

# William Greer

## Full Stack & Blockchain Engineer

Williamgreer327@gmail.com | in/william-greer-chase | Austin, TX, United States

---

### Professional Summary

- ◆ **Seasoned Blockchain Engineer** with over 7 years of experience designing, implementing, and optimizing **multi-chain blockchain solutions** across ecosystems such as Ethereum, BSC, Polygon, Solana, and Cosmos.
- ◆ Proficient in **full-stack development**, integrating blockchain systems with Web2 applications using **modern front-end frameworks (React, Angular, Vue.js)** and backend technologies like **Node.js, Flask, and Django**.
- ◆ Skilled in developing and deploying **smart contracts** using Solidity, Rust, and Vyper, ensuring robust, efficient, and secure decentralized applications (dApps).
- ◆ Extensive experience building **multi-chain dApps**, cross-chain bridges, and integrating **oracles** and APIs to connect blockchain ecosystems with real-world data.
- ◆ Strong expertise in creating **custom token standards** (ERC-20, ERC-721, ERC-1155) and implementing advanced DeFi solutions, including **decentralized exchanges (DEXs)**, **staking mechanisms**, **liquidity pools**, and governance frameworks.
- ◆ Expertise in **Cosmos SDK** and **Tendermint Core**, creating custom blockchain modules and enabling seamless interoperability using **IBC (Inter-Blockchain Communication)** protocols.
- ◆ Knowledgeable in **layer-1 and layer-2 scaling solutions**, such as Rollups, Plasma, and sidechains, for enhanced transaction throughput and cost efficiency.
- ◆ Advanced understanding of **cryptographic principles**, including public/private key encryption, elliptic curve cryptography, and zero-knowledge proofs, ensuring robust blockchain implementations.
- ◆ Experienced in deploying and maintaining blockchain infrastructure, including **node management**, **consensus configurations**, and monitoring for scalability and performance.
- ◆ Proficient in designing and implementing **cross-chain interoperability** solutions to enable seamless communication between distinct blockchain networks.
- ◆ Demonstrated ability to architect and deploy **end-to-end full-stack blockchain applications**, combining user-friendly front-end interfaces with highly performant blockchain-integrated backends.
- ◆ Skilled in blockchain security, including **smart contract auditing**, **vulnerability assessments**, and implementing secure key management practices.
- ◆ Passionate about innovation in blockchain technology, contributing to the community through **open-source projects**, **technical writing**, and mentoring upcoming developers.
- ◆ Committed to staying at the forefront of blockchain advancements, with expertise in **ZK-rollups**, **decentralized identity solutions**, **DAO frameworks**, and tokenomics design.
- ◆ **Strong team player and communicator**, with a proven ability to lead cross-functional teams, mentor developers, and collaborate effectively across technical and non-technical stakeholders.

## Professional Skills

<b>Blockchain Architecture &amp; Development</b>	Blockchain Protocols & Ecosystems	Ethereum, BSC, Solana, Polygon, Cosmos SDK, Polkadot, Hyperledger
	Smart contract Development & Optimization	Solidity, Rust, Substrate, Vyper, EVM-based contract design, gas-efficient contract architecture, proxy contract
	Layer-2 Scaling Solutions	Zkrollups, Optimistic Rollups, Plasma, sidechains, Raiden, stateful smart contracts
	Blockchain Security & Auditing	MythX, Slither, Oyente, Remix, ZKPs, ECDSA, EdDSA, RSA, self-sovereign identity(SSI)
	Tokenomics	DeFi tokenomics, ERC-20, ERC-721, NFT, metamask, phantom, DAO governance frameworks
	Cross-chain Bridges	Inter-blockchain Communication, atomic swaps, Cosmos SDK, Polkadot parachains
<b>Full Stack Development</b>	Frontend Development & Web3 Integration	React, Angular, Vue, Next.js, Redux, Zustand, Vuex, web3.js, Ethers.js, Drizzle, WalletConnect, web3-react, IPFS integration, Arweave, MUI, Ant Design, Tailwind CSS, GraphQL, on-chain/off-chain data
	Backend Development & Integration	Node.js, Express, NestJS, Python(Flask, Django), Go(Gin), Rest APIs, gRPC, WebSockets, microservices, MongoDB, PostgreSQL, MySQL, Cassandra, DynamoDB, Redis, RabbitMQ, Kafka
	Authentication & Authorization	JWT, OAuth, OpenID Connect, MetaMask, WalletConnect, SSO via blockchain
<b>Blockchain Development Tools &amp; Protocols</b>	Development Frameworks	Hardhat, Truffle, Brownie, Truffle Boxes, Scaffold-eth
	Decentralized Storage & File Sharing	IPFS, Filecoin, Arweave, Arweave bundles, NFT metadata storage
	Private Chains & Decentralized Authentication	Hyperledger Fabric, Quorum, Tendermint, Polkadot, Cosmos SDK, DID, SSI, Verifiable Credentials
<b>Security &amp; Project Management</b>	Cryptographic Principles & Tools	SHA-256, Keccak-256, Ed25519, elliptic curve cryptography, RSA, BLS signatures, zero-knowledge proofs (ZKPs) (zk-SNARKs, zk-STARKs), Homomorphic Encryption
	Security Auditing Tools	Mythril, Oyente, Manticore, Echidna, Certora
	Cloud Providers	AWS, Google Cloud Platform, Microsoft Azure, IBM Cloud
	Version Control & Project Management	Jenkins, GitLab CI/CD, CircleCI, GitHub Actions, automation of smart contract testing and deployment pipelines, Prometheus, Grafana, AWS CloudWatch, ELK Stack, Decentralized Logging Solutions

