

Lab 2 Sections 1 & 2 - Art Guardian Product Specifications

Team Blue

Old Dominion University

CS 411

Professor Brunelle

27 November 2022

Final Version

Table of Contents

1.	Introduction.....	4
1.1.	Purpose.....	4
1.2.	Scope.....	4
1.3.	Definitions, Acronyms, and Abbreviations.....	5
1.4.	References.....	6
1.5.	Overview.....	6
2.	General Description.....	6
2.1.	Prototype Architecture Description.....	6
2.2.	Prototype Functional Description.....	8
2.3.	External Interfaces.....	9
2.3.1.	Hardware Interfaces.....	9
2.3.2.	Software Interfaces.....	9
2.3.3.	User Interfaces.....	10
2.3.4.	Communication Protocols and Interfaces.....	10
2.3.5.	Product Requirements.....	10

List of Figures

Figure 1: Art Guardian Major Functional Component Diagram.....	7
--	---

List of Tables

Table 1: Prototype Features.....	8
----------------------------------	---

1. Introduction

Non-fungible tokens (NFTs) became popularized in 2017 and have grown exponentially over the past five years. (2022, August 31) During the end of 2021, the NFT market was estimated to be 41 billion dollars. (2022, January 6) This is a massive increase from 2020 when it was worth a measly 100 million dollars. While the hype of NFTs has steadily increased over the past year, there doesn't exist standard regulation to 'mint' these NFTs of talent artists' work without permission. This lack of regulation allows for artwork to be stolen and profits to be made. The popular artwork platform, DeviantArt, has seen an exponential rise in unlawful NFT minting of their artists' work and artists are through with sitting idly and watching their hard work become someone else's payoff.

Art Guardian is *the* solution to the problem of stolen artwork and wrongly-minted NFTs. Art Guardian will detect stolen artists' work on NFT marketplaces and automate the Digital-Millennium Copyright Act (DMCA) generation/sending process for taking down these fraudulent NFTs.

1.1 Purpose

Art Guardian is a progressive web application that will automatically detect fraudulent NFTs that use an artists' stolen art pieces, generate a DMCA take-down request, and send the DMCA request to the respective NFT marketplace. Digital artists who want to protect their artwork from being minted and sold on NFT marketplaces will be the focal point for Art Guardian. They will use Art Guardian by uploading images of their artwork to the web application, from there Art Guardian will monitor popular, notable NFT marketplaces for their art. If a piece of artwork is detected and deemed stolen, Art Guardian will send a notification to the artist and begin generating a DMCA takedown request. Once the artist confirms their artwork, they'll be able to sign and send the DMCA request to the marketplace. *If* the NFT marketplace chooses to ignore the DMCA takedown request, then Art Guardian will not take any further legal action.

1.2 Scope

The goal of Art Guardian is to simplify and consolidate the process of removing stolen NFT artwork from NFT marketplaces for artists. This will be done by automatically detecting stolen NFTs, generating DMCA requests, and sending the DMCA to the respective marketplaces. Artists that use Art Guardian will feel more confident knowing that their artwork will be safely monitored and protected.

The Art Guardian prototype will give users the ability to create an account and have access to all basic features that Art Guardian offers. Users will have the ability to change their account information, like email and username, as well as upload their artwork. The prototype will have a notification system that will notify users if their art was detected on an NFT marketplace. Users will be able to view all of their artwork in their art library page. There they will be able to edit, remove, and add uploaded artwork. The other main features of the Art Guardian prototype are the stolen art detection system and the DMCA generation automator.

1.3 Definitions, Acronyms, and Abbreviations

NFT: A sellable, tradeable, non-fungible token that exists on the blockchain and represents some form of data.

Blockchain: a decentralized, immutable, public ledger that is split among multiple computers.

Art Platform: a site that artists use to publish their art.

DMCA (Digital Millennium Copyright Act) Takedown: act of taking down a copyrighted work from a website on behalf of the owner of that work.

Minting: using a piece of data, such as an image, to create a unique.

NFT NFT Marketplace: website where NFTs are sold.

AWS (Amazon Web Services): Largest provider of various cloud computing services.

AWS Amplify: an AWS service for building full-stack web applications.

Amazon RDS (Relational Database Service): cloud-based database service which can work with other AWS services.

Amazon DynamoDB: cloud-based database service which can work with other AWS services.

NFTport API: interface for working with popular NFT blockchains and markets. **React:** open-source, front-end JavaScript library for creating websites with modern user interface.

Stripe API: programming interface for verifying a user's identity.

OpenCV: open-source computer vision library for Python.

Gmail API: a programming interface for the creation and sending of emails.

MySQL: a relational database management system.

Git: version control system for tracking software changes.

GitHub: online hosting of the git version control system.

HTML: markup language for displaying documents in the web browser.

CSS: style sheet language that specifies the style and layout of how documents are displayed in a web browser.

1.4 References

NFTs ballooned to a \$41 billion market in 2021 and are catching up to the total size of the global fine art market. (2022, January 6). Markets Insider.

<https://markets.businessinsider.com/news/currencies/nft-market-41-billion-nearing-fine-art-market-size-2022-1>

(2022, August 31). *Reasons Behind the Growing Popularity of NFTs*. Antier Solutions.

<https://www.antiersolutions.com/reasons-behind-the-growing-popularity-of-nfts/>

1.5 Overview

The product specification details the architecture of the prototype, the functions of the prototype, and the external interfaces. The section on the product specification architecture details and explains the major functional components and their interactions with each other. The section going over the prototype functions section summarizes the functions of the prototype and how it compares and contrasts with the real-world product. Also, the details about the software, hardware, and user interfaces for the prototype will be in the external interfaces section.

2. General Description

Users will be able to access Art Guardian via a web browser. The Art Guardian prototype will demonstrate the core features of the fraudulent NFT artwork detection system and automation of generating and sending DMCA takedown requests. Basic user functionality will also be available, this includes notifications, art upload, account creation/deletion, and the art library.

2.1 Prototype Architecture Description

Most major functional components will be available in the Art Guardian prototype. The only components that will be either partially implemented or removed are the user ID verification API, Gmail API, and the user information database, this can be seen in Figure 1.

