Annette Bazan

ITAI 2372

Class Notes

AI and Blockchain Integration

Blockchain is a decentralized, distributed ledger technology that records transactions across a network of computers in a secure transparent, and tamper resistant way. Each “block” contains a list of transactions linked to the previous block via cryptographic hashes, farming a “chain”. It is the backbone of cryptocurrencies can be used in supply chain management, smart contracts, and identity verification, and NFT.

Blockchain and AI integration complement each other, AI thrives on data and computation while blockchain provides a secure, immutable way to store and share that data.

AI can use “Blockchain” to ensure integrity of training datasets. AI can enhance “blockchain” systems by optimizing consensus algorithms or detect fraud on real-time on “blockchain” networks.

* It has its own applications, blockchain doesn’t need a middleman, it has trust through math, it has immutability, resilience if one point has failed there’s still more that have the real data. It can settle transactions in minutes or hours instead of days and is cheaper than traditional systems.
* Smart contracts: a self-executing program stored on blockchain that runs when predefined conditions are met. The trust is in the code itself. It is used in crypto, NFTs, insurance, voting, and supply chains.
* Downsides: Bugs are forever if the code has a flaw once its live, legal gray area, cost, and oracles aren’t perfect it can not see the real-world without external data feed which can be wrong or hacked.

Smart contract optimization: AI can draft, optimize smart contract to make them more effective and adaptable.

Fraud detection and security- AI can monitor the entire blockchain and can spot fraud/anomalies.

Data marketplace and monetization – open sea is an example.

Supply chain – amazon warehouse or Walmart

Identity verification- AI is the best to detect AI fraud

Crypto trading using AI to generate

There are other tedious things like hospital records speed is not important but Trust is!

Healthcare records and research

Last use case decentralized AI Governments- all the records that is being stored in smart ledgers

Opensea.io- homework for spring break how to list a smart contract and opensea [The beginner's guide to creating & selling digital art NFTs | OpenSea](https://opensea.io/blog/articles/the-beginners-guide-to-creating-selling-digital-art-nfts)

### List ABCS for nft and how ai can help Step 1: Set Up Your Tools and Wallet

Before you can sell art or deploy a smart contract on OpenSea, you need a cryptocurrency wallet and some Ethereum (ETH) or another supported blockchain’s native currency for gas fees.

1. **Get a Crypto Wallet**:
   * Download and install a wallet like **MetaMask** (a browser extension or mobile app). It’s widely used and compatible with OpenSea.
   * Create a new wallet, save your seed phrase securely (never share it), and set a strong password.
2. **Fund Your Wallet**:
   * Buy Ethereum (ETH) or Polygon (MATIC) via an exchange like Coinbase, Binance, or Kraken, then transfer it to your MetaMask wallet. ETH is used for the Ethereum blockchain, while MATIC is for Polygon (cheaper gas fees).
   * You’ll need this for gas fees—small transaction costs on the blockchain. Polygon is often recommended for beginners due to lower fees.
3. **Connect to OpenSea**:
   * Go to [opensea.io](https://opensea.io).
   * Click the wallet icon in the top right, select MetaMask, and connect your wallet. If it’s your first time, OpenSea may ask you to sign a message to verify ownership (no gas fee for this).

**Step 2: Prepare Your Art**

Your art needs to be in a digital format to become an NFT. OpenSea supports various file types, but keep it simple for your first try.

1. **Create or Digitize Your Art**:
   * Use tools like Photoshop, Procreate, or Canva to create digital art, or scan physical artwork.
   * Supported formats: Images (JPG, PNG, GIF), videos (MP4), audio (MP3), or 3D models (GLB), up to 100MB.
2. **Store Your Art**:
   * You don’t need to upload it anywhere yet—OpenSea handles this during minting. However, ensure you have the file ready on your computer.

**Step 3: Create an NFT Without a Custom Smart Contract (Beginner Option)**

If you’re new to this and don’t want to code a smart contract right away, OpenSea’s built-in tools let you create NFTs easily. This uses OpenSea’s shared contract, so you skip coding for now.

1. **Access OpenSea Studio**:
   * On OpenSea, click your profile icon (top right) and select **Studio** from the sidebar.
2. **Create a Collection**:
   * Click **Create** > **Collection**.
   * Add a name (e.g., “My Fae Art”), logo, description, and category (e.g., Art).
   * Choose a blockchain: Ethereum (higher fees) or Polygon (lower fees, good for beginners).
   * Click **Create**—this sets up a collection without deploying a custom contract yet.
3. **Mint Your Art as an NFT**:
   * In Studio, select your collection, then click **Create** > **Item**.
   * Upload your art file.
   * Add a name, description, and optional traits (e.g., “Rarity: Rare,” “Type: Fae”).
   * Set the supply (1 for a unique piece).
   * Click **Create**. You’ll sign a transaction (Polygon has no gas fee for this; Ethereum might).
4. **List for Sale**:
   * Once minted, go to your collection, find your NFT, and click **Sell**.
   * Choose a fixed price (e.g., 0.05 ETH) or auction.
   * Sign the listing transaction. For your first sale on Ethereum, you’ll pay a one-time gas fee to initialize your account (could be $20–$100 depending on network traffic). Polygon is usually free or very cheap.

**Step 4: Develop a Custom Smart Contract (Advanced Option)**

If you want full control over your NFTs (e.g., for a unique fae-themed collection with special features), you’ll need to deploy your own smart contract. This requires some coding knowledge or a developer tool.

**Option A: Use OpenSea Studio (No Coding)**

1. **Deploy a Contract via Studio**:
   * In OpenSea Studio, click **Create** > **Drop** (or **Collection** if Drop isn’t available).
   * Fill out:
     + Logo: Upload an image.
     + Contract Name: e.g., “FaeArtCollection” (can’t change later).
     + Token Symbol: e.g., “FAE” (short, unique).
   * Pick a blockchain (Ethereum, Polygon, etc.) and check gas fee estimates.
   * Click **Continue**, approve the transaction in MetaMask (gas fee applies—e.g., $10–$50 on Polygon, more on Ethereum).
   * Once deployed, you can mint NFTs into this contract as in Step 3.

**Option B: Code Your Own (Technical)**

1. **Set Up Development Environment**:
   * Install **Node.js** and **npm** (package manager).
   * Use a framework like **Hardhat** or **Truffle**:
     + Run npm install -g hardhat in your terminal.
     + Create a project: npx hardhat init.
2. **Write a Simple NFT Contract**:
   * Use Solidity (Ethereum’s programming language). Here’s a basic ERC-721 contract:

solidity

CollapseWrapCopy

pragma solidity ^0.8.0;

import "@openzeppelin/contracts/token/ERC721/ERC721.sol";

contract FaeNFT is ERC721 {

uint256 public tokenCounter;

constructor() ERC721("FaeNFT", "FAE") {

tokenCounter = 0;

}

function mintNFT(address recipient) public returns (uint256) {

tokenCounter += 1;

\_mint(recipient, tokenCounter);

return tokenCounter;

}

}

* + Save this as FaeNFT.sol in your Hardhat project’s contracts folder.

1. **Deploy the Contract**:
   * Configure Hardhat with your MetaMask private key and an Ethereum/Polygon node (e.g., via Infura or Alchemy).
   * Write a deployment script (e.g., in scripts/deploy.js):

javascript

CollapseWrapCopy

async function main() {

const FaeNFT = await ethers.getContractFactory("FaeNFT");

const faeNFT = await FaeNFT.deploy();

await faeNFT.deployed();

console.log("Contract deployed to:", faeNFT.address);

}

main();

* + Run npx hardhat run scripts/deploy.js --network polygon (after funding your wallet with MATIC).

1. **Verify and Import to OpenSea**:
   * Verify your contract on a block explorer like Polygonscan (optional but recommended).
   * In OpenSea Studio, import your contract by entering its address under **My Collections** > **Add Existing Contract**.
   * Mint NFTs via your contract’s interface or OpenSea’s tools, then list them.

