SAAMAJSEVA

An NGO–Donor Connect Platform

TEAM META MINDS
23251A05H5- J SAHITHI
23251A05J0-KEERTHANA
23251A05J6-SAI SHALINI

PROJECT OVERVIEW

To bridge the gap between NGOs seeking assistance and donors looking for verified causes. The platform aims to democratize charitable giving by making it easy for any individual to find and contribute to projects that align with their values.

Key Features:

- Dual User Interface: Separate dashboards and authentication flows for Donors and NGO representatives.
- NGO & Project Listing: A clear, navigable directory displaying active projects, funding goals, and organization details.
- Contribution Interface: A guided, secure user flow for making donations to specific causes.
- Scope / Target Users: NGOs (managing their profiles and projects), Individual Donors (contributing funds), and the platform's administrators.

REACT TOPICS & TECHNIQUES USED

Components & JSX:

- Built the entire UI using reusable functional components for better readability and maintainability.

State Management (useState):

- Used extensively to manage local state form data, UI toggles, and feedback messages.

Event Handling:

- Event Handling manages user interactions, such as using onSubmit in Auth.jsx to trigger the login logic, or onClick in Navbar.jsx to handle user logout.

Client-Side Routing (React Router):

- Routes implemented for /home, /ngos, /donate/:id, /auth, and /dashboard.Enables smooth navigation without full page reloads.

REACT TOPICS & TECHNIQUES USED

Conditional Rendering:

- Dynamically displays login prompts, loading messages, or dashboard components based on user state.

Forms & Controlled Inputs:

- Managed registration and donation forms with controlled components for validation and data handling.
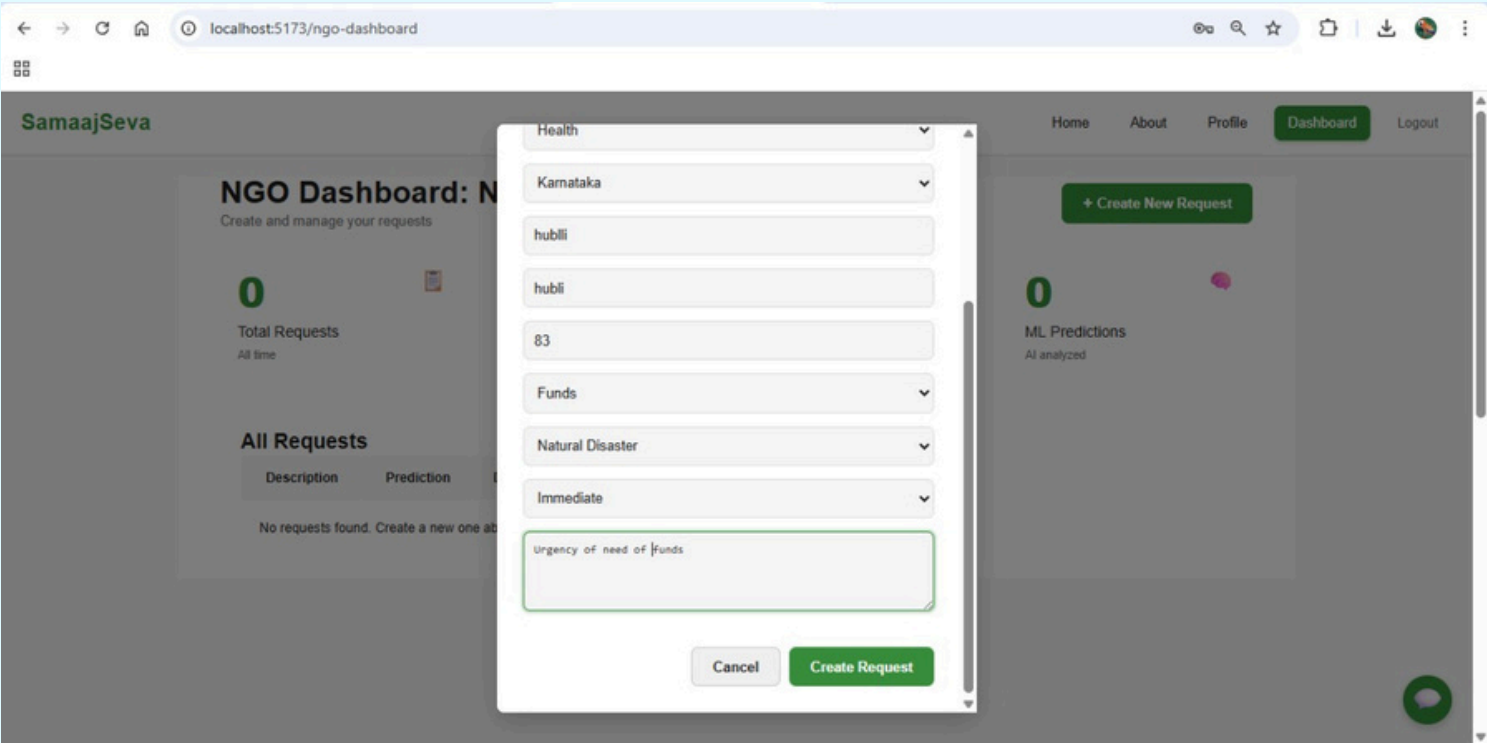