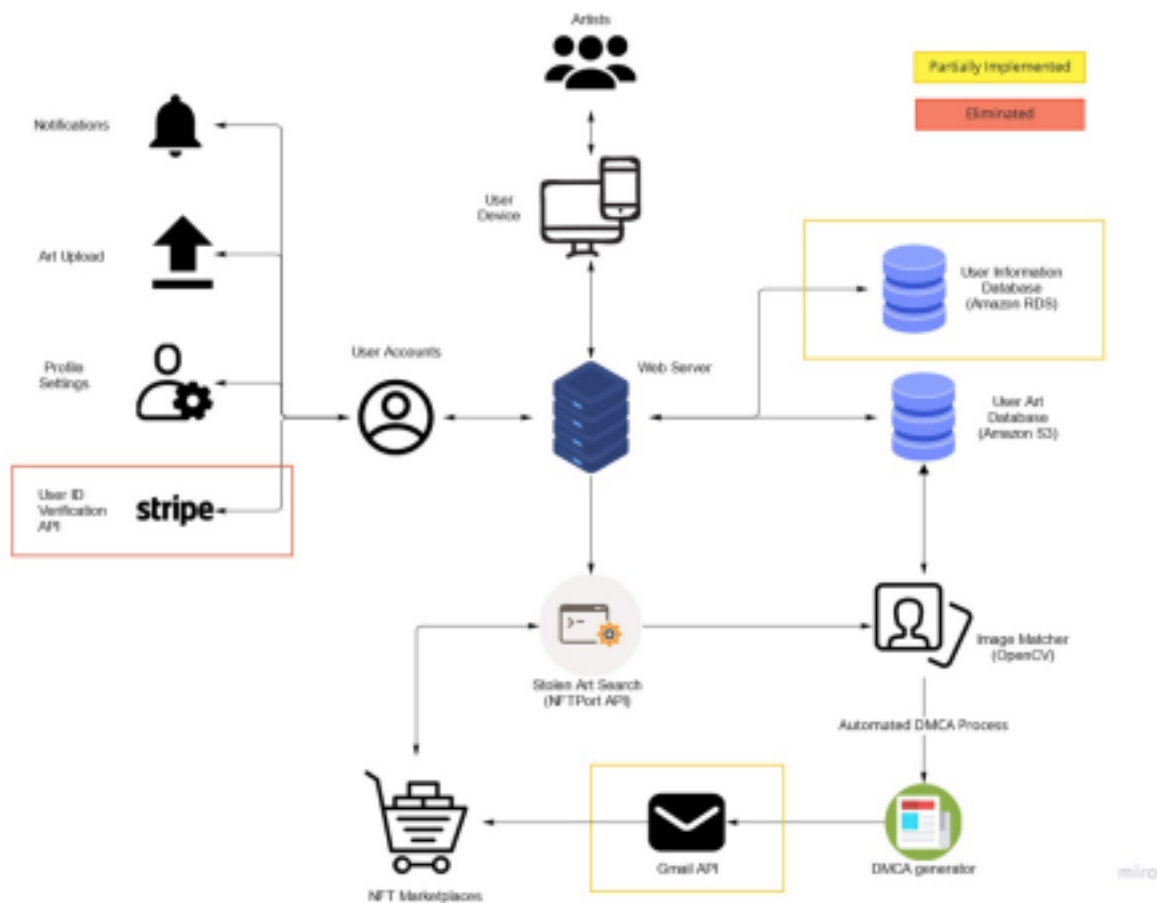


Figure 1
Art Guardian Major Functional Component Diagram

The components of the web server will host the web application and will act as a hub for all other components to interact through. The artist can connect to Art Guardian through a web browser on their respective device. The Art Guardian accounts have many smaller components. These will be the notification, profile settings components, and art upload. The notification system will be the detection system that alerts users of their stolen art on NFT marketplaces.

User information in the Art Guardian prototype will be stored in a database using Amazon RDS and a user art database will be using Amazon S3. Because of the situation of this being a prototype, the user information database will only be partially implemented since only mock data will be used. The image recognition component will use OpenCV to compare art in the database to that found in NFT marketplace through NFTport API. The API component will search through the NFT marketplace for the art. The DMCA generator component will only be activated if a substantial match is found. Once this happens, the DMCA will be sent to the

marketplace using the Gmail API. The Gmail API will only be partially implemented because emails will be sent to test emails instead of real NFT marketplaces.

2.2 Prototype Functional Description

The majority of the features of the real-world product (RWP) will be fully implemented in the Art Guardian prototype. As seen in Table 1, the DMCA monitoring system will be partially implemented and user verification and DMCA tracking from the RWP will be eliminated from the prototype. Art Guardian will have all necessary features to detect stolen artwork and automatically generate and send the DMCA takedown requests. Marketplace monitoring and image recognition will be fully implemented so that detection of stolen art is completed. Account creation will be fully implemented for the prototype. Once the user creates an account, their data will be stored in the user information database. The art upload system will store user's images in the user art database. The image library will use the uploaded art and display them for the user. Periodically, Art Guardia prototype will routinely scan the NFT marketplaces for stolen art. If a match is found the user will be notified. The DMCA generation will be fully implemented, but the filing won't be fully implemented since the DMCA's will be sent to test emails rather than real NFT marketplaces.

Table 1

Prototype Features

Features	Description	Prototype Implementation
Account Controls		
Account Creation	Create account for users to utilize Art Guardian features	Fully Implemented
GUI		
Art Upload	Users upload images of artwork to the database	Fully Implemented
Image Library	Users view the artwork they have uploaded	Fully Implemented
Whitelisting	Remove art previously uploaded to the database	Fully Implemented

Automated Processes		
Marketplace Monitoring	Checks NFT marketplaces periodically for image matching	Fully Implemented
Image Matching	Matches images from NFT marketplaces to images on the database	Fully Implemented
Stolen Art Alert	Send the user notifications of stolen art and DMCA request statuses	Fully Implemented

DMCA Takedown		
DMCA Generation	Generate DMCA takedown notices that are automatically prefilled based on a template.	Fully Implemented
DMCA Filing	Issue generated DMCA takedown notice to the offending NFT marketplace through the Gmail API.	Partially Implemented
DMCA Cataloging	Catalog and monitor all generated DMCA takedown notices.	Fully Implemented

2.3 External Interfaces

Many different protocol interfaces for hardware, software, user, and communication will be used in the Art Guardian prototype

2.3.1 Hardware Interfaces

Users can connect to Art Guardians using a web browser on a computer. Operating systems that can be used are Windows, MacOS, and Linux.

2.3.2 Software Interfaces

Art Guardian will utilize the React Framework so that a modernized simplistic user interface will be available. The Gmail API will be utilized for sending the DMCA takedown request. To match images, OpenCV library will image match using matching detection algorithms. NFTport API will be used to search images on NFT marketplaces. Both Github and Git version control will be imperative in developing the Art Guardian prototype since it allows many team members to collaborate on the same code without syncing errors. AWS Amplify and AWS RDS will be implemented to host the web application and provide the database.

2.3.3 User Interfaces

The user will have the ability to use the web application through use of the mouse and keyboard. A screen resolution of 1280x720 or higher is required.

2.3.4 Communications Protocols and Interfaces

Because Art Guardian is a web application used on a browser with the internet, HTTPS will be utilized to secure the connection of data over the internet.

3. Product Requirements