Milestone	**Source of Truth**	**Deadline (post- approval)**
Planning and Architecture	Define Project Scope: Clearly outline the features and functionality of RaumFi DEX. Market Research: Analyze existing blockchain state and identify unique features for DEX. Conduct UID analysis. Technical Architecture: Design the overall technical architecture of RaumFi DEX. Choose the appropriate setup for deployment (Server/IPFS) and smart contract language. Team Formation: Assemble a development team with expertise in blockchain, smart contracts, frontend and backend development.	Month 1

Smart Contract Developmen	Smart Contract Prototyping: Develop prototype smart contracts for essential functionalities. Begin testing on a local blockchain env. Security Audits: Conduct initial security audits for the prototype smart contracts. Address any vulnerabilities identified. Token Integration: Integrate token standards (e.g., ERC-20, ERC-721) for trading and liquidity provision. Implement token swap functionalities. Initial Testing: Deploy smart contracts to a testnet for comprehensive testing. Test token swaps, liquidity provision, and other core	Month 2-3

Frontend and Backend Development	Frontend Development: Start developing the user interface (UI) for RaumFi DEX. Implement features for seamless token swaps, order book display, and user account management. Backend Integration: Develop the backend infrastructure to support the frontend. Integrate with the chosen blockchain platform. Wallet Integration: Integrate wallet functionalities, allowing users to connect with MetaMask or other wallets.	Month 4

Testing, Optimization, and Deployment	User Testing: Conduct extensive user testing on the testnet to identify and address any user experience issues. Optimization: Optimize smart contract code and frontend/backend performance. Address any security vulnerabilities or issues identified during testing. Deployment to Mainnet: Deploy RaumFi DEX to the mainnet. Announce the launch and provide documentation for users. Community Outreach: Engage with the community, solicit feedback, and address any post-launch issues. Implement any necessary updates or improvements based on user feedback.	Month 5