## Soft Skills & Language

- Team Collaboration
- Problem Solving & Innovation
- Communication & Documentation
- Agile & Project Management
- Continuous Learning & Community Involvement
- Leadership & Mentorship
- Language - English(Fluent)

## Professional Work Experience

### Senior Blockchain Engineer, *Beco Devs*, 04/2023 - Present

- **Designed and implemented a multi-layered DeFi ecosystem**, including decentralized exchanges (DEXs), yield farming protocols, staking mechanisms, and liquidity pools, leveraging **Ethereum, Binance Smart Chain, Polygon, and Cosmos** with **cross-chain interoperability** via custom bridges.
- Built and audited highly efficient **smart contracts** for core DeFi functionalities such as automated market maker (AMM) algorithms, liquidity pool logic, and token minting using **Solidity** and **Vyper**, incorporating features like dynamic fee adjustment and impermanent loss mitigation.
- Architected **layer-2 scaling solutions** using **Optimistic Rollups** and **zkRollups** to reduce transaction costs and improve throughput, achieving a **50% improvement in transaction speed** while maintaining security.
- Designed **custom tokenomics models** with dynamic reward mechanisms for liquidity providers and yield farmers, integrating vesting schedules and deflationary mechanisms for token value sustainability.
- Developed and optimized **cross-chain bridges** using **IBC (Inter-Blockchain Communication)** protocols, enabling seamless token and data transfers between Ethereum, Binance Smart Chain, and Cosmos ecosystems.
- Built a **secure CI/CD pipeline** using **Git, GitHub Actions, and Docker** for automated deployment of blockchain nodes, smart contracts, and supporting services, reducing deployment time by **40%**.
- Deployed and maintained blockchain infrastructure on **AWS**, including setting up **EC2 instances**, managing containerized services using **ECS** and **EKS**, and leveraging **CloudFormation** for scalable infrastructure as code.
- Optimized smart contract gas usage using **EVM assembly** techniques and gas profiling tools, achieving a **40% reduction in operational costs** for end-users.
- Engineered a modular **oracle integration layer** with providers like **Chainlink** and custom APIs to enable real-time asset pricing, liquidation mechanisms, and dynamic collateral adjustments.
- Implemented **staking pools** with advanced reward distribution mechanisms, integrating auto-compounding and multi-tiered incentives using Solidity.
- Built and monitored **blockchain nodes** on AWS, utilizing **CloudWatch, Prometheus, and Grafana** to ensure 99.9% uptime and quick issue resolution.
- Designed and integrated **zero-knowledge proof (ZKP)** mechanisms to enable confidential transactions and enhance platform privacy.
- Developed a high-performance front-end using **React.js**, integrated with back-end systems via **Web3.js, Ethers.js, and GraphQL**, offering a seamless user experience for DeFi operations.
- Deployed secure multi-signature wallets for governance treasury management using **Gnosis Safe** and implemented a **Decentralized Autonomous Organization (DAO)** framework for on-chain voting and community participation.
- Established and enforced best practices for version control, including branching strategies, code reviews, and release management using **Git**.
- Collaborated with DevOps teams to implement robust **AWS architecture**, utilizing **S3, Lambda, RDS, and CloudFront** to support decentralized application scalability and high availability.
- Contributed to the blockchain community by writing technical articles, conducting workshops, and mentoring junior developers on smart contract development, Git workflows, and AWS deployments.

### Full Stack Blockchain Engineer, *Bird Money*, 05/2022 - 06/2023

- **Developed and maintained a decentralized on-chain oracle system on Ethereum** to provide secure, real-time price feeds for various DeFi protocols, focusing on **reliable and tamper-proof data** such as asset prices, exchange rates, and interest rates.
- Integrated **Chainlink oracles** into Ethereum-based smart contracts, ensuring **accurate and decentralized price oracles** for DeFi applications, ensuring low-latency, high reliability, and minimal trust in a single data source.

- Developed **custom Ethereum-based oracles** to aggregate data from multiple trusted sources (e.g., centralized exchanges, decentralized exchanges) and feed that data into smart contracts, reducing single points of failure and price manipulation risks.
- Implemented **data fallback and redundancy mechanisms** in the oracle system, utilizing Ethereum smart contract logic for failover processes when primary data feeds are unavailable or compromised.
- Created and deployed a **DeFi lending protocol on Ethereum** that allows users to lend and borrow digital assets, using smart contracts to manage collateralized loans, interest rates, and liquidations.
- **Built core lending protocol smart contracts** using **Solidity**, enabling users to take out loans by locking their assets as collateral and automatically triggering liquidations if the collateral value drops below a certain threshold based on real-time oracle data.
- Designed and implemented **collateralized debt positions (CDPs)** on Ethereum, allowing borrowers to lock ERC-20 tokens as collateral and generate **Ethereum-based stablecoins** for borrowing, with **automatic liquidation triggers** to prevent insolvency.
- **Integrated real-time oracle data feeds** for dynamic interest rate models, utilizing the price feeds to adjust interest rates based on supply/demand in the Ethereum-based lending pools.
- Developed **liquidity pools** in the lending protocol for lenders to provide assets and earn interest, using smart contract logic to distribute rewards, ensuring liquidity and incentivizing participation in the Ethereum-based lending ecosystem.
- Engineered a **dynamic interest rate algorithm** that adjusts borrowing rates based on market liquidity, integrating oracle-driven data to determine optimal lending and borrowing conditions.
- Led the development of **governance mechanisms** for the lending protocol, enabling stakeholders to vote on key protocol parameters such as collateral types, loan-to-value (LTV) ratios, and interest rate models through Ethereum-based **DAO governance**.
- Implemented Ethereum-based **multi-signature wallets** and governance frameworks to ensure the security and decentralization of protocol decisions, with active participation from the community in protocol upgrades and asset management.
- **Optimized smart contract efficiency** by reducing gas costs and implementing **gas-efficient patterns** and modular smart contract design to ensure scalability and cost-effectiveness for end users interacting with Ethereum.
- Utilized **Hardhat** for Ethereum contract testing and **Ganache** for local testing environments, employing unit tests, integration tests, and deployment pipelines to ensure security and functionality of smart contracts on the Ethereum network.
- Collaborated with **security audit teams** to perform comprehensive audits on the Ethereum smart contracts, ensuring best practices for contract security, mitigating risks such as reentrancy, overflow, and other common vulnerabilities.
- **Monitored protocol performance and data feeds** in real-time using **Ethereum event logs** and external monitoring tools, ensuring data accuracy and tracking the health of the lending protocol.
- Deployed and managed Ethereum nodes using **AWS EC2** instances and **Docker** containers, ensuring high availability of Ethereum nodes for interacting with the lending protocol and oracle systems.
- **Built the frontend application using Next.js** to provide users with an intuitive and responsive interface for interacting with the Ethereum-based DeFi lending protocol, integrating with smart contracts via **Web3.js** and **Ethers.js**.
- Designed and developed key features in the frontend application, including **real-time loan status**, **interest rate calculations**, **collateral management**, and **transaction history**, ensuring seamless user experience.
- Implemented **dynamic data fetching** and state management in Next.js, utilizing **React hooks** and **Redux** for efficient data flow, and integrated Ethereum events with the frontend for live updates on contract states.
- Ensured high-performance frontend application by optimizing **SSR (server-side rendering)** in Next.js for fast page loads, SEO optimization, and dynamic content delivery for decentralized applications (dApps).
- Integrated **MetaMask** and **WalletConnect** for Ethereum wallet connections, enabling users to securely interact with the lending protocol and manage their assets directly through the frontend interface.
- **Tested and deployed** the frontend application using **Vercel**, ensuring fast, secure, and scalable hosting for the dApp, with continuous deployment pipelines connected to GitHub for smooth updates and new feature releases.
- **Used Git** for version control and **GitHub** for collaboration with cross-functional teams, maintaining a clean and efficient development pipeline while following best practices for code management.

