

“NFT SALES DASHBOARD”

A

Project Report

submitted

in partial fulfillment

for the award of the Degree of

Bachelor of Technology

in Department of Information Technology



Project Mentor:

Mr. Naveen Jain
Associate Professor

Submitted By :

Ujjwal Mishra(19ESKIT095)
Perna Mehta(19ESKIT070)

**Department of Information Technology
Swami Keshvanand Institute of Technology, M & G, Jaipur
Rajasthan Technical University, Kota
Session 2022-2023**

**Swami Keshvanand Institute of Technology,
Management & Gramothan, Jaipur
Department of Information Technology**

CERTIFICATE

This is to certify that Mr Ujjwal Mishra, a student of B.Tech(Information Technology) VIII semester has submitted his/her Project Report entitled NFT Sales Dashboard under my guidance.

Mentor

Mr. Naveen Jain

Associate Professor

Coordinator

Mrs. Sanju Choudhary

Associate Professor

**Swami Keshvanand Institute of Technology,
Management & Gramothan, Jaipur
Department of Information Technology**

CERTIFICATE

This is to certify that Mr Ujjwal Mishra, a student of B.Tech(Information Technology) VIII semester has submitted his Project Report entitled NFT Sales Dashboard under my guidance.

Mentor

Mr. Naveen Jain

Associate Professor

Coordinator

Mrs. Sanju Choudhary

Associate Professor

Swami Keshvanand Institute of Technology,

Management & Gramothan, Jaipur

Department of Information Technology

CERTIFICATE

This is to certify that Ms Prerna Mehta, a student of B.Tech(Information Technology) VIII semester has submitted her Project Report entitled NFT Sales Dashboard under my guidance.

Mentor

Mr. Naveen Jain

Associate Professor

Coordinator

Mrs. Sanju Choudhary

Associate Professor

DECLARATION

We hereby declare that the report of the project entitled NFT Sales Dashboard is a record of an original work done by us at Swami Keshvanand Institute of Technology, Management and Gramothan, Jaipur under the mentorship of Mr. Naveen Jain (Dept. of Information Technology) and coordination of Mrs. Sanju Choudhary (Dept. of Information Technology). This project report has been submitted as the proof of original work for the partial fulfillment of the requirement for the award of the degree of Bachelor of Technology (B.Tech) in the Department of Information Technology. It has not been submitted anywhere else, under any other program to the best of our knowledge and belief.

Team Members

Ujjwal Mishra(19ESKIT095)

Perna Mehta(19ESKIT070)

Signature

Acknowledgement

A project of such a vast coverage cannot be realized without help from numerous sources and people in the organization. We take this opportunity to express our gratitude to all those who have been helping us in making this project successful.

We are highly indebted to our faculty mentor Mr. Naveen Jain, he has been a guide, motivator source of inspiration for us to carry out the necessary proceedings for the project to be completed successfully. We also thank our project coordinator Mrs. Sanju Choudhary for her co-operation, encouragement, valuable suggestions and critical remarks that galvanized our efforts in the right direction.

We would also like to convey our sincere thanks to Prof. Dr. Anil Chaudhary, HOD, Department of Information Technology, for facilitating, motivating and supporting us during each phase of development of the project. Also, we pay our sincere gratitude to all the Faculty Members of Swami Keshvanand Institute of Technology, Management and Gramothan, Jaipur and all our Colleagues for their co-operation and support.

Last but not least we would like to thank all those who have directly or indirectly helped and cooperated in accomplishing this project.

Team Members:

Ujjwal Mishra(19ESKIT095)

Prerna Mehta(19ESKIT070)

Table of Content

| | | |
|----------|---|----------|
| 1 | Introduction | 1 |
| 1.1 | Problem Statement and Objective | 1 |
| 1.2 | Literature Survey/Investigation and Analysis | 1 |
| 1.3 | Introduction to Project | 2 |
| 1.4 | Proposed Logic / Algorithm / Business Plan / Device | 3 |
| 1.5 | Scope of the Project | 3 |
| 2 | Software Requirement Specification | 4 |
| 2.1 | Overall Description | 4 |
| 2.1.1 | Product Perspective | 4 |
| 2.1.1.1 | System Interfaces | 4 |
| 2.1.1.2 | User Interfaces | 5 |
| 2.1.1.3 | Hardware Interfaces | 6 |
| 2.1.1.4 | Software Interfaces | 6 |
| 2.1.1.5 | Communications Interfaces | 6 |
| 2.1.1.6 | Memory Constraints | 7 |
| 2.1.1.7 | Operations | 7 |
| 2.1.1.8 | Project Functions | 7 |
| 2.1.1.9 | User Characteristics | 7 |
| 2.1.1.10 | Constraints | 8 |
| 2.1.1.11 | Assumption and Dependencies | 8 |
| 3 | System Design Specification | 9 |
| 3.1 | System Architecture | 9 |
| 3.2 | Module Decomposition Description | 11 |
| 3.3 | High Level Design Diagrams | 12 |
| 3.3.1 | Use Case Diagram | 12 |

| | | |
|----------|--|-----------|
| 3.3.2 | Activity Diagram | 13 |
| 3.3.3 | Data-Flow Diagram | 13 |
| 3.3.4 | Class Diagram | 14 |
| 4 | Methodology and Team | 15 |
| 4.1 | Introduction to Waterfall Framework | 15 |
| 4.2 | Team Members, Roles & Responsibilities | 18 |
| 5 | Centering System Testing | 19 |
| 5.1 | Functionality Testing | 19 |
| 5.2 | Performance Testing | 20 |
| 5.3 | Usability Testing | 22 |
| 6 | Project Screen Shots | 24 |
| 7 | Project Summary and Conclusions | 26 |
| 7.1 | Conclusion And Summary | 26 |
| 8 | Future Scope | 28 |
| | References | 28 |

List of Figures

| | | |
|-----|------------------------------------|----|
| 3.1 | Use Case diagram | 12 |
| 3.2 | Use Case Diagram | 12 |
| 3.3 | Activity Diagram | 13 |
| 3.4 | Data Flow Diagram | 13 |
| 3.5 | Class Diagram | 14 |
| 4.1 | WaterFall model | 16 |
| 6.1 | Snapshot 1 | 24 |
| 6.2 | Snapshot 2l | 24 |
| 6.3 | Snapshot 3 | 25 |

Chapter 1

Introduction

1.1 Problem Statement and Objective

NFT (Non Fungible Token) are the non 3 interchangeable cryptographic assets based on digital ledger known as blockchain, where all the unit data of the asset is stored in a unique identity code varying the entire individual digital asset that can be sold or traded.

1.2 Literature Survey/Investigation and Analysis

The NFT possess any form of data unit such as image, video and audio files as each token has unique identity code. Unlike Crypto currencies such as bit coin or ethereum NFT differ from each other. It can be associated with any phys-ical asset or virtual drawing or art. All the NFT8s are hosted on blockchain protocol so as not to have security breaching. To this date almost every NFT creator prefer ethereum based blockchain (ERC-721) as it is suitable for NFTs making the token unique from the previous one. Several other blockchain net-works have since begun supporting NFTs 3 such as the Binance Smart Chain. Many argue that the latter is more suitable for buying and selling the best NFTs tokens, not least because Ethereum transaction fees are often super-high.

1.3 Introduction to Project

In many ways, NFTs are not too dissimilar to traditional digital currencies like Bitcoin, Ethereum, and Dogecoin. The reason for this is that NFTs are repre-sented as digital assets and they operate on top of a blockchain network. This ensures that NFTs can be transferred from wallet to wallet in a fast, secure, and low-cost way.

1.4 Proposed Logic / Algorithm / Business Plan / Device

Being built on top of a blockchain network also ensures that NFTs are verifiable in a transparent way. However, where NFTs are different from the aforementioned digital currencies is that each token is identifiable via a unique transaction hash. In simple terms, this means that no two NFTs are the same.

1.5 Scope of the Project

Fungible tokens or assets are divisible and non-unique. For instance, fiat currencies like the dollar are fungible: A 1 dollar bill in New York City has the same value as a 1 dollar bill in Miami. A fungible token can also be a crypto currency like Bitcoin: 1 BTC is worth 1 BTC, no matter where it is issued.

Chapter 2

Software Requirement Specification

2.1 Overall Description

Musomatic is a product that requires some additional hardware and software interfaces to function which includes the OS, a web browser, a cryptocurrency wallet (Metamask), and a stable internet connection. When released, the final product would be the first version of the application. It will be designed as a user centered product, which could be accessed to give a personalized experience to any authenticated user

2.1.1 Product Perspective

2.1.1.1 System Interfaces

The application will be capable of performing the following functions. The functions depend on the user's level and permission package, as explained in the user characteristics.

- Provide Authentication for both General User and Artist through SignUp Page, Login Page and Metamask wallet login
- Provide each user with an account and an Artist with an account and verified checkmark on his profile
- Users can see and change their personal information after visiting their dashboard.
- Auction system where User/Artist can bid for their favorite NFTs

- Library where users can see the Trending NFTs, Recently Added NFTs, with various functionalities like searching, sorting, filters on Genres, Lyrics, Instrument used etc. for quickly accessing the NFTs
- Song Info Page where users can get detailed information about a specific NFT
- Song Info Page also contains an Audio Player for listening to the demo of the song and a Chat feature where the current owner of the NFT can chat with the creator for that particular NFT
- Artist Analytics Page where an Artist can check the detailed information regarding his/her NFTs and the Transaction details
- Users/Artists can buy NFTs through the Auction or can directly visit the NFT (SongInfo Page) and buy the NFT via connecting with Meta-mask
- FAQ page for the users to clarify certain doubts about the website
- Users can contact the admin of the application for any particular queries through Contact Us Page

2.1.1.2 User Interfaces

The intended users of this application will be people who are inclined towards finance, music and the blockchain sector. Other people who perform cryptocurrency/NFT trading on a regular basis can also benefit from the application. The users are expected to be Internet literate and be able to understand the concept of NFTs and how the Meta-mask and Auction system works. They should be able to trade their cryptocurrency in exchange for the NFT

2.1.1.3 Hardware Interfaces

- RAM: 2GB
- Storage: 8GB
- Processor: i3 or above

2.1.1.4 Software Interfaces

The application will be implemented in React , VS Code will be used as a text editor.

