

Time Plan

From August 1 to September 25

	August 1 to August 21	August 22 to September 1	September 2 to September 15	From September 16 to September 25
Tasks during this period	Content Verification MVP	Smart Contract Integration	External Integration & Enhancements	Final Testing & Submission
	<p>1 Building a basic interface with Next.js.</p> <p>2 Bind identity using ICP Identity and Web3 wallets.</p> <p>3 Set up a fingerprint storage architecture on the ICP blockchain.</p> <p>4 Create a Hash for each piece of content (image/video) using an appropriate algorithm.</p> <p>5 Capture a user profile picture and associate it with the content.</p> <p>6 Store the Hash on the blockchain as a direct digital identifier (DID).</p>	<p>1 Write smart contracts using Motoko to store and verify content.</p> <p>2 Connect smart contracts to the interface using Agent.js and the ICP SDK.</p> <p>3 Develop an API to interact with contracts (addContent, VerifyContent, GetHash).</p> <p>4 Test contracts and verify integration between the UI and the backend.</p>	<p>1 SDK integration for integration with Twitter/Facebook or any other supporting platform.</p> <p>2 Support sharing and authentication of content from external sources.</p> <p>3 Design of a "Share and Verify" interface.</p> <p>4 Improved overall performance, responsive UI support, and enhanced user experience.</p> <p>5 Enhancement of all support and issues.</p>	<p>1 Test everything, from login to content validation.</p> <p>2 Fix any final bugs and improve performance.</p> <p>3 Prepare your final presentation and demo.</p> <p>4 Prepare the required files for delivery.</p>
An organized GitHub repository containing	<input type="checkbox"/> Next.js project <input type="checkbox"/> Identity binding files <input type="checkbox"/> Fragment generation functions <input type="checkbox"/> Deployment interface screenshots and documentation <input type="checkbox"/> A PDF or README document explaining the flow documentation	<input type="checkbox"/> Motoco Smart Contract Files <input type="checkbox"/> Simple API Documentation (Even Markdown or Postman) <input type="checkbox"/> Short Interface and Contract Interaction Demonstration Video <input type="checkbox"/> UI Validation Page	<input type="checkbox"/> SDK Code Integration + User Guide <input type="checkbox"/> New UI Screens for External Linking <input type="checkbox"/> Share and Verify Experience Video <input type="checkbox"/> Bug Fixes and Technical Notes	<input type="checkbox"/> PowerPoint or Google Slides for final presentation <input type="checkbox"/> Comprehensive demo video (3–5m) <input type="checkbox"/> Updated GitHub Repo + full README <input type="checkbox"/> Project documentation PDF

Supporting sources

- Official Node.js Documentation
<https://nodejs.org/en/docs/>
- Official Web3.js Documentation
<https://web3js.readthedocs.io/>
- Official Solidity Documentation
<https://docs.soliditylang.org/>
- Official Ethereum Website
<https://ethereum.org/>
- Medium Educational Articles on Web3 and Blockchain Technology
<https://medium.com/>
- Global Hackathons: Devpost/DoraHacks
<https://dorahacks.io/>
- Official Figma Design Library
<https://www.figma.com/resources/learn-design>

Team instructions

Commitment to each member's assigned tasks according to the schedule.

Any code changes must be approved and reviewed by the leader before merging.

Update work on GitHub.

Weekly meetings are mandatory to update progress; **no excuses are accepted**.