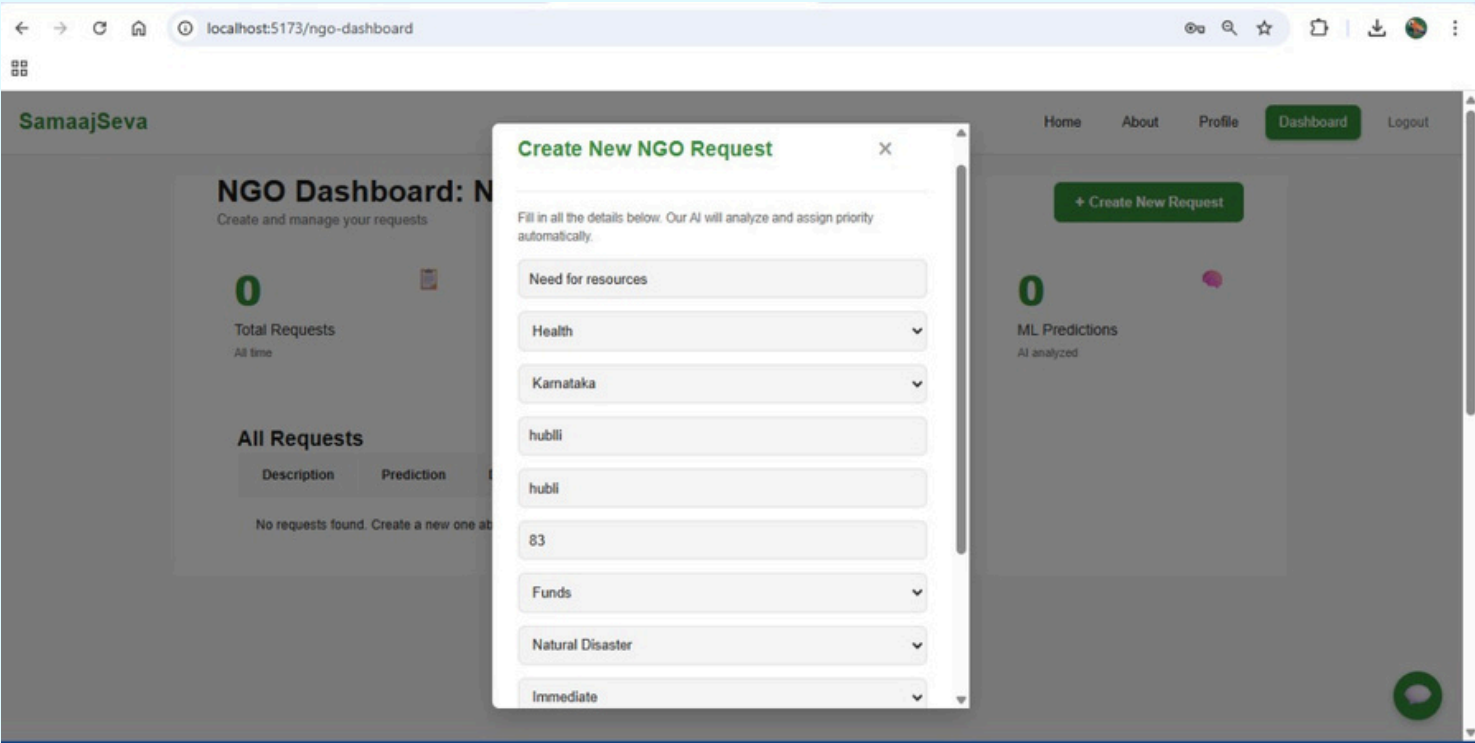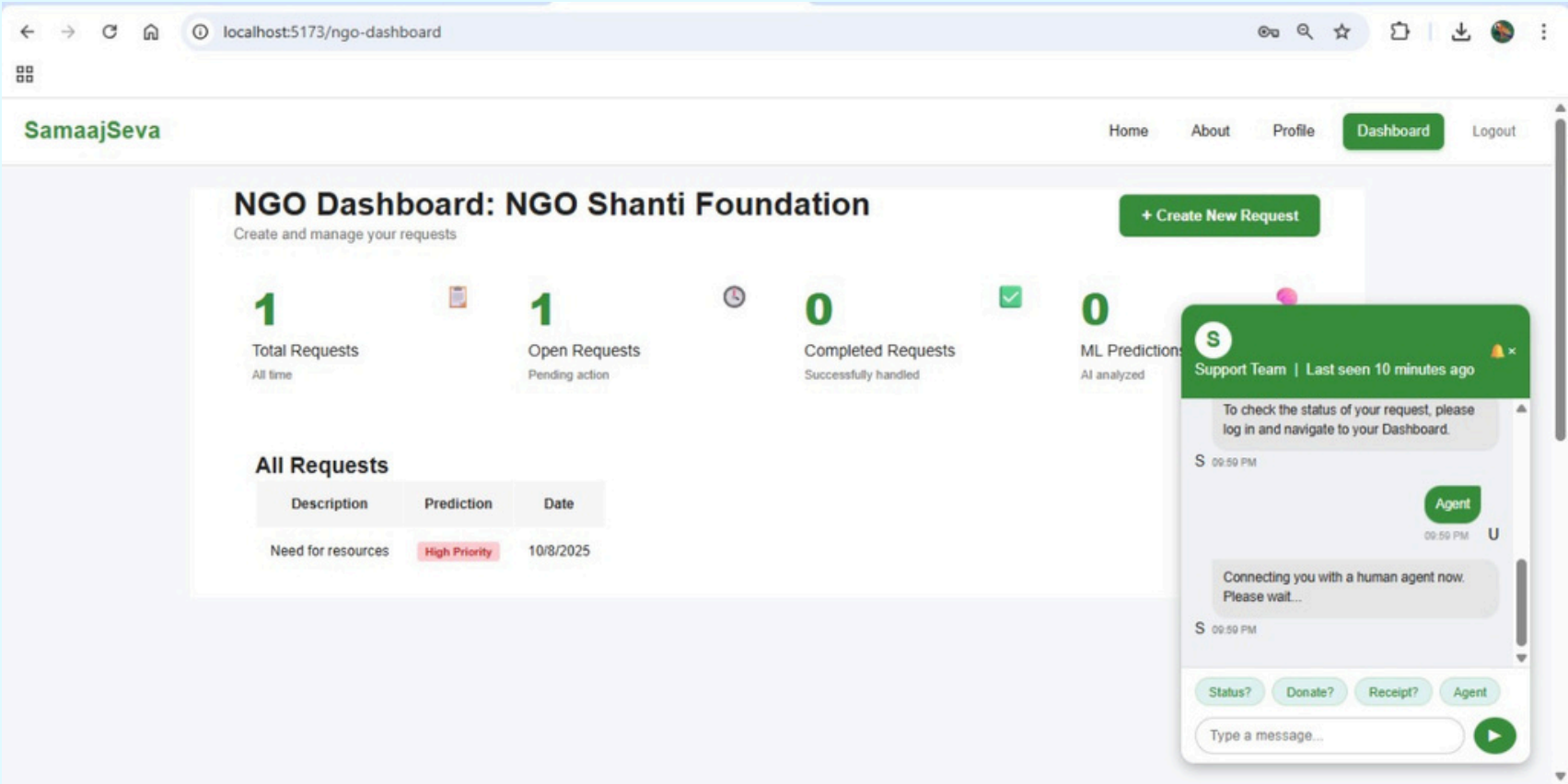
Chat / Messaging Feature (ChatPopup):