**Step 5: Sell Your Art**

1. **Promote Your Listing**:
   * Share your OpenSea NFT link on social media, Discord, or art communities.
   * Add unlockable content (e.g., a high-res file) to entice buyers.
2. **Monitor Sales**:
   * Check your OpenSea profile for offers or sales. When sold, funds go to your wallet minus OpenSea’s 2.5% fee.
3.  **Start Simple**: Use OpenSea’s no-code option first to test the waters.
4.  **Gas Fees**: Polygon is cheaper than Ethereum—ideal for beginners. Check [gas trackers](https://etherscan.io/gastracker) for timing.
5.  **Learning Resources**: OpenSea’s Help Center (support.opensea.io) or YouTube tutorials (search “OpenSea NFT 2025”) can fill gaps.
6.  **Safety**: Never share your seed phrase or private key.
7. **Digital Art**: Your artwork needs to be in a digital format (e.g., PNG, JPEG, GIF, MP4). Max file size on OpenSea is usually 100MB.
8. **Crypto Wallet**: You’ll need a wallet like MetaMask, Coinbase Wallet, or WalletConnect to interact with OpenSea and the blockchain.
9. **Ethereum (ETH)**: OpenSea runs primarily on Ethereum, and you’ll need ETH to pay "gas fees" (transaction costs) for minting and listing. Gas prices fluctuate, so have a little extra ETH handy (think $20-$100 depending on network traffic).
10. **OpenSea Account**: It’s free to sign up, but you’ll connect it to your wallet.

### Step-by-Step Process

1. **Set Up Your Wallet**
   * Download MetaMask (or your preferred wallet) as a browser extension or app.
   * Create a wallet, save your seed phrase somewhere safe (offline!), and fund it with ETH. You can buy ETH on exchanges like Coinbase, Binance, or Kraken, then transfer it to your wallet.
2. **Sign Up on OpenSea**
   * Go to [opensea.io](https://opensea.io).
   * Click the profile icon (top right), then "Sign In" or "Get Started."
   * Connect your wallet when prompted. Sign the transaction (no cost here) to link it.
3. **Create Your NFT**
   * Once signed in, click "Create" (top right).
   * You’ll be asked to set up a collection (a group for your art). Give it a name, description, and logo—think of it like your artist portfolio.
   * Upload your artwork file, add a title, description, and any extras (e.g., properties like "edition" or "color," or unlockable content like a secret link).
   * Choose the blockchain—Ethereum is standard, but OpenSea also supports Polygon (cheaper fees, less visibility).
   * Hit "Create" to mint it. If you’re on Ethereum, this might cost a one-time gas fee (around $50-$150 depending on the network). Polygon is often free or super cheap.
4. **List It for Sale**
   * Go to your profile, find your newly minted NFT under "Collected" or "Created."
   * Click it, then hit "Sell" (top right).
   * Pick your sale type:
     + Fixed Price: Set a price in ETH (e.g., 0.1 ETH ≈ $250 at today’s rough rates).
     + Timed Auction: Set a starting price and duration (e.g., 7 days). Buyers bid, highest wins.
   * Set a duration (default is 1 month, or customize it).
   * Optional: Reserve it for a specific buyer by adding their wallet address.
   * Sign the listing transaction with your wallet. First-time sellers on Ethereum pay a one-time "initialization" gas fee (could be $20-$100). After that, listing is usually free until it sells.
5. **Promote and Sell**
   * Once listed, share the link on social media, Discord, or wherever your audience hangs out.
   * When it sells, OpenSea takes a 2.5% cut, and you get the rest (minus gas fees if applicable). You can also set a royalty (e.g., 5%) for future resales.

### Tips for Success

* **Research Pricing**: Check similar art on OpenSea to gauge value. New artists might start low (0.01-0.05 ETH) to attract buyers.
* **Timing**: Ethereum gas fees spike during high traffic—use [etherscan.io/gas-tracker](https://etherscan.io/gas-tracker) to list when fees are lower (e.g., weekends or late nights).
* **Build Hype**: Tweet about it, join NFT communities, or drop free editions to grow a following.
* **Quality Matters**: Unique, high-effort art tends to stand out over quick doodles.

### Costs to Expect

* **Gas Fees**: One-time wallet initialization and minting fees on Ethereum. Could total $50-$200 upfront, depending on network congestion. Polygon avoids most of this.
* **OpenSea Fee**: 2.5% of the sale price, taken automatically.
* **No Upfront Listing Fee**: After initialization, listing is free until a sale happens.