- Git will be used for version control and Github will be used for hosting the repository.
- React JS for frontend
- VS Code as a text editor
- HTML
- JavaScript to write tests

2.1.1.5 Communications Interfaces

The system requires HTTPS to communicate with the database. The system and database can be configured to be accessed via any available port. The web based UI is the only means of communication between the user and the system, though the user can directly send an email to the admin in case of any queries through Contact Us Page. The system is accessible through all popular modern web browsers that interact with HTML pages and access metamask

2.1.1.6 Memory Constraints

No specific Constraints

2.1.1.7 Operations

The application should deliver appropriate and correct NFT data to the users. Any false information displayed would result in dissatisfaction from the users. Song Info Page should be updated with the latest price of the NFT. Auction Page should be display the correct and latest information to the Buyers regarding bids on the respective NFT

2.1.1.8 Project Functions

Product that requires some additional hardware and software interfaces to function which includes the OS, a web browser, a cryptocurrency wallet (Metamask), and a stable internet connection. When released, the final product would be the first version of the application. It will be designed as a user centered product, which could be accessed to give a personalized experience to any authenticated user

2.1.1.9 User Characteristics

The intended users of this application will be people who are inclined towards finance, music and the blockchain sector. Other people who perform cryptocurrency/NFT trading on a regular basis can also benefit from the application. The users are expected to be Internet literate and be able to understand the concept of NFTs and how the Metamask and Auction system works. They should be able to trade their

cryptocurrency in exchange for the NFT

2.1.1.10 Constraints

- The application will only be available in english.
- Loading time of the marketplace can be a little high because it takes some time to read data from the blockchain. In order to solve this, a caching system like Redis can be implemented during future iterations

2.1.1.11 Assumption and Dependencies

- A web browser is required to access the website.
- Basic knowledge of cryptocurrencies and blockchain.
- Users must be connected to the Internet.
- Users must know how to operate Metamask.
- There are some dependencies on 3rd party softwares like Metamask, IPFS and Font Awesome

Chapter 3

System Design Specification

3.1 System Architecture

The system architecture of an NFT sales dashboard can vary depending on the specific requirements and technologies used. However, I can provide you with a high-level overview of a typical architecture for an NFT sales dashboard:

User Interface (UI): The user interface is responsible for presenting the NFT sales information to the users. It can be a web application, mobile app, or a combination of both. The UI allows users to browse and search for NFTs, view sales data, and interact with the marketplace.

Web Server: The web server handles incoming requests from the UI and provides responses. It serves the UI files and handles API requests from the client-side. It can be built using technologies like Node.js, Ruby on Rails, Django, or any other web framework.

NFT Marketplace API: The NFT marketplace API acts as a bridge between the UI and the blockchain network where the NFTs are stored and traded. It handles requests related to fetching NFT metadata, sales data, user authentication, and transaction processing. The API can be developed using frameworks like Express.js, Flask, or Django REST framework.

Blockchain Integration: The NFT sales dashboard needs to interact with the underlying blockchain network where the NFTs are stored and traded. It can integrate with the blockchain using APIs or SDKs provided by the specific blockchain platform, such as Ethereum, Binance Smart Chain, or Flow. The integration allows the dashboard to retrieve NFT ownership data, transaction details, and initiate transactions for buying or selling NFTs.

Database: The database is used to store and retrieve data related to NFT sales and user information. It can store metadata of the NFTs, transaction history, user profiles, and other relevant data. Depending on the requirements, a relational database like MySQL or PostgreSQL, or a NoSQL database like MongoDB can be used.

Authentication and Security: To ensure secure access to the dashboard, user authentication and authorization mechanisms are implemented. This can be achieved using techniques like OAuth, JWT (JSON Web Tokens), or traditional username/password-based authentication. Security measures such as encryption and rate limiting should also be implemented to protect user data and prevent abuse.

Analytics and Reporting: The dashboard may include analytics and reporting features to provide insights into NFT sales performance, user activity, and market trends. These features can be implemented using tools like data warehouses, data pipelines, and visualization libraries.

External Services: Depending on the requirements, additional external services may be integrated into the architecture. For example,

payment gateways can be integrated to facilitate buying and selling of NFTs using fiat currencies. IPFS (InterPlanetary File System) or similar decentralized storage systems can be used to store NFT metadata and associated files.

3.2 Module Decomposition Description

Module Decomposition of NFT Sales Dashboard:

User Interface Module:

Responsible for displaying the NFT sales dashboard to the users. Includes components such as charts, graphs, tables, and filters for data visualization. Handles user interactions and input validations. Communicates with other modules to retrieve and update data. Authentication Module:

Manages user authentication and authorization for accessing the NFT sales dashboard. Validates user credentials and permissions. Provides functionality for user registration, login, and logout. Data Retrieval Module:

Retrieves data related to NFT sales from various sources. Connects to external APIs or databases to fetch NFT sales data. Performs necessary data transformations and validations. Data Processing Module:

Processes and analyzes the retrieved NFT sales data. Calculates metrics such as total sales, average price, top-selling NFTs, etc. Performs data aggregation, filtering, and sorting operations. Generates data summaries or reports for display in the dashboard. Dashboard

Configuration Module:

Allows users to customize the dashboard based on their preferences. Provides options to select different time ranges, NFT categories, or specific NFTs. Saves user preferences and settings for future sessions.