- Implemented a ChatPopup component to enable message input and display within a popup interface.
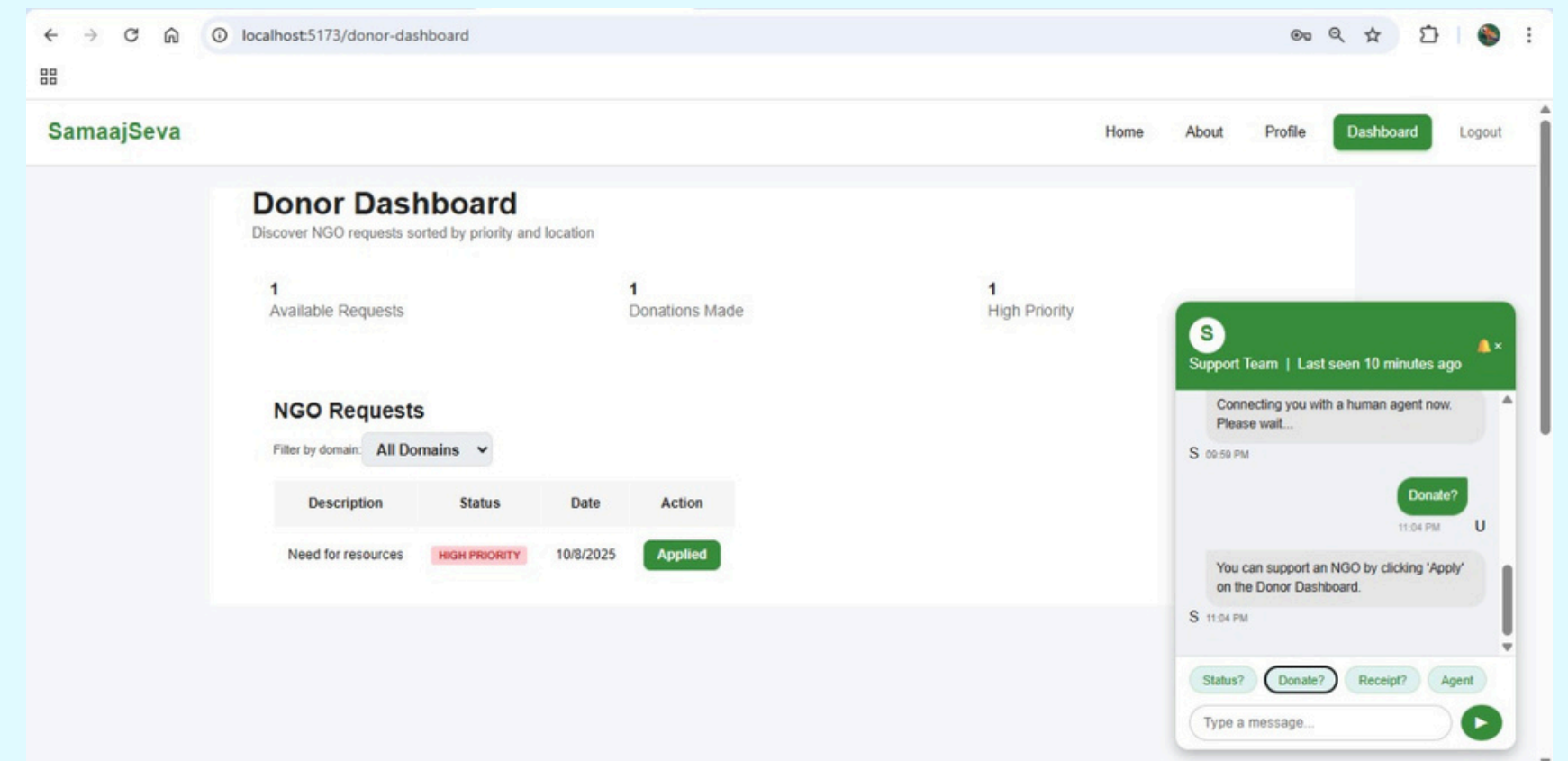