CODE 4 GovTech
Building Digital Public Goods

STRK Schedule - Blockchain-Enabled Diabetes Management for Government Healthcare Systems

Name	CHIRAG SINGHAL
Email ID	Chiragsinghal04@gmail.com
Phone Number	9818907290
GitHub ID	https://github.com/CS251000
Discord ID	https://discord.gg/4yNfuRwP Username- chiragsinghal4639
Current occupation	Student
Education Details	NETAJI SUBHAS UNIVERSITY OF TECHNOLOGY COMPUTER SCIENCE AND TECHNOLOGY (DATA SCIENCE)
Technical skills with level	• Python: Intermediate • JavaScript (ES6+): Expert • C++: Expert • React, Next.js, TailwindCSS: Intermediate

	• Firebase & MongoDB: Expert • AWS, PostgreSQL, MySQL: Intermediate

Title – Unifying on-chain medication system with Digital Pillbox and reminder system

Summary

Diabetic patients today juggle multiple apps and paper logs just to remember their medications—leading to missed doses, poor adherence, and fragmented care. STRK Schedule may enhance its secure, blockchain-backed tracking by adding a **Digital Pillbox** module that mirrors a real-life pill organizer, complete with on-chain medication slots and automated reminders. Patients load their medication schedule into the pillbox UI once; the PWA then sends push and in-app alerts at each dose time. All pillbox entries and reminder acknowledgements are recorded on Starknet—giving patients and providers a tamper-proof adherence record, while preserving privacy via zero-knowledge proofs. This integration also has the scope to integrate a reminder engine with STRK Schedule's decentralized, patient-owned data model.

Project Detail

Understanding of the Project

- **Existing STRK Schedule**: On-chain logging of meds, blood sugar, appointments; NFT minting for verified incidents; DAO governance.

- **New Enhancement**: A **Digital Pillbox** UI that:
  1. Displays a weekly/monthly grid of "pill slots" populated from the patient's on-chain medication schedule.
  2. Sends automated push/browser notifications and in-app reminders at each scheduled dose.
  3. Lets patients "check off" doses; these acknowledgements are hashed and stored on-chain for audit.

Issues & Support Needed

- **Browser Push Integration**: Guidance on reliable background notification triggers in a PWA context.
- **On-chain Acknowledgement Costs**: Mentor advice on batching reminder-acks into rollup proofs to minimize gas.
- **UI/UX for Low-Tech Users**: Best practices for an intuitive pillbox interface in a Web3 PWA.

Proposed Solutions

- Leverage the existing **Argent account abstraction** to sign batched “dose acknowledgment” transactions gas-abstracted via Starknet’s ZK rollups.
- Utilize the PWA’s service worker for **background sync** of scheduled notifications, with fallback to SMS/email via an off-chain relay for critical reminders.
- Base the pillbox UI on a simple calendar grid (inspired by MedRem) with color-coded on-chain “status” badges (taken/missed), updated after each acknowledgment.

Macro Implementation Details with Timelines:

Week	Focus Area & Tasks	Deliverables / Checkpoint
1	**Project Setup & Architecture**– Finalize system architecture for pillbox & reminder modules– Install & configure Starknet toolchain, Argent SDK, PWA boilerplate	• Project repo scaffolded with folder structure • Starknet testnet configured •PWA boilerplate with service-worker stub
2	**Smart Contract Design: Pillbox Schedule**– Define on-chain data model for daily/weekly pill slots– Write & unit-test core PillBoxSchedule contract	• PillBoxSchedule contract spec complete• Contract deployed to Testnet • Unit tests covering CRUD of pill slots
3	**Frontend: Pillbox UI Framework**– Build calendar-style grid component in PWA (weekly view)– Hook up grid to on-chain contract read calls	• Interactive pillbox grid showing empty slots • On-chain read integration (Testnet) retrieving schedule data
4	**Pillbox CRUD & Sync**– Add “Add/Edit” medication dialogs– Implement write calls to store schedule on-chain	• “Add/Edit Pill” modal implemented • Test transactions writing schedule entries • Basic form validation & feedback
5	**Reminder Engine: Notifications**– Wire up PWA service-worker for push/in-app notifications– Prototype off-chain SMS/email fallback relay	• PWA can send test notifications at arbitrary times • SMS/email relay stub running (off-chain)

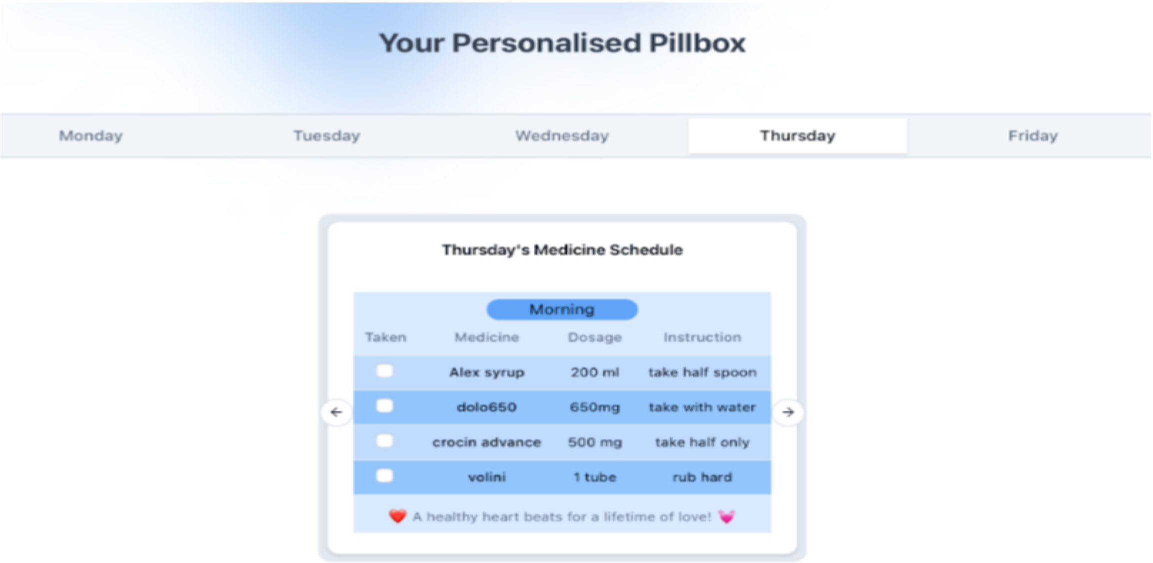
Week	Focus Area & Tasks	Deliverables / Checkpoint
6	**On-Chain Acknowledgement Prototype**– Design & deploy contract for recording confirmations– Frontend ack button & basic batching logic	• Mid-point Goal Reached: – Digital Pillbox UI fully CRUD-capable – Reminder engine firing real notifications – Prototype on-chain ack transactions via batching

Mid-Point Milestone (End of Week 6):

A complete pillbox experience on Starknet —patients can schedule meds, receive reminders, and “check off” doses with those acks recorded on-chain. From here on, we’ll optimize, secure, and prepare for Mainnet .

Week	Focus Area & Tasks	Deliverables / Checkpoint
7	**Batching & Gas Abstraction**– Integrate Argent for gasless writes– Build zk-proof bundler for batched acks **MAY ALSO NEED 4-5 DAYS OFF FOR FAMILY VACATION (however this won’t hinder the progress track as work will be adjusted accordingly)**	• Argent integration demo • Batching script producing zk-proofs ready for Starknet submission
8	**UI Enhancement & Dose History**– Add history view showing taken/missed badges– Color-coded status updates	• Dose history component implemented • Status badges reflecting on-chain state
9	**Performance & Security Optimization**– Audit contracts for gas & ZKP efficiency – Optimize service-worker for reliability	• Gas usage report & optimizations applied • Notification delivery success ≥ 95% in simulated low-connectivity test
10	**End-to-End Testing**– Write integration tests covering full flows (schedule → notify → ack)– Begin writing user/developer docs	• Automated test suite passing on CI • Draft of user guide and developer README
11	**Mainnet Preparation**– Final security audit (contracts & PWA)– Configure Mainnet deployment scripts	• Audit report with action items addressed • Deployment pipeline configured for Starknet Mainnet
12	**Production Deployment & Handoff**– Deploy contracts & PWA to Mainnet– Finalize docs, demo video, and handoff	• Live STRK Schedule App with Digital Pillbox & Reminders • Published docs, walkthrough video, and “Getting Started” guide

POC (Proof of Concept)- https://medrem.vercel.app



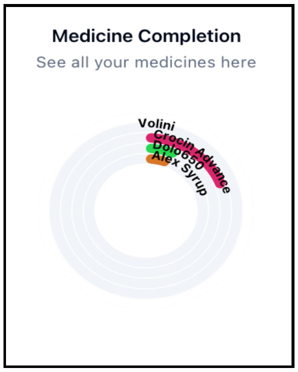
Your Personalised Pillbox

Thursday's Medicine Schedule

Morning

Taken	Medicine	Dosage	Instruction
□	Alex syrup	200 ml	take half spoon
□	dolo650	650mg	take with water
□	crocin advance	500 mg	take half only
□	volini	1 tube	rub hard

❤️ A healthy heart beats for a lifetime of love! 💗



Medicine Completion

See all your medicines here

Acceptance Criteria (Digital Pillbox & Reminders)

1. **Pillbox Functionality**
   - Patients can add/edit their daily medications into a calendar-style pillbox UI.
   - Each pill slot shows medication name, dosage, and scheduled time.
2. **Automated Reminders**
   - Push and in-app notifications fire at each scheduled time.
   - If the browser/PWA is closed, an SMS/email fallback is sent within 15 minutes.
3. **Acknowledgement & Auditing**
   - When a patient “checks off” a dose, an on-chain transaction (batched) is submitted.
   - The UI displays a history of taken vs. missed doses, with timestamps.
4. **Performance & Security**
   - Reminder delivery succeeds ≥ 95% of the time in live tests.
   - All ack transactions are ZK-proofed before Starknet submission, ensuring privacy.

Availability

Number of hours available to dedicate to this project per week	40 hours
Do you have any other engagements that will require your time? (projects/internships)	NO

Personal Information

About Me:

I am Chirag Singhal, a full stack software developer pursuing B.Tech in Computer Science and Engineering at NSUT, passionate about building impactful tech solving crucial daily life problems using latest tech stack. My journey so far has involved crafting real-world solutions like MedRem, a digital pillbox app built with Next.js and Firebase that helped users manage their medications through smart reminders. I've contributed to open-source platforms like in HacktoberFest and GSSOC, built scalable cloud workflows at myNachiketa Foundation using AWS and TextLocal APIs, and led development of dynamic user experiences across multiple full-stack projects. I have also participated in various hackathons and also secured 1st position in InnovateNSUT'24. My interest in decentralized tech and healthcare led me to contribute in **STRK Schedule**. I bring strong backend and system design skills, with proficiency in JavaScript, Firebase, PostgreSQL. Whether it's building a privacy-first App or scaling real-time systems, I'm passionate about solving complex problems that create meaningful social impact.

Apart from my academics I am also the president of DevComm-NSUT which is the largest technical society of NSUT catering to around 300 students. During my tenure I have worked upon various projects like NSUTX, AmulX and DSAX while conducting events like DSAweek, DevWeek and much more

Motivation to Apply

My technical background and real-world project experience align seamlessly with the goals of the STRK Schedule initiative. Through developing MedRem I've gained firsthand insight into the challenges of digital health adherence and the importance of intuitive, privacy-conscious design. My experience spans **React, Next.js, Firebase, and PostgreSQL**, and I've built full-stack solutions that scale—from trip planning platforms to medicine tracking workflows to warehouse stock managers. With growing expertise in **Databases, Express.JS, and Javascript**, I'm well-equipped to extend STRK Schedule with features like digital pillbox reminders and scheduling medical logs. I see this project as an opportunity to fuse my interests in healthcare and decentralized technology, creating a secure, transparent system that supports patients and providers alike. Contributing to this platform will allow me to apply my skills toward a high-impact, public-good solution at the intersection of health and Web3.

Previous experience/open source projects:

Project Name	Project Description	Links (if any)
Tech Intern at myNachiketa Foundation	Designed an automated workflow to collect user emails and phone numbers, then efficiently send promotional emails and messages using TextLocal and Mailchimp APIs. • Engineered a custom CDN using Next.js for the Google Sign-In popup .	
MedRem	A digital pillbox with medicine tracking and reminder notifications. Built using – Next.js, TailwindCss, Firebase, chart.js	Github repository https://medrem.vercel.app
Trippy	One stop trip management solution with real time chat feature using web sockets and Firebase. Database is handled using PostgreSQL and Drizzle ORM. Built using NextJS, TailwindCSS	Github repo https://trippy-pi.vercel.app/
CodeArena	Online coding platform with real-time analysis & leaderboards built using NextJS and Gemini AI.	Github Link https://codearena-five.vercel.app/