

Md Imran

[mdimran29](#) | [MdImran](#) | dev.mdimran@gmail.com | [+91 8276904977](#)

EDUCATION

Netaji Subhash Engineering College
B.Tech in Information Technology

2022 – 2026
GPA: 7.5

EXPERIENCE

Petition.io — Blockchain Engineer Intern (Remote, Washington) June 2025 – Present

- Engineered staking contracts in **Solidity** and **Foundry**, enabling users to stake **ETH** for **PTNR** and lock **PTNR** for yield-bearing **stPTNR** governance tokens.
- Integrated **Rocket Pool** with **Chainlink Automation** to route staking rewards into a **quadratic funding pool**, users can donate rewards and receive verifiable **NFT tax receipts**.
- Implemented a **Uniswap V3** swap path to convert **rETH** to **ETH** during **PTNR** withdrawals for efficient liquidity management.
- Built an **ERC721**-based system to mint NFT tax receipts and issue governance tokens to protocol liquidity providers.

PROJECTS

DeFi Stablecoin (DSCEngine) — Solidity, Foundry, Chainlink, OpenZeppelin, Ethereum [Link](#)

- Built a decentralized stablecoin engine (**DSCEngine**) that mints **USD-pegged stablecoins (DSC)** using **WETH/WBTC** as collateral.
- Implemented **over-collateralization (200%)** to ensure asset backing and prevent insolvency.
- Integrated **Chainlink Price Feeds** (via **Chainlink Brownie Contracts**) for real-time asset valuation and liquidation thresholds.
- Designed a **liquidation system** with a **10% bonus** for liquidators.
- Achieved **85%+ test coverage** using **Foundry's fuzz testing**, ensuring contract reliability and minimizing failure points.
- Deployed the project on **Sepolia Testnet** using **Foundry scripts**, optimized gas costs, and verified smart contracts.

NFT Token Creation — Solidity, Foundry, OpenZeppelin, IPFS, Ethereum [Link](#)

- Built custom **ERC721 NFTs** (**MoodToken**, dynamic SVG-based; **DogToken**, PNG-based) with metadata stored on **IPFS**.
- Applied **Foundry-DevOps** for automated deployment, testing (**90% coverage**), and verification on **Sepolia Testnet**.

Foundry-Lottery — Solidity, Foundry, Chainlink VRF & Automation [Link](#)

- Developed a **smart contract-based lottery system** using **Foundry** for development and testing.
- Integrated **Chainlink VRF** for provably fair random winner selection and **Chainlink Automation** for automated prize distribution.
- Achieved **80%+ test coverage** using Foundry's testing framework, ensuring contract reliability.
- Deployed on **Sepolia Testnet**, allowing real-world blockchain interactions and automated execution.

VOLUNTEERING

Blockchain Team Member, GDSC NSEC October 2024 - Present

- Organized and supported multiple **blockchain workshops**, including a major event funded by **Avalanche**, engaging over **350 students** in **smart contract development** and **hackathon projects**.

TECHNICAL SKILLS

Programming Languages: **JavaScript, TypeScript, Solidity, Python, C++, HTML**
Libraries/Frameworks: **ExpressJs, ReactJs, Tailwind CSS, MongoDB, NextJs, Etherjs**
Version Control: **Git, Github, Gitlab**
Tools/ Platforms: **Linux, AWS, Nodejs, Alchemy, Foundry, Chainlink VRF, Chainlink Automation, OpenZeppelin, IPFS, Rocket Pool, Uniswap V3**