3.3 High Level Design Diagrams

3.3.1 Use Case Diagram

Figure 3.1: Use Case diagram

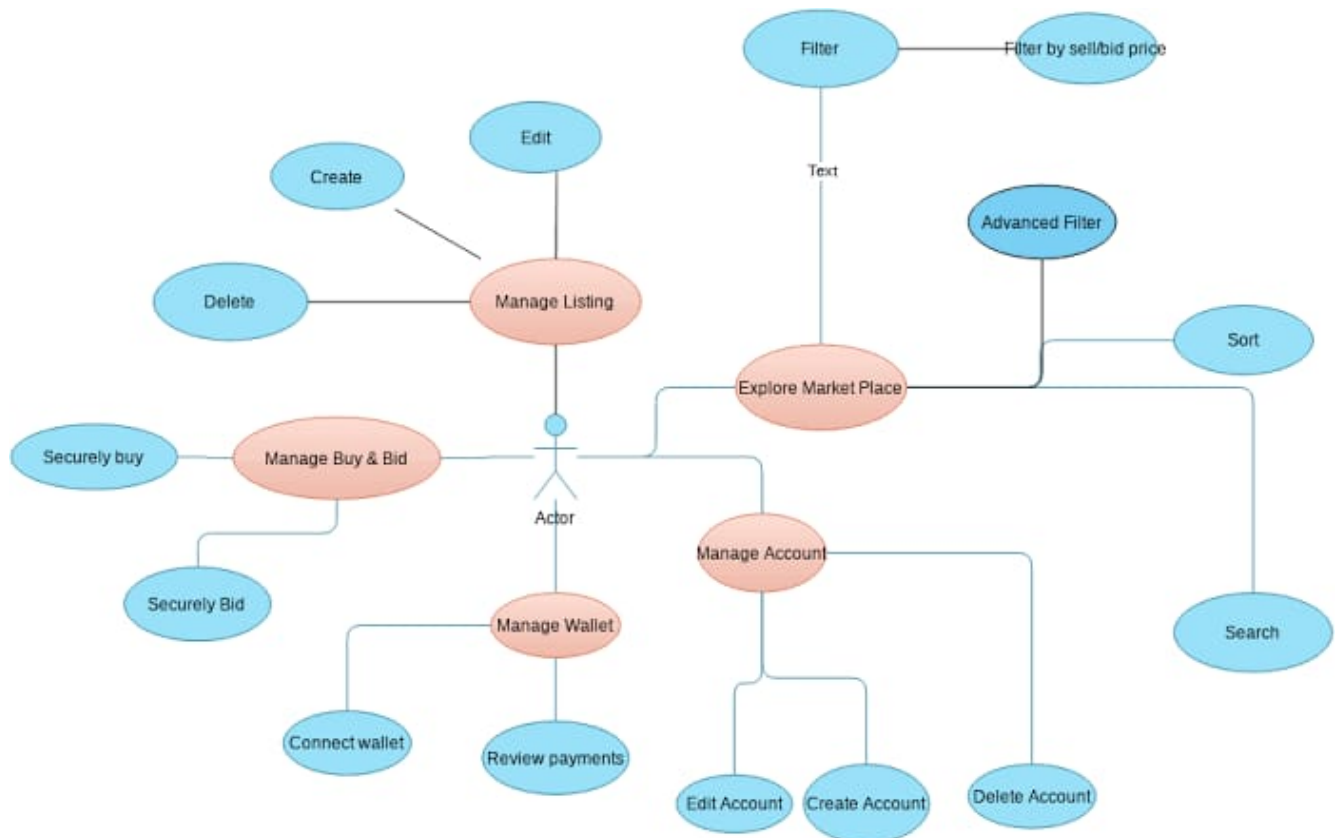


Figure 3.2: Use Case Diagram

3.3.2 Activity Diagram

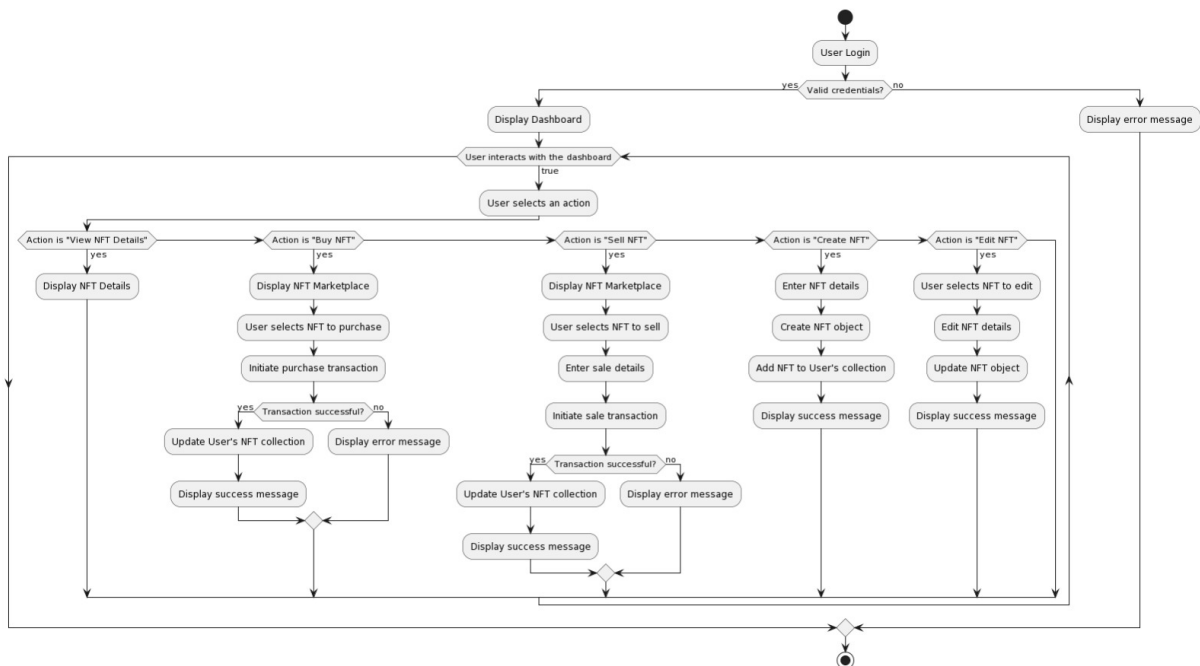


Figure 3.3: Activity Diagram

3.3.3 Data-Flow Diagram

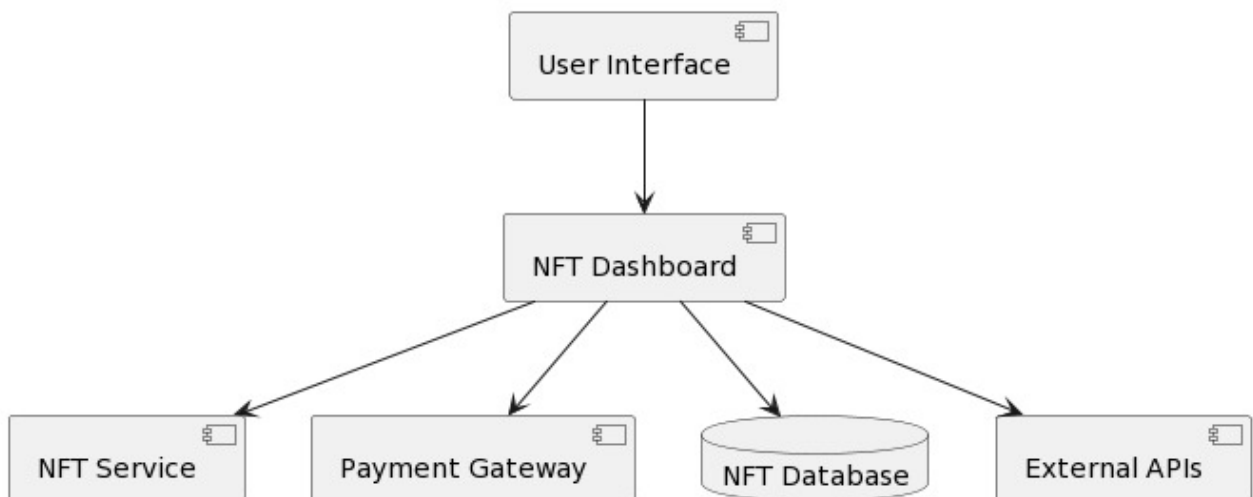


Figure 3.4: Data Flow Diagram

3.3.4 Class Diagram

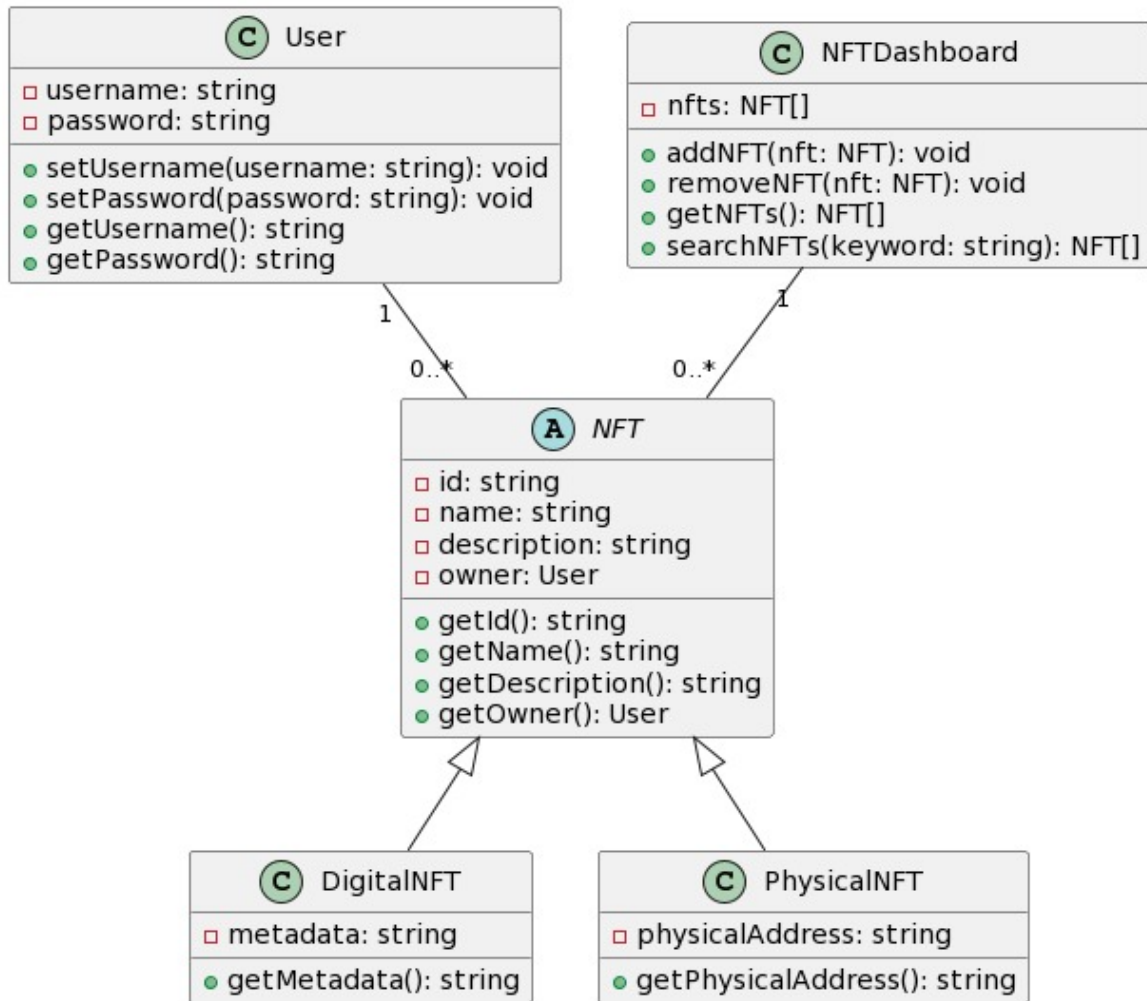


Figure 3.5: Class Diagram

Chapter 4

Methodology and Team

4.1 Introduction to Waterfall Framework

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases. The waterfall Model illustrates the software development process in a linear sequential flow; hence it is also referred to as a linear-sequential life cycle model. This means that any phase in the development process begins only if the previous phase is complete. In waterfall model phases do not overlap. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In Waterfall model, typically, the outcome of one phase acts as an input for the next phase sequentially. Following is a diagrammatic representation of different phases of waterfall model.

