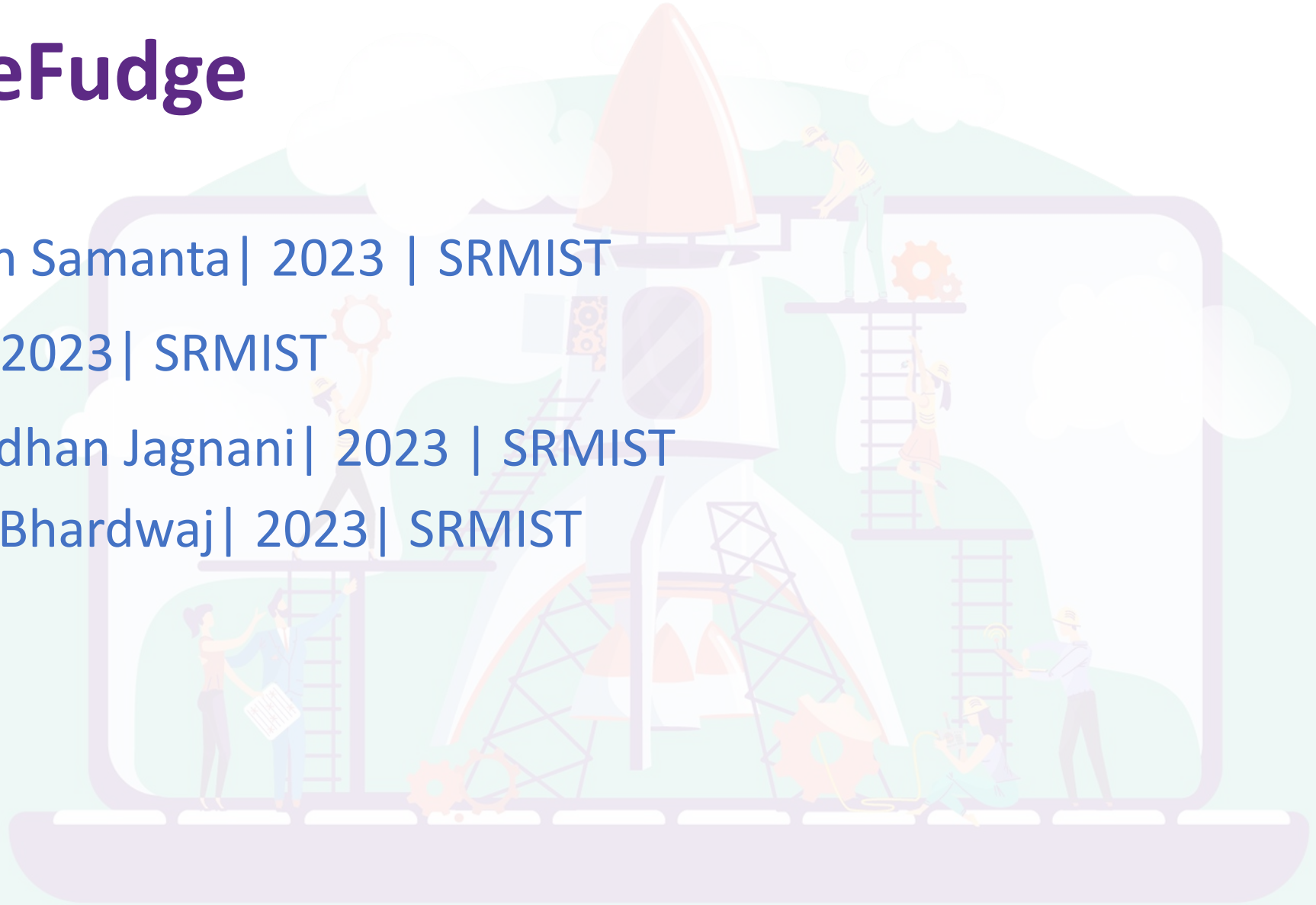
HackRx

BAJAJ FINSERV

NudgeFudge

- Gyanesh Samanta| 2023 | SRMIST
- Snehil| 2023| SRMIST
- Yashvardhan Jagnani| 2023 | SRMIST
- Eshaan Bhardwaj| 2023| SRMIST



HackRx

BAJAJ FINSERV

Tell us a bit about yourself

- Previous Projects:
  - *Contracts* – An Expense splitting app leveraging Blockchain (hedera, a side chain of ethereum) (Gyanesh and Yashvardhan) – Winner, Rookie Hacks II by MLH
  - *Pico* – URL Shortener with superpowers powered by DigitalOcean and Google Safe Browsing API (Snehil)
  - Smart Machine Authorizer – A flutter based mobile application that serves as an online portal for booking, monitoring, and remotely controlling machines in the college fabrication bay.

  (Check our GitHub for more :P)
- Hackathon Wins:
  - 2nd Runners up HackRx 2020 (Gyanesh)
  - Best First year team by ACMVIT(Snehil, Yashvardhan)
  - Category Prize, Ahoy Hacks by MLH (Eshaan and Gyanesh)
  - 1st Prize, HackThisFall by MLH (Yashvardhan)
  - 2022 HackChennai runners up by (Eshaan)
  - 1st overall & Best use of Twilio, HackEmpowered by MLH (Gyanesh)
  - Winner - SIIC Open Projects Hackathon (Eshaan)

HackRx

BAJAJ FINSERV

Hacking App Nudges

- The high level diagram is given in the next slide. Our proposed solution comprises 3 parts:
  (1) The web/mobile app where nudges are to be integrated.
  (2) React Admin panel.
  (3) Common Node Backend.

