level the system testing looks for errors in the end-to-end functional quality. Attributes such as performance, reliability,

Volume, stress tolerance, usability, maintainability, security etc. Independent testers can carry out this testing.

**Acceptance Testing**

After system testing was complete, the system was handed over to the training section. Acceptance testing marks the transaction from ownership by the developer to ownership by the users.The acceptance test is different in nature to the development testing in three ways.

Firstly, it is a responsibility of the accepting section rather than the development department (computer Centre).

Secondly, the purpose of the acceptance testing was to find out whether the software is working rather than trying to find errors.

Thirdly, it also includes the testing of the user's department’s working practices to ensure that the computer software will fit into clerical & administrative procedures of the concerned section well.

Acceptance testing gave confidence to the user that the system is ready for operational use.

**TESTING OBJECTIVES**

No bugs blocks the execution of

tasksSystem states are visible

All factors affecting the output are

visibleFunctional simplicity

Distinct output is generated for each input

**TEST REPORTS**

**Debugging**

Debugging occurs as a consequence of successful testing. Debugging refers to the process of identifying the cause for defective behavior of a system and addressive that

problem. In less complex terms- fixing a bug. When a test case uncovers an Error, debugging is the process that results in the removal of the error. The debugging process begins with the execution of a test case. The debugging process attempts to match symptoms with cause, thereby leading to error correction. The following are two alternative outcomes of the debugging:

the cause will be found and necessary action such as correction or removal will be taken.The cause will not be found.

**Characteristics of bugs**

1. The symptoms ant the cause may be geographically remote. That is, the symptoms may appear in one part of a program. While the cause may actually be located at a site that is far removed. Highly coupled program structures exacerbate this situation.

2. The symptoms may disappear (temporarily) when another error is connected. 3. the Symptom may actually be caused by non error(e.g, round-Off inaccuracies). 4. The symptoms may be caused by human error that is not easily traced.

5. The symptom may be a result of timing problems, rather than processing problems.

6. It may be difficult to accurately reproduce input conditions(e.g, a real-time application in which input ordering is indeterminate).

7. The symptoms may be intermittent. This is particularly common in embedded systems that couple hardware and software inextricably.

8. The symptoms may be due to causes that are distributed across a number of tasks running on different processors.

SYSTEM SECURITY MEASURES

To do an adequate job on security, a systems analyst must analyze the risk, exposure, and costs and specify measures such as passwords and encryption to provide protection. The backup copies of software and recovery restart procedures must be available when needed.



**CHAPTER 06**

**CONCLUSION**



**Chapter 6**

**CONCLUSION**

**CONCLUSION:**

**FUTURE SCOPE**

This NFT genrator can create and use a lot of diffrent algortihms and make a good moeny with it and create spactacular artwork.

**CONCLUSION**

The project entitled “NFT GENRATOR ” is developed using HTML CSS and Javascript as front end to display the basic information of the student along with the attendance and marks. This project covers only the basic features required.



**CHAPTER 7**

**REFERENCES**



**Chapter 7**

**REFERENCES**

**REFERENCES**

1. **https://www.geeksforgeeks.org/python-pillow-a-fork-of-pil/**
2. **https://decrypt.co/resources/erc-721-ethereum-nft-token-standard**
3. **https://www.coursera.org/articles/what-is-python-used-for-a-beginners-guide-to-using-python**
4. **https://cointelegraph.com/news/nft-market-well-positioned-to-grow-35-into-a-13-6b-industry-by-2027**
5. **https://en.wikipedia.org/wiki/Algorithmic\_art**
6. **https://opensea.io/collection/untitled-collection-436974976**
7. **https://opensea.io/collection/montreal-friend-scale-by-amon-tobin**
8. **https://www.google.com/search?q=mandelbrot+set&sxsrf=ALiCzsYD2rpGa79aDjXp-afIWrvSICx\_oQ:1659687620739&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjp0smYoq\_5AhV8zTgGHdcQBicQ\_AUoAXoECAIQAw&biw=1314&bih=636&dpr=1**



**CHAPTER 8**

**BRIEF PROFILE**



**063**

**Chapter 8**

**BRIEF PROFILE**

I, KARAN SINGH BISHT, students of B. Tech Computer Science and Engineering, have created the above project report during my 6th semester on the title **NFT GENERATOR** for the partial fulfillment of the award for B. Tech Degree. Highlights of the project, my project includes-

• Use of Python which is a General Purpose Programming

Language, programmed using the SUBLIME.

• Easy to access and bunches of updated NFT’s which are trending topic and can be use in game , in videos and a lot of other digital creations .

• You can permentaly own the licence of these nfts by buying them and for every sale i can genrate my own commison .

I am hereby submitting my project report.

Yours Sincerely,

Karan Singh Bisht