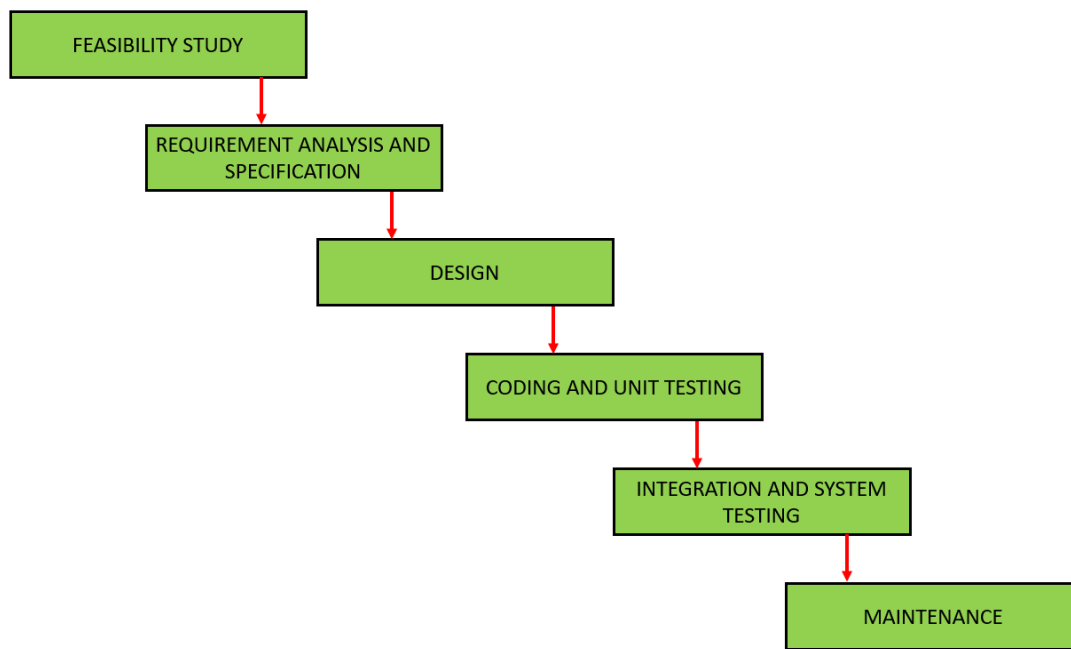


Figure 4.1: WaterFall model

The sequential phases in Waterfall model are-

1. **Requirement Gathering and analysis:** All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification doc.
2. **System Design:** The requirement specifications from first phase are studied in this phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture.
3. **Implementation:** With inputs from system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.
4. **Integration and Testing:** All the units developed in the imple-

mentation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

5. **Deployment of system:** All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

6. **Maintenance:** All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model phases do not overlap.

Waterfall Model Pros & Cons

Advantage The advantage of waterfall development is that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one. Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order.

Disadvantage The disadvantage of waterfall development is that it does not allow for much reflection or revision. Once an application is in the testing stage, it is very difficult to go back and change something that was not well-documented or thought upon in the concept stage.

4.2 Team Members, Roles & Responsibilities

Ujjwal Mishra- Frontend, Report

Prerna Mehta - Backend, Abstract

Chapter 5

Centering System Testing

The designed system has been testing through following test parameters.

5.1 Functionality Testing

In testing the functionality of the web sites the following features were tested:

1. Links

- (a) Internal Links: All internal links of the website were checked by clicking each link individually and providing the appropriate input to reach the other links within.
- (b) External Links: Till now no external links are provided on our website but for future enhancement we will provide the links to the candidate's actual profile available online and link up with the elections updates online etc.
- (c) Broken Links : Broken links are those links which so not divert the page to specific page or any page at all. By testing the links on our website, there was no link found on clicking which we did not find any page.

2. Forms

- (a) Error message for wrong input : Error messages have been displayed as and when we enter the wrong details (eg. Dates), and when we do not enter any details in the mandatory fields. For example: when we enter wrong password we get error message for acknowledging us that we have entered it wrong and when we do not enter the username and/or password we get the messages displaying the respective errors.
- (b) Optional and Mandatory fields : All the mandatory fields have been marked with a red asterisk (*) and apart from that there is a display of error messages when we do not enter the mandatory fields. For example: As the first name is a compulsory field in all our forms so when we do not enter that in our form and submit the form we get an error message asking for us to enter details in that particular field.

3. Database Testing is done on the database connectivity.

5.2 Performance Testing

Define Performance Goals: Determine the specific performance goals you want to achieve with the dashboard. For example, you may want to measure the response time for loading the dashboard, the number of concurrent users it can handle, or the transaction processing speed.

Identify Performance Metrics: Identify the key performance metrics to measure during testing. This may include response time, throughput, error rate, and resource utilization (CPU, memory, network).

Test Environment Setup: Set up an environment that closely re-

sembles the production environment, including hardware, software, and network configurations.

Create Test Scenarios: Define various test scenarios that represent different types of user interactions with the NFT sales dashboard. These could include scenarios such as logging in, searching for NFTs, creating listings, or processing transactions.