- Led the **deployment of smart contracts to the Ethereum mainnet**, managing the migration process from testnets (Ropsten, Kovan) to production, ensuring a smooth transition and full protocol functionality on the Ethereum blockchain.

## Full Stack Engineer, *Erricks IT Solutions*, 04/2019 - 08/2023

- ❖ **Designed and developed** a scalable **Test Case Management System** using **Vue.js** (Vue 3.x, Vue Router, Vuex), implementing **component-based architecture** for modular UI components, allowing seamless integration with other testing tools and features.
  - Built dynamic forms and test case templates with **Vue.js** and **Vuetify**, providing an intuitive interface for adding, editing, and managing test cases, as well as test execution workflows.
  - Implemented a custom API with **Node.js** and **Express.js**, utilizing **RESTful principles** for routing and **MongoDB** for flexible document-based storage, allowing easy querying and updates of test cases and bug reports.
  - Developed secure role-based authentication and authorization logic using **JWT (JSON Web Tokens)** and **bcrypt** for password hashing, ensuring only authorized users can access and modify sensitive test case data.
  - Optimized MongoDB queries and indexes for fast retrieval of test case records, reducing latency when dealing with large sets of test cases and bug reports.
  - Integrated third-party testing tools like **Jenkins**, **Travis CI**, and **Slack** APIs for automated notifications, allowing test execution status updates to be sent in real-time to Slack channels.
  - Implemented a dynamic reporting system to automatically generate PDF reports of test results using **Puppeteer** and **Node.js**, enabling stakeholders to track testing progress without needing manual intervention.
  - Used **WebSockets** for real-time updates in the UI, allowing the QA team to instantly see changes to test case statuses, execution results, and bug tracking as they are updated by different users.
  - Leveraged **AWS S3** for scalable and secure storage of test case-related documents and reports, implementing file encryption for data protection.
  - Followed **Agile methodologies** and collaborated with cross-functional teams, ensuring the project's iterative delivery and continuous improvement through sprint reviews and retrospectives.
- ❖ Engineered a decentralized **NFT Car Racing Game** on the **Ethereum blockchain** using **Solidity** for smart contract development, enabling players to mint, trade, and race NFT cars. The game runs entirely on-chain, ensuring verifiable ownership of assets and race results.
  - Developed smart contracts for minting NFT cars with the **ERC-721** token standard, leveraging **OpenZeppelin contracts** for secure and modular smart contract development, ensuring each car is unique and verifiably owned by the player.
  - Implemented a **race simulation logic** in the smart contracts, using randomization techniques for race results based on factors like car performance and user participation, ensuring transparency and fairness in gameplay.
  - Built the front-end with **Vue.js**, utilizing **Vuex** for state management and **Vue Router** for routing, providing real-time interaction with the blockchain, displaying user profiles, race results, and car details.
  - Integrated **Web3.js** and **Ethers.js** for Ethereum interaction, enabling the game to connect with users' wallets (via **MetaMask** and **WalletConnect**), allowing secure minting, trading, and racing of NFTs.
  - Developed the car marketplace where users can buy and sell NFT cars, with dynamic pricing based on rarity and market demand, powered by smart contracts on the Ethereum network.
  - Ensured **gas efficiency** by optimizing smart contract functions, such as minimizing storage operations and using off-chain calculations for race results, to reduce transaction fees for users.
  - Incorporated **Chainlink VRF (Verifiable Random Function)** for generating random, verifiable race outcomes, ensuring fairness and preventing manipulation of race results.
  - **Optimized** user interactions by reducing the transaction latency on the frontend with **Vue.js**, ensuring that blockchain transactions were confirmed quickly without blocking the user interface.
  - Implemented **token rewards** for winning races, which are credited as **ERC-20 tokens** to the user's wallet, offering a play-to-earn mechanism within the game ecosystem.
  - Designed a **secure wallet connection flow** with **MetaMask** and **WalletConnect**, ensuring players' assets were securely stored and transactions were protected with private keys on the blockchain.