The admin panel provides the administrator with the flexibility to choose from a variety of sample nudges that can be integrated into the desired app. It takes in inputs like the *elementID* of fields where nudges are to be implemented, triggers, content and other configuration. This is all provided with an intuitive user interface which eliminates the need for the admin to code anything.

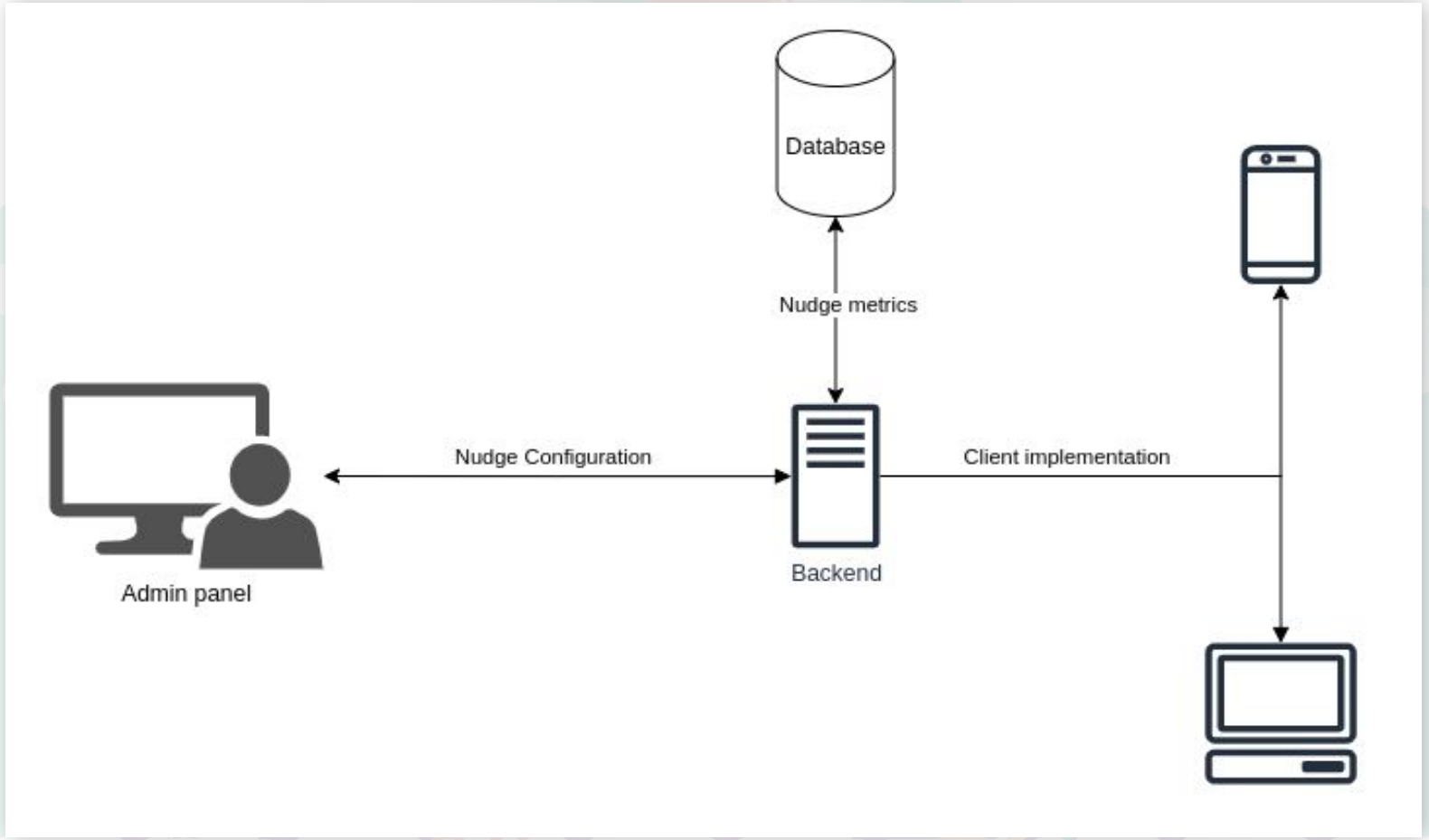
Once the nudges have been setup by the admin, the nudge data is sent to the Node server. The server analyzes the nudge information and generates a CDN script, taking into account parameters such as priority, triggers, timeout, customer threshold, and nudge sequence (nudge configuration).

Every interaction with the user is being recorded by the backend via fire-and-forget API calls for analysis of metrics. We plan to have a recommendation-system that can analyse the metrics for changes and suggest improvements.

On the app side, whenever the user progresses to a new screen, the server responds with the nudge configuration for the particular screen and the user is prompted with the appropriate nudges.

HackRx

Hacking App Nudges



Tech Stack

Tech Stack

- **Backend** - NodeJS, Firebase, Typescript, Python, Jupyter Notebook
- **Frontