

This is a tight turnaround. Moving the deadline to Sunday Midnight removes your safety buffer. You now have approximately 3.5 days (including today).

Since the README analysis confirms the architecture is modular (Client/Server split), you can parallelize work effectively to meet this deadline.

Here is the Revised Sprint Schedule & Action Plan to ensure you complete the work and generate the necessary "proof of collaboration" by Sunday night.

 The 3.5-Day Emergency Sprint (Thurs - Sun Midnight)

Thursday (Today): Setup & Foundation (CRITICAL)

* Goal: By 11:59 PM tonight, the repo must be live, everyone invited, and environments running on all 5 laptops.

* Tasks:

* Lead (Wasike): Create GitHub Org &

Repo. Push the skeleton code (Client/Server/Mobile folders).

- * Backend Team (Elongu, Mukunzi): Run npm install and npx prisma init. Ensure MySQL connects. Push schema.prisma changes.
- * Frontend Team (Mwima, Adong): Run npm install. Ensure the React server starts. Commit the folder structure.
- * Proof: ~5-10 commits today (Initial Setup, .gitignore, Dependency adds). Friday: The "Heavy Lift" (Features)
- * Goal: Core logic implementation.
- * Backend Team:
 - * Implement User and Vote models in Prisma.
 - * Create the "OTP/Token" logic for the One-Person-One-Vote requirement.
- * Frontend Team:
 - * Build the Login Page (Input for Student ID/Token).

- * Build the Voting Dashboard (List of candidates).
 - * Collaboration: Create GitHub Issues for "API Endpoints" and "UI Wireframes". Comment on them to clarify data types.
- Saturday: Integration (The Danger Zone)
- * Goal: Connect Frontend to Backend. Do not wait until Sunday.
 - * Action:
 - * Frontend Team consumes the API. Fetch candidates from the database.
 - * Backend Team enables CORS and fixes connection errors.
 - * Auditability Feature: Backend team implements the logging system for "Immutable logs".
 - * Proof: Pull Requests (PRs) opened. Reviews must happen today. "LGTM" (Looks Good To Me) comments are vital for your grade.
- Sunday: Polish, Docs & Submission

- * Morning (8:00 AM – 12:00 PM):

- * Testing: Run npm test as specified in the README. Fix breaking bugs.

- * Afternoon (12:00 PM – 6:00 PM):

- * Documentation: Fill the /docs folder.

Upload the ERD and API Flow charts.

- * Readme Update: Ensure the README.md contributors list is accurate and setup steps work.

- * Evening (6:00 PM – 11:59 PM):

- * Final Merge: Merge all branches to main.

- * Mobile Bonus: If time permits, push the basic React Native screen to mobile/. If not, ensure the folder exists but is documented as "Future Work".

 Critical Adjustments for the Sunday Deadline

- * Drop "Nice-to-Haves":

- * If React Native (Mobile) is slowing you down, abandon it. The README lists it as "Optional/Bonus". Focus on the Web Client

and Server.

- * Simplify the "Real-time Reporting". A simple refresh button is faster to build than live WebSockets.
- * Strict Branching Strategy:
 - * With less time, merge conflicts will kill you.
 - * Frontend works strictly in client/folder.
 - * Backend works strictly in server/folder.
 - * Docs person works strictly in docs/ and README.md.
- * The "Proof" Checklist (For Grading):
 - * Ensure the Gantt Chart (GitHub Projects) shows items moving from "Todo" to "Done".
 - * Ensure Contributors Graph shows activity from all 5 members. If one person (e.g., Elongu) does backend, make sure Adong reviews his code so her name

appears in the history.