- ❖ Designed and developed a full-stack **Medical Appointment and Drone Delivery System** using the **MERN stack**, ensuring the platform's scalability and seamless user experience across all devices.
  - Developed the front-end with **React.js**, using **Hooks** and **Context API** for efficient state management, and **React Router** for handling dynamic route navigation, ensuring smooth navigation and state updates in the appointment scheduling process.
  - Implemented an intuitive calendar system for appointment booking with **React Big Calendar**, allowing users to select available time slots, schedule appointments, and automatically receive confirmation emails.
  - Built a real-time **drone delivery tracking system** using **WebSockets** and **Google Maps API**, enabling users to monitor the exact location of their medical deliveries in real-time, with live updates shown on the map.
  - Designed the **back-end** using **Node.js** and **Express.js**, implementing a RESTful API to handle user authentication, medical records, appointment scheduling, and drone delivery management.
  - Leveraged **MongoDB** for storing patient data, appointment details, and drone delivery statuses, using **Mongoose** for data validation and schema definition, ensuring data consistency and fast queries.
  - Implemented **JWT-based authentication** for secure user access and role management, enabling different access levels for doctors, patients, and delivery personnel to interact with the system securely.
  - Created **automated email and SMS notifications** for appointment reminders and delivery status updates using **Twilio API** and **Nodemailer**, ensuring timely communication with users.
  - Integrated a **third-party API** for drone control, enabling scheduling, dispatching, and real-time tracking of drones delivering medical supplies, leveraging **AWS Lambda** and **AWS S3** for backend storage of delivery data and media files.
  - Implemented **data security** by encrypting sensitive medical information stored in the database using **AES encryption** and ensuring HIPAA compliance for medical data handling.
  - Optimized **server performance** by implementing **load balancing** with **AWS EC2** and **Docker** containers, ensuring high availability and scalability to handle growing traffic and large datasets.
  - Automated **deployment** using **CI/CD pipelines** with **GitHub Actions**, ensuring smooth integration and delivery of new features with minimal downtime, while also implementing **unit testing** with **Jest** for API endpoints and front-end components.

## Web Developer, *Cody Web Development Inc*, 09/2016 - 03/2019

- Designed and developed multiple dynamic websites, delivering high-quality user experiences and scalable applications by focusing on both **frontend** and **backend** development.
- Built **interactive user interfaces** using **React.js** and **Vue.js**, integrating **Redux** and **Vuex** for state management, ensuring smooth and consistent user interactions across large-scale applications.
- Developed **RESTful APIs** and backend services using **Node.js**, **Express.js**, and **Django**, providing scalable solutions for data processing, user authentication, and business logic integration.
- Implemented **secure user authentication** and authorization systems using **JWT (JSON Web Tokens)**, **OAuth**, and **bcrypt** for password hashing, ensuring robust security for user data and API endpoints.
- **Created** full-stack solutions for various industries, including e-commerce, social networking, and business dashboards, utilizing **Express.js** and **Django** for backend development and **MongoDB/PostgreSQL** for efficient data storage.
- Developed and optimized backend systems for handling product management, user profiles, shopping carts, order processing, and notifications, focusing on scalability and performance with **Node.js/Express.js** and **Django**.
- Optimized **database queries** and server performance using **SQL** (for **PostgreSQL**) and **NoSQL** (for **MongoDB**) solutions, implementing **query optimization**, **database indexing**, and **caching** with **Redis** to reduce latency and improve response times.
- Created and maintained **custom CMS** platforms and admin dashboards, providing clients with intuitive interfaces to manage their content, user interactions, and reports.
- Collaborated closely with frontend teams to create responsive and intuitive designs using **CSS3**, **Flexbox**, and **CSS Grid**, ensuring cross-browser compatibility and mobile-first design principles.
- Integrated **third-party services** like **Stripe**, **PayPal**, and **Twilio** for payment processing, SMS notifications, and real-time chat functionality, enhancing website features and user engagement.

- Deployed **applications** using **AWS**, **Heroku**, and **DigitalOcean**, and containerized services with **Docker** for consistent development, testing, and production environments.
- Managed **version control** using **Git** and **GitHub**, coordinating with cross-functional teams through pull requests, code reviews, and efficient branch management practices.
- Enhanced **web performance** by implementing **lazy loading**, **code splitting**, and **minification** for CSS and JavaScript files, improving page load times and user experience.
- Applied **SEO best practices** by optimizing metadata, implementing schema markup, and ensuring content accessibility to improve search engine visibility and site traffic.
- Monitored **site performance** and user analytics through **Google Analytics**, **New Relic**, and custom logging solutions, identifying issues, and optimizing code for better performance and user engagement.
- Provided **ongoing maintenance** and support for deployed websites, debugging issues, adding new features based on client feedback, and ensuring system stability and security over time.

