COMPUTER SCIENCE · CAPSTONE 2 · SUMMER 2026

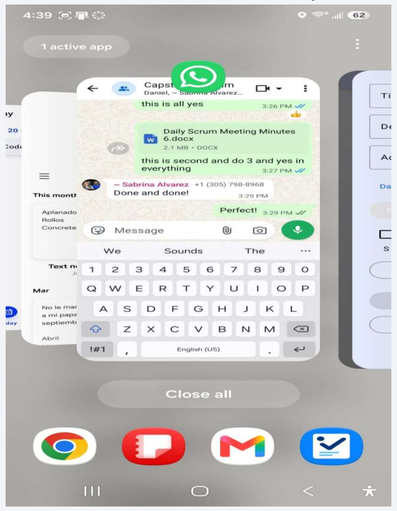
Multi

Team members: Samuel Pino · Daniel Pino · Sabrina Alvarez · Instructor: Prof. Masoud Sadjadi

FIU | FLORIDA INTERNATIONAL UNIVERSITY

Problem

- Students and small teams manage project updates, tasks, notes, and events using many apps.
- Multiple apps lead to confusion, loss of data, inefficient work, and it becomes impossible to communicate properly.
- Teams need effective ways to organize themselves without spending a lot of money on multiple subscriptions.
- Multi can reach multiple devices and complete multiple project tasks.
- Project work was scattered across Calendar, Notes, WhatsApp, Tadu, and Multi, making organization harder than it needed to be.
- We have been struggling with this problem for a long time and the screenshot shows 4 of the 5 apps open at the same time to do what Multi now does in one place.



Stakeholders & Requirements

- *Stakeholders are students, small teams, team leaders, Scrum Masters, and professors reviewing project progress.*
- *They need a simple and quick way to manage notes, tasks, project communication, file sharing, chatting, events, calendar, and analytics.*
- *Functional requirements: users can create tasks, notes, events, view calendars, collaborate, and manage project roles.*
- *Non-functional requirements: the system should be quick and easy to use. It must be responsive across different devices and be reliable.*
- *The system must work across different web browsers and be accessible remotely.*

Our Solution

The solution to this problem is our Multi App Management System. We designed Multi to incorporate Notes, Tasks, Project Management, Chatting, Calendar with Events, plus dashboard Analytics, Organization offered by several pages like today, and a Task Board all in one.

We designed Multi to work as a full-stack web app that runs in mobile devices as well as PCs, opposite to only Android devices. We needed Multi to be simple enough for teams to use quickly.

We built Multi so Notes can be attached to a project, and tasks can appear in the Calendar together with the events. Multi has projects with invites and multiple teammates can work in several projects together. Projects have roles like editor and viewer so the viewers cannot edit the project. Team leaders can assign members a task and Multi does multiple other things.

Architecture

Users (Students and Small Teams)

↓

Browser (Laptop, Phone, Table)

↓

React + TypeScript Front End

↓

Next.js Web App (Tasks, Projects Dashboard)

↓

API Routes / Server Logic
Authentication - CRUD Operations - Upload Handling

↓

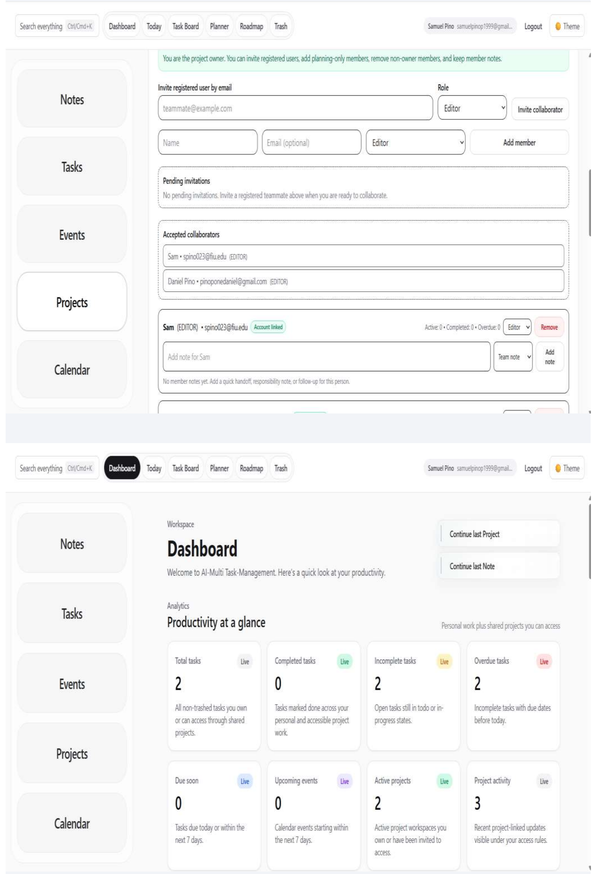
Data & Storage
PostgreSQL + Prisma – Vercel Blob

Deployment (GitHub -> Vercel -> Production Website)

The user interacts with a front-end built with React + TypeScript. The team used Next.js as the framework to connect the UI with the back-end logic, API routes, authentication, database access, and deployment. We chose Next.js because it was also developed by Vercel and it would work smoothly for deployment.

The team used pgAdmin 4 to interact with the PostgreSQL database easily without using the terminal. Prisma is needed in the tech stack in order to let the Next.js backend talk to PostgreSQL without always having to write the raw SQL. File uploads and downloads API routes used Vercel Blob storage.

Demo & Screenshots



Results & Evaluation

- Measured whether the main user flows worked from login to project collaboration.
- Tested CRUD features, calendar views, project chat, uploads, downloads, and deployment.
- Evaluated the app through local testing, production testing, GitHub PRs, Vercel logs, and team feedback.
- Headline result: the app was successfully deployed as a live web system accessible through a browser.
- Headline result: users can manage tasks, notes, events, projects, chat messages, and uploaded files in one workspace.
- No errors were found when the app was deployed. The users worked without reaching usage limits, so the app ran smoothly during testing.
- All environmental variables were reviewed, and the sensitive data is protected.

Engineering Process (SO6)

- SDLC phases we followed were planning, requirements, design, implementation, testing, deployment, and improvement.
- Key requirements were support for tasks, notes, events, calendar views, project collaboration, file sharing, and access control.
- Design decisions were to build a web-based system that works across Windows, Mac, Android, and iOS through a browser.
- For the design it was fundamental to keep all of the functionality in one app without device restrictions.
- Testing was done using the deployment link to reach the live demo of the app through Vercel.
- The entire team opened the app and tested the main functionalities such as project creation, note attachment to a project, writing mode, links in the calendar, and other features.
- Bugs were reviewed using local testing, GitHub pull requests, Vercel logs, Resend logs, and feedback from team members.

What's Next

The application cannot use Resend in order to send emails to its users. For example, if I create a project and register the email of a coworker, the app cannot automatically send an email. This would require purchasing a domain and maintaining it, which is costly. The app can register the email of its users but cannot send them an email. This is the same issue with domain expenses. The app was designed with the name of AI Multi Task Management System. For it to include AI is not possible at the time. It is desired, but not currently possible. The OpenAI API token usage cost is high and the key can only be obtained through payments. The biggest next improvement is one of the ambitious purposes it was created for: to become a fully working AI application that merges all of the existing features with AI. This would include AI-generated reports from a mix of tasks and notes and sending them to a specific email. The project also has usage limits. They are generous but real. Vercel Blob has usage limits that could be expanded.

Acknowledgments

Product owner: Samuel Pino, and: "KFSCIS Capstone Program, Prof. Masoud Sadjadi."