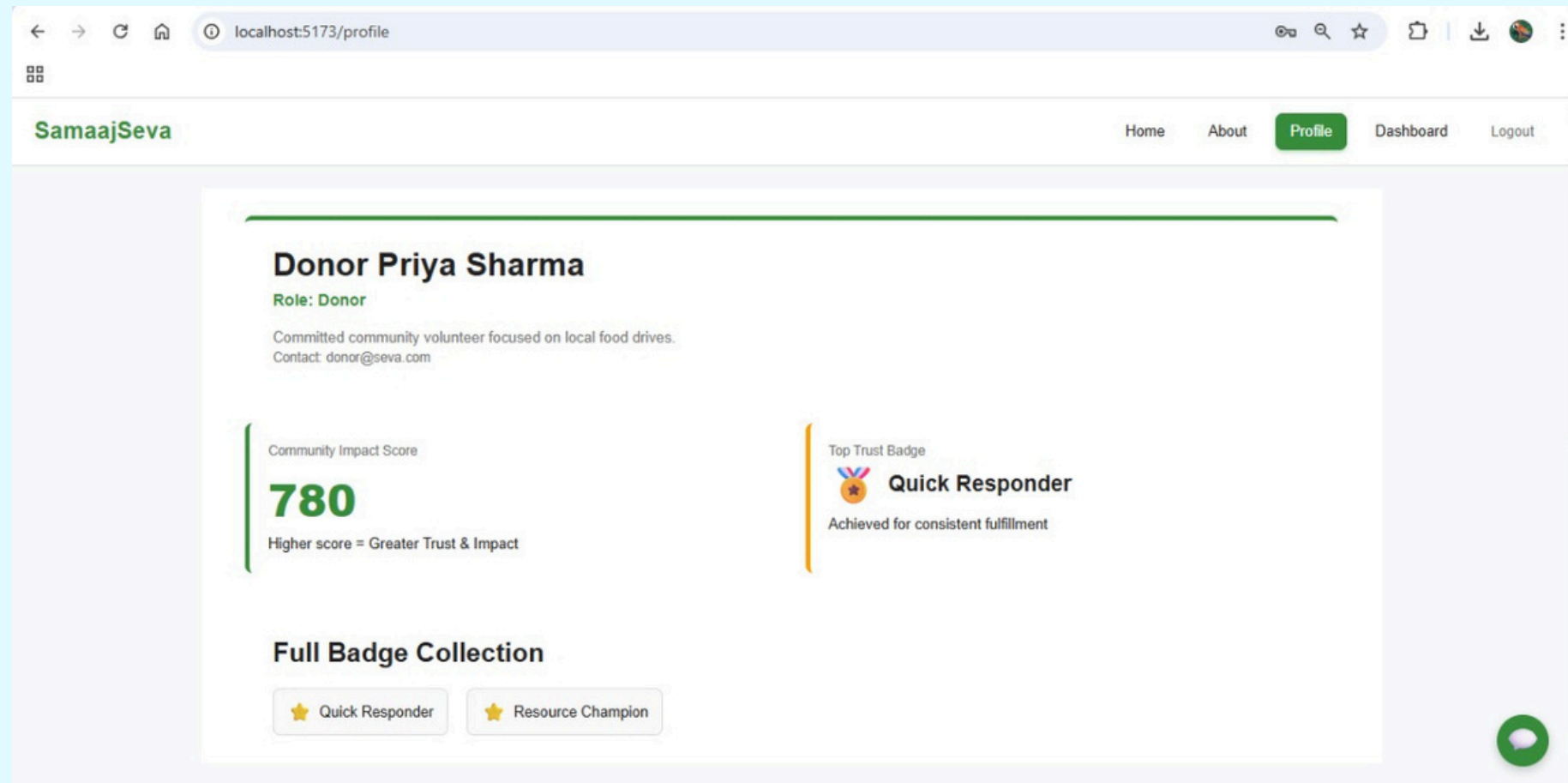- Utilized React's useState hook to manage chat messages and input field states.

OUTPUTS
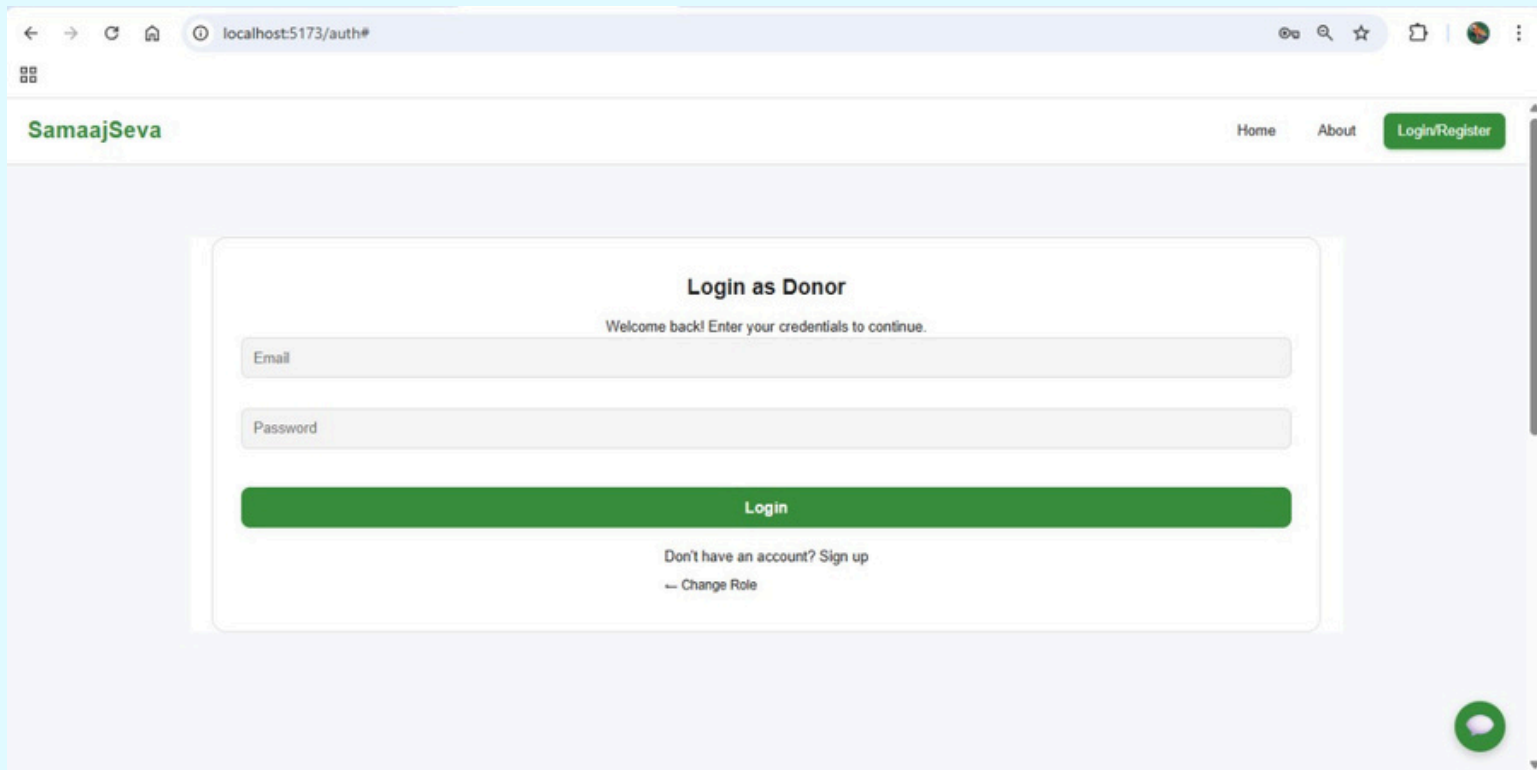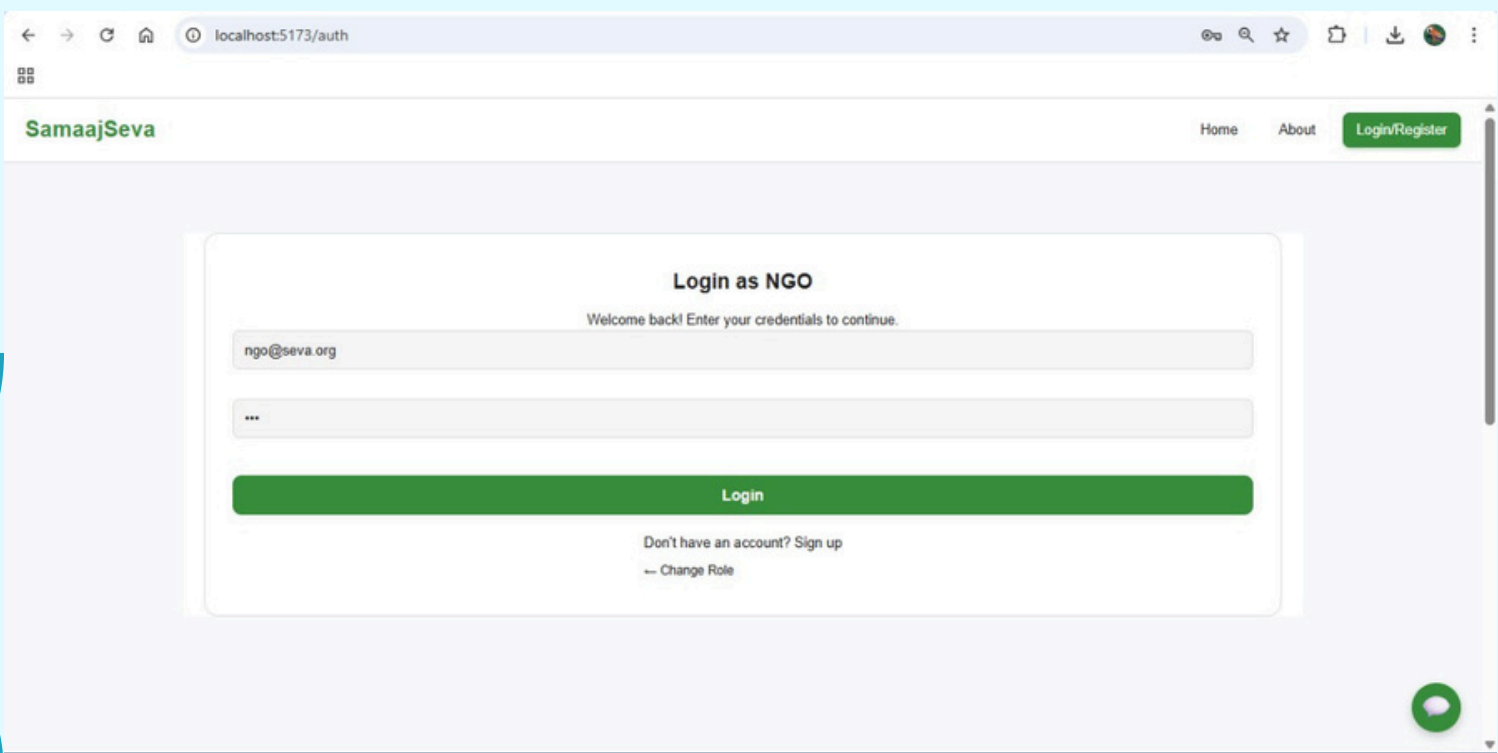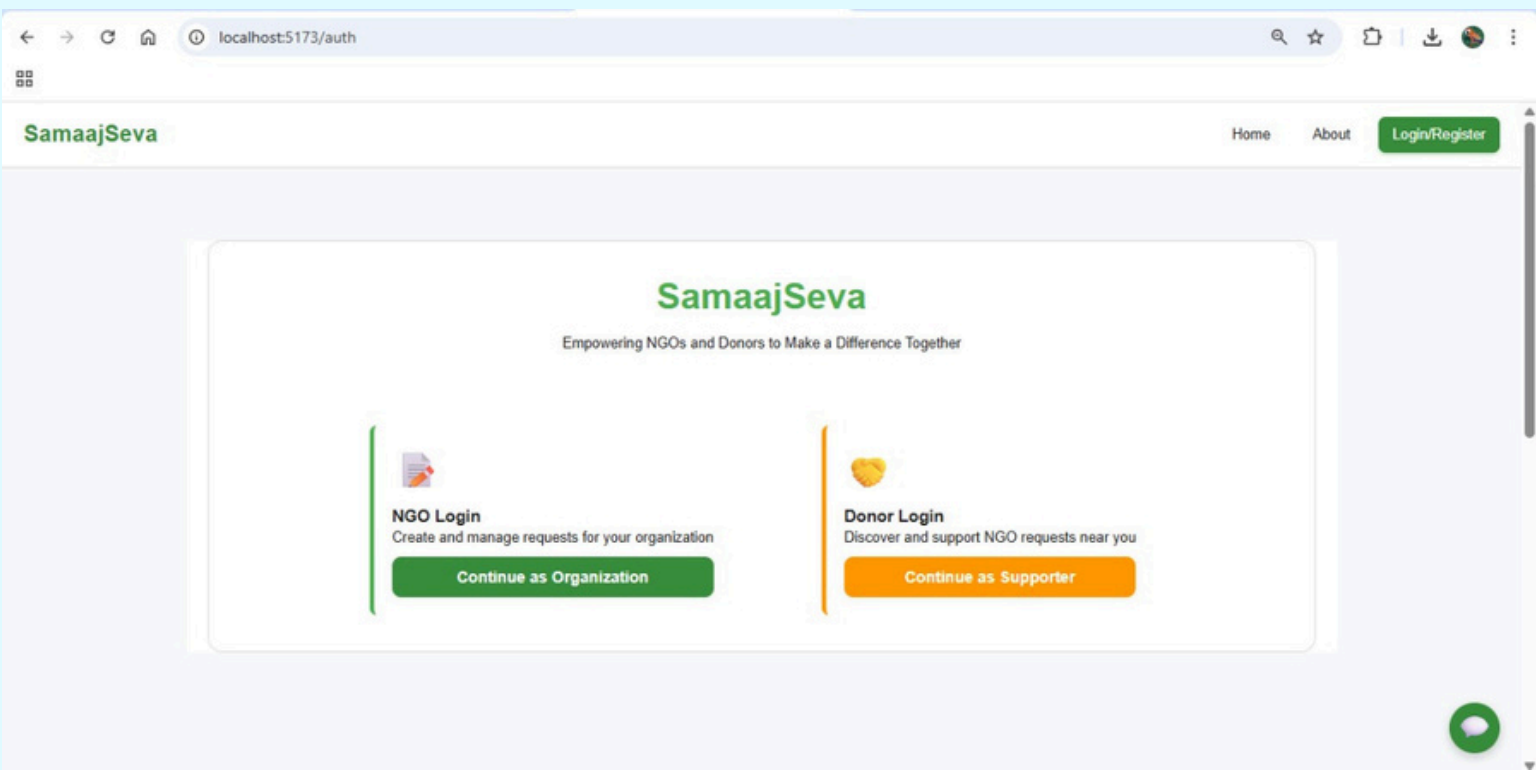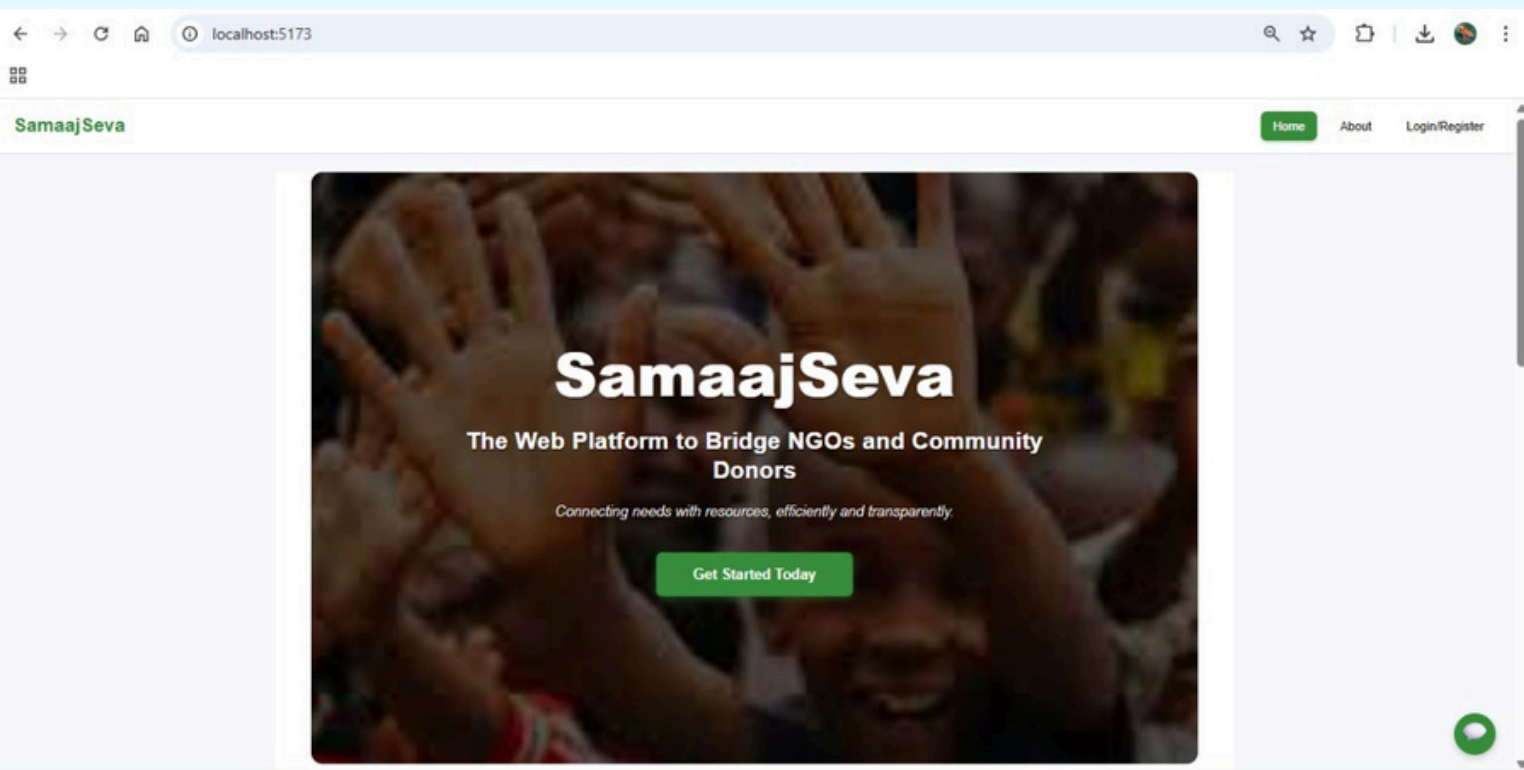
NGO DASHBOARD

DONOR DASHBOARD

LEARNING OUTCOMES

- Gained strong, practical skills **using functional components to build the UI** and **event handling to manage all user interactions**.

- Learned how to **use the useContext hook** to create a central place for important application data, making it easy to share information across all pages.

- Successfully implemented user session maintenance using localStorage, so users stay logged in even if they refresh the page.

- Designed the project **using a Component-Driven Architecture**, which means the code is highly organized, reusable, and easy to read.

- Developed proficiency in **making the application dynamic & responsive**, including conditional rendering to show personalized content (like dashboards) and integrating complex CSS to ensure the app looks professional and works well on all screen sizes.

- Gained experience building real-time functionality by **creating the Chat/Messaging feature**, learning how to manage quick-changing data for live updates.

THANK YOU