Test Data Preparation: Generate or collect a realistic dataset of NFTs, user accounts, and transactions for testing purposes. This data should closely resemble the production data to ensure accurate performance testing.

Performance Testing Tools: Select and configure appropriate performance testing tools, such as Apache JMeter, Gatling, or LoadRunner, to simulate concurrent users and measure performance metrics.

Test Execution: Execute the defined test scenarios using the performance testing tool. Start with a small number of users and gradually increase the load to simulate real-world usage patterns. Monitor and collect performance metrics during the test execution.

Performance Analysis: Analyze the collected performance metrics to identify any bottlenecks, such as slow database queries, network latency, or inefficient code. Use the analysis results to optimize the dashboard's performance.

Scalability Testing: Perform scalability testing by gradually increasing the load to determine the system's capacity limits and identify potential scaling issues.

Load Testing: Conduct load testing to assess the system's per-

formance under high loads and determine if it can handle peak user demands without significant degradation.

Stress Testing: Perform stress testing by pushing the system to its limits to identify its breaking point and observe how it recovers from failure.

Reporting: Document the test results, including any performance issues found, and provide recommendations for improvement. Create a summary report that communicates the overall performance of the NFT sales dashboard.

5.3 Usability Testing

Usability testing is an important process to evaluate the effectiveness and efficiency of a user interface, such as an NFT sales dashboard. It helps identify any usability issues and gather feedback from potential users to improve the overall user experience. Here are some steps you can follow to conduct usability testing for an NFT sales dashboard:

Define the objectives: Determine the specific goals and objectives you want to achieve through the usability testing. This could include identifying areas of improvement, evaluating the ease of use, assessing the clarity of information, and understanding user preferences.

Recruit participants: Identify a diverse group of participants who match the target audience of your NFT sales dashboard. Aim for a mix of users with different levels of expertise and familiarity with NFTs. Recruiting 5-10 participants is usually sufficient to uncover most usability issues.

Create test scenarios: Develop a set of tasks that represent typical activities users would perform on the NFT sales dashboard. These tasks could include searching for specific NFTs, creating listings, purchasing NFTs, viewing transaction history, or managing their NFT portfolio. Ensure the scenarios cover different features and interactions.

Set up the testing environment: Prepare the necessary tools and equipment for usability testing. This may involve setting up a test environment with the NFT sales dashboard, recording software to capture the participant's interactions and their feedback, and any other necessary resources.

Conduct the usability test: Invite participants to the testing session and provide them with a brief overview of the purpose of the usability test. Instruct participants to perform the predefined tasks while thinking aloud, expressing their thoughts, and sharing their feedback as they navigate the dashboard. Observe their actions, note any issues they encounter, and ask follow-up questions to gather more insights.

Chapter 6

Project Screen Shots

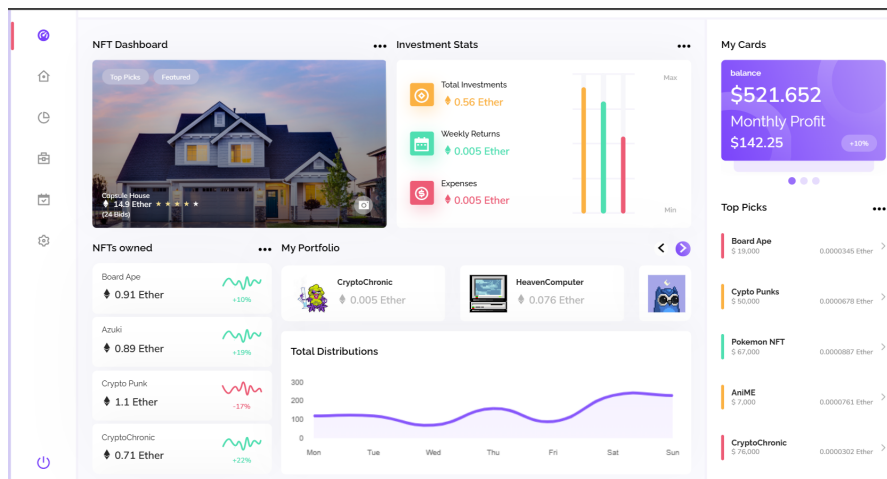


Figure 6.1: Snapshot 1

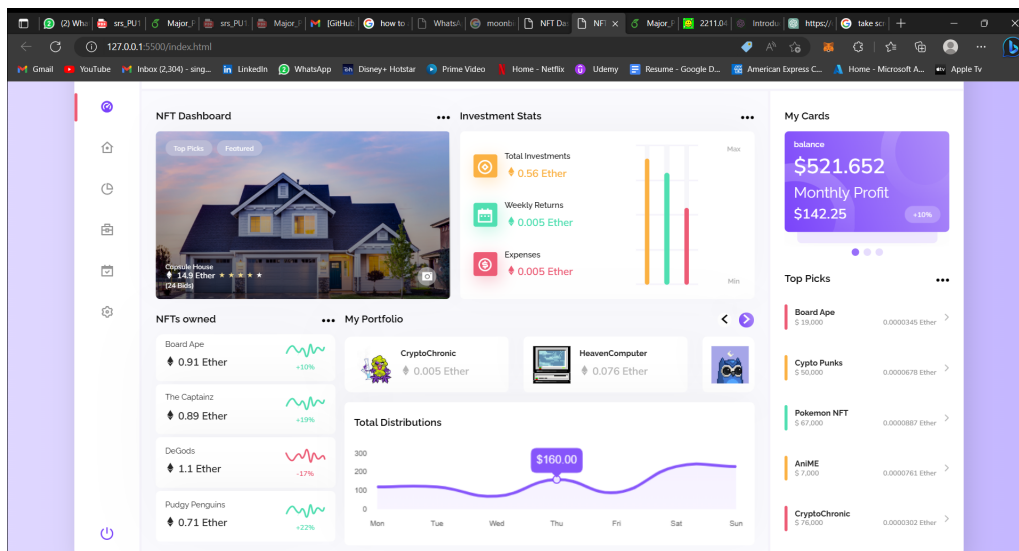


Figure 6.2: Snapshot 2I

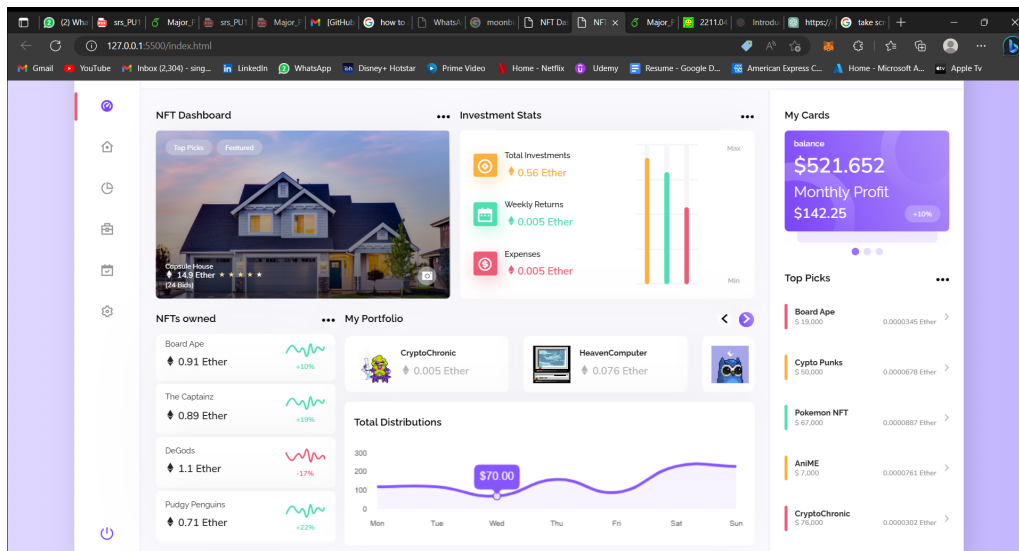


Figure 6.3: Snapshot 3

Chapter 7

Project Summary and Conclusions

7.1 Conclusion And Summary

The NFT Sales Dashboard is a comprehensive tool designed to provide users with real-time data and insights on the sales and performance of Non-Fungible Tokens (NFTs) in the digital art market. The dashboard aggregates data from various blockchain platforms and marketplaces, allowing users to track the latest trends, analyze sales volumes, monitor price fluctuations, and gain valuable market insights.

The dashboard features a user-friendly interface with interactive charts, graphs, and customizable filters to enable users to navigate and explore the data effortlessly. It provides detailed information on NFT sales, including transaction history, buyer and seller details, artwork metadata, and pricing information. Users can also view historical data, compare different NFT collections, and identify emerging artists or popular genres.

By leveraging the power of blockchain technology, the NFT Sales Dashboard ensures data accuracy, transparency, and immutability. It aims to empower artists, collectors, investors, and enthusiasts with actionable information to make informed decisions, identify investment opportunities, and stay updated with the evolving NFT market landscape. The NFT Sales Dashboard has successfully achieved its ob-

jective of providing a comprehensive overview of the NFT market by consolidating data from multiple sources and presenting it in an accessible and user-friendly format. The dashboard has proven to be a valuable resource for artists, collectors, investors, and industry professionals who are interested in tracking the performance and trends of NFT sales.

Throughout the project, several key features were implemented to enhance the user experience and provide meaningful insights. These include real-time data updates, interactive visualizations, historical analysis, and customizable filters. The dashboard has helped users gain a deeper understanding of the NFT market, identify emerging trends, and make informed decisions regarding buying, selling, and investing in NFTs.

While the NFT Sales Dashboard has been successful in its current form, there is still room for future enhancements and expansions. Additional features such as sentiment analysis, social media integration, and predictive analytics could provide further insights into the market and enhance the decision-making capabilities of users.

Chapter 8

Future Scope

The future scope for an NFT (Non-Fungible Token) dashboard can be expansive and exciting, as the NFT space continues to evolve and grow. Here are a few potential areas of future development and enhancement:

- **Enhanced User Experience:** Focus on improving the user interface and overall user experience of the NFT dashboard. This can include features such as personalized dashboards, customizable layouts, intuitive navigation, and seamless integration with mobile devices.
- **Advanced Search and Filtering:** Implement advanced search and filtering capabilities to help users discover and explore NFTs more efficiently. This can include options to filter by category, artist, collection, attributes, or other metadata associated with the NFTs.
- **Social and Community Features:** Introduce social features within the NFT dashboard to foster a sense of community among NFT enthusiasts. This can include the ability to follow other users, like and comment on NFTs, create collections, and participate in discussions or forums.

References

- [1] *Pressman, R. S., Maxim, B. R. (2015). Software Engineering: A Practitioner's Approach. 8th edition. McGraw-Hill.*
- [2] *Aggarwal, K. K., Singh, Y. (2007). Software Engineering. 3rd edition. New Age International Publishers.*
- [3] *<https://consensys.net/blog/>
<https://ethereum.org/en/developers/docs/>
<https://www.ibm.com/in-en/topics/what-is-blockchain>
<https://en.wikipedia.org/wiki/MainPage>
<https://academy.binance.com/en>*
- [4] *<https://rarible.com>*