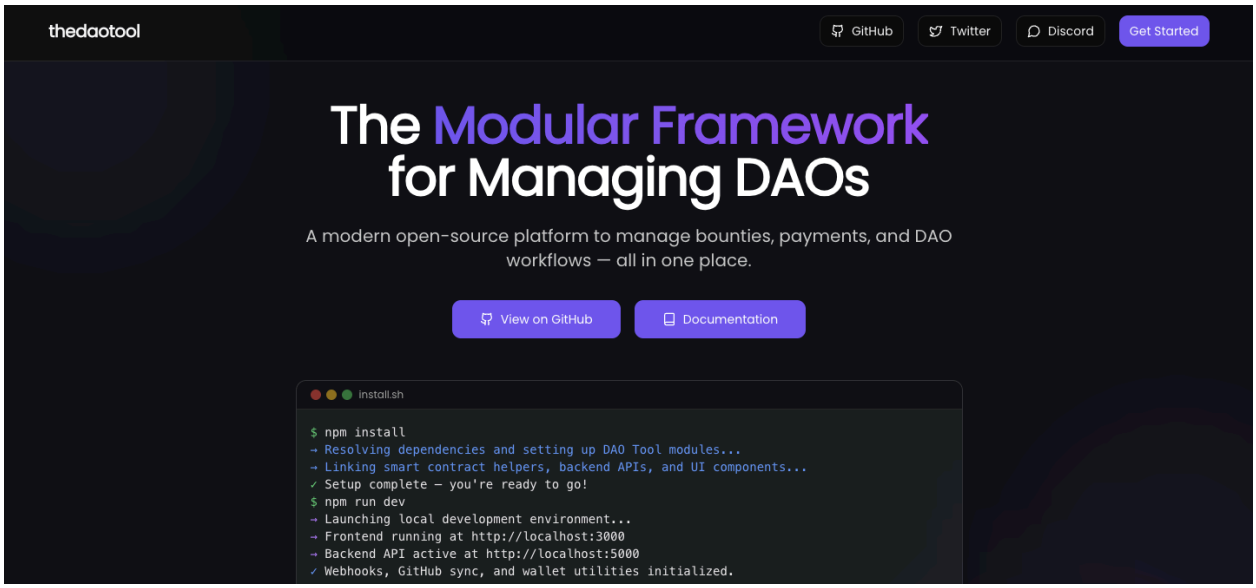
The DAO Tool – Project Goals & Milestones (Project Documentation)



Project Overview

The DAO Tool is an open-source platform designed to streamline how DAOs manage contributors, bounties, and payments. It brings together key workflows — like GitHub issue syncing, Discord automation, wallet-based payouts, and real-time collaboration — into one modular, extensible system. With built-in authentication, role-based dashboards, and support for Solana/USDC via Privy, it's tailored for modern, async teams. Whether you're running a DAO, an open-source collective, or a contributor network, The DAO Tool helps you move from task to payout with zero hassle.

Platform	Link
Website	https://www.thedaotool.com
DAO Application	https://app.thedaotool.com
Documentation (Under Development)	https://docs.thedaotool.com
GitHub	https://github.com/kartikmehta8/thedaotool
X	https://x.com/thedaotool
Discord	https://discord.gg/VsyDp52Saq
Telegram	https://t.me/thedaotool
YouTube Demo	https://youtu.be/eG6GO4EMYjc

(Would **RECOMMEND WATCHING THE DEMO** once for gaining a clear picture of the system!)

Milestones (v1)

1. **Post New Bounties to Discord Channel** **[DONE]**

- Integrate Discord via bot or webhook to automatically post new bounties to a selected Discord channel.
- Include bounty title, tags, and direct link for easy access.
- Add admin toggle to enable/disable Discord sync.
- **Goal:** Boost visibility and engagement by keeping the community updated in real-time.

2. **Authorize GitHub Repositories via Profile Page** **[DONE]**

- Allow organizations to authorize their GitHub repositories directly from their profile page using OAuth.
- Automatically sync issues labeled '*dao*' from the repo and post them as open bounties on the platform using an async background task.
- **Goal:** Streamline bounty creation by letting organizations manage work through GitHub.

3. **Modularize Frontend Codebase** **[DONE]**

- Split large components into smaller, manageable modules.
- Create a component library for reusable elements such as modals, buttons, and forms.
- **Goal:** Easier maintenance, faster iterations, and streamlined future development.

2. **Abstract Firebase Calls & Build API Layer** **[DONE]**

- Set up an Express/Node.js backend server to handle all business logic.
- Replace direct Firebase interactions with REST APIs to improve security and scalability.
- Implement role-based access control to restrict sensitive operations.

- **Goal:** More secure and optimized backend operations.

3. **Allow Multiple Contributors on a Single Bounty** **[TBD]**

- Modify bounty schema to allow multiple contributors to apply and collaborate.
- Implement task splitting and assignment tracking.
- Enable payment distribution among multiple contributors based on task completion.
- **Goal:** Expand contributor capabilities to handle complex projects.

4. **Show Past Payments to Organizations** **[DONE]**

- Create a payments dashboard that tracks previous payments, bounty status, and payout history.
- **Goal:** Increase transparency for organizations and allow better financial tracking.

5. **Implement Email Notifications for Key Events** **[DONE]**

- Send email notifications to organizations and contributors for:
    - Bounty assignment
    - Task submission
    - Payment confirmation
- **Goal:** Keep both parties informed and reduce dependency on manual tracking.

6. **Smart Bounty Matching Based on Skills** **[DONE]**

- Create a recommendation engine:
  - Matches contributors to bounties based on overlapping tags and skills.
  - Prioritizes bounties that align with the contributor's expertise.
- **Goal:** Ensure that contributors only see relevant bounties, increasing task relevance and efficiency.

7. **Integrate Solana Wallet & On-Chain Payments** **[DONE]**

- Integrate Solana Wallet using Privy to allow secure on-chain payments via USDC or SPL tokens.
- **Goal:** Reduce payment friction and add on-chain transparency.

8. **Transform The DAO Tool into a Versatile Template** **[DONE]**

- Refactor the platform to support custom roles and fields, allowing adaptation to various use cases:
  - Investors & Projects – **Fundraising platforms**
  - Organization & Contributors – **DAO Model (Current)**
- **Goal:** Create a flexible and scalable template that can be adapted for multiple industries.

Other processes include setup of server and CI/CD pipeline. **[DONE]**

Milestones (v2)

1. **Fix Responsiveness on Mobile** **[DONE]**

- Audit the entire application for layout issues on mobile and tablet devices.
- Ensure all major components (modals, forms, navbars) support Ant Design breakpoints and are fully responsive.
- Resolve scroll, padding, and alignment issues to ensure usability on small screens.
- **Goal:** Ensure a consistent and user-friendly experience across all devices, improving accessibility and engagement.

2. **Add Analytics with Charts** **[DONE]**

- Introduce a dedicated analytics section in the dashboard to visualize platform activity.
- Use charts to display trends in bounty creation, completion rates, and contributor engagement.
- Implement backend aggregation endpoints to support real-time data visualization.
- **Goal:** Provide organizations with data-driven insights to track project health and contributor performance.

3. **More Granularity in Editing Bounty** **[DONE]**

- Extend editing capabilities for organizations to include all bounty fields, not just amount or status.
- Update the ViewBountyModal to allow inline editing of name, description, deadline, tags, and issue metadata.
- Update Joi validation schemas and ensure all form changes are properly saved to the database with live feedback.
- **Goal:** Improve project management flexibility and reduce admin dependency by empowering organizations to fully manage their bounties.

4. **Improve UI/UX Across Key Flows** **[DONE]**

- Conduct a UI/UX audit to identify and fix inconsistencies in layout, spacing, font usage, and interaction patterns.
- Update core flows such as bounty creation, profile management, and payment actions with cleaner design and smoother transitions.
- **Goal:** Deliver a polished, intuitive experience that makes the platform feel reliable, modern, and easy to use for both contributors and organizations.

5. **Add "Forgot Password" Support** **[DONE]**

- Introduce a secure and user-friendly password recovery flow to improve account accessibility.
- Include token or OTP-based validation, password confirmation, and secure password update on success.

- **Goal:** Strengthen user account recovery and improve trust in the platform's security and usability.

6. **Email Verification for Profiles** **[DONE]**

- Introduce email verification for both organizations and contributors to enhance platform trust and communication reliability.
- Allow users to trigger verification from their profile and confirm identity via a time-limited token link sent to their inbox.
- **Goal:** Build trust and improve delivery of critical notifications like payments, assignments, and submissions.

7. **Separate Queues for GitHub Sync & Background Jobs** **[DONE]**

- Isolate long-running and time-sensitive background operations into dedicated queues for improved performance and reliability.
- Eliminate contention between GitHub sync, OTP cleanup, and email dispatch by running them in parallel using independent workers.
- **Goal:** Create a scalable, resilient background processing architecture that supports real-time and scheduled tasks independently.

8. **Guided User Tour for Onboarding** **[IN-PROGRESS]**

- Introduce a step-by-step interactive tour to help new users understand the platform's layout and key actions.
- Highlight core features such as bounty creation, profile setup, and GitHub/Discord integrations with overlays and navigation controls.
- **Goal:** Reduce friction for first-time users, encourage feature discovery, and improve activation rates with a clear, structured onboarding experience.

9. **Document the Entire Project on GitBook** **[IN-PROGRESS]**

- Set up structured, accessible, and continuously updated documentation on The DAO Tool Docs.
- Cover core areas including architecture, environment setup, backend/frontend guides, API references, integrations, and contribution standards.
- **Goal:** Empower new developers and contributors to onboard quickly, understand the architecture, and contribute confidently.

Goals (v3)

1. **Super Admin Panel for Full Platform Control** **[PLANNED]**
- Add a dedicated super admin role and dashboard interface to view, edit, and manage all data including contributors, organizations, bounties, and payouts.
- **Goal:** Ensure complete control and oversight of the system for moderation and support.

2. **Implement Robust Logging System** **[PLANNED]**
- Set up centralized structured logging for backend actions like bounty creation, GitHub sync, and payment events using tools like Winston or Pino.
- **Goal:** Improve traceability, debugging, and audit capability of the backend.

3. **Add Monitoring and Health Checks** **[PLANNED]**
- Integrate health monitoring endpoints and alert systems for uptime, queue failures, and API error spikes using tools like UptimeRobot or Grafana.
- **Goal:** Proactively detect and resolve downtime or critical failures.

4. **Enhance Accessibility with ARIA and Semantic HTML** **[DONE]**
- Improve the UI by adding ARIA labels, semantic elements, and keyboard navigation to support screen readers and meet WCAG 2.1 compliance.
- **Goal:** Make the platform accessible and inclusive for all users.

5. **Write Unit Tests for Backend** **[PLANNED]**
- Add a comprehensive test suite for critical backend routes using Jest and Supertest.
- **Goal:** Improve code reliability, prevent regressions, and enforce test-driven development.

Final

By achieving these milestones, The DAO Tool will become a highly adaptable and secure platform that organizations and developers can customize and deploy in various industries. **The long-term vision is to build an open-source ecosystem that will facilitate the collaboration between two parties (like organization and contributor) seamlessly with managing their payment and operations.**

Note from the Developer

There is currently no template in place for companies to seamlessly outsource tasks, initiate bounties related to various needs, and process payments efficiently. This project has the potential to be expanded for numerous use cases **(IT WILL BE AN OSS WITH MIT LICENSE)**, and if successfully executed, it will be even safer than being just a loosely coded product.

It will remain open-sourced and maintained not merely as a project but as a fully-fledged template that can be utilized by anyone who resonates with the spirit of this product.

Do watch the video (even at 2x) for the project's working. :)

Kartik Mehta

X / LinkedIn / GitHub