## Software Developer Internship, *Digital8*, 02/2016 - 08/2019

- Assisted in **code reviews** by providing feedback on code quality, maintainability, and best practices.
- Wrote and maintained **unit tests** using **Jest** and **Mocha**, ensuring software stability and functionality.
- Participated in **manual and automated testing**, identifying bugs and collaborating with the development team to resolve issues.
- Contributed to **debugging** and troubleshooting production issues, improving software performance.
- Collaborated with the QA team to ensure comprehensive test coverage and validate new features.
- Gained experience with **version control** using **Git** and participated in team discussions to improve coding practices.

## Education

### The University of Western Australia, 02/2011 - 11/2015

#### *Bachelor's Degree in Computer Science*

- Top **1%** performer of department.
- Participated in department coding research & development for **2** years.
- Successfully delivered **20+** tasks for different global clients on **Freelancer.com** and **Upwork**.
- **GPA ( 6.5 / 7.0 )**

## Certification

- Solidity Smart Contracts: Build DApps In Ethereum Blockchain - Udemy
- Ethereum Smart Contract Programming 201 - Moralis Academy
- Javascript General Coding Assessment - CodeSignal
- Full-Stack Web Assessment - Angel

## Projects

### 1. DeFi Ecosystem

Developed a decentralized finance (DeFi) ecosystem, including a lending protocol and token swaps, on the **Ethereum** blockchain. Integrated **ERC20** token contracts for asset management, liquidity pools, and **Chainlink Oracles** for real-time pricing data. Enabled users to lend, borrow, and swap assets on the platform with full transparency and decentralized governance.



**Technical Skills:** Solidity, Ethereum, Truffle, Web3.js, ERC20, Chainlink Oracles, JavaScript, Smart Contracts, Decentralized Finance (DeFi).

## 2. On-Chain Oracle and DeFi Lending Protocol

Built an on-chain oracle solution and a **DeFi lending protocol** for decentralized finance applications on **Ethereum**.

Developed smart contracts to facilitate secure and trustless borrowing/lending, while integrating **Chainlink** to fetch reliable off-chain data such as interest rates and collateral values. Ensured high security and decentralized data feeds for real-time adjustments to the lending protocols.

**Technical Skills:** Solidity, Ethereum, Chainlink, Smart Contracts, Decentralized Finance (DeFi), Oracles, JavaScript, Blockchain Development.

## 3. Test Case Management System

Developed a **Test Case Management System** to streamline the tracking, execution, and reporting of test cases for software development teams. Implemented a user-friendly interface with **Vue.js** to create, organize, and manage test cases, while integrating backend APIs for test case storage and management. Enabled automated reporting of test results and tracked the progress of testing in real-time.

**Technical Skills:** Vue.js, JavaScript, Node.js, Express.js, MongoDB, RESTful APIs, Unit Testing, Git, HTML5, CSS3, Software Testing.

## 4. NFT Car Racing Game

Created an interactive **NFT-based car racing game**, allowing players to mint unique NFT cars and race them in a decentralized, on-chain environment. Built using **React.js** for the frontend and **Ethereum** for the backend, leveraging **Web3.js** to interact with smart contracts for car ownership and race outcomes. Players could trade cars, customize them, and participate in live races, with in-game rewards stored as **ERC721 NFTs**.

**Technical Skills:** React.js, Solidity, Ethereum, Web3.js, NFTs, JavaScript, HTML5, CSS3, Blockchain Development, Smart Contracts.

## 5. Medical Appointment and Drone Delivery System

**Description:** Developed an integrated **Medical Appointment and Drone Delivery System** to streamline patient care and medical deliveries. Built with the **MERN stack**, the system allows users to schedule medical appointments and track deliveries of medical supplies via drones. Real-time appointment scheduling, GPS tracking, and notifications were integrated into the platform. Leveraged **AWS** to deploy the backend and manage cloud storage for user data and appointment records.

**Technical Skills:** MongoDB, Express.js, React.js, Node.js, AWS, API Integration, JavaScript, Drone Technology, Real-time Applications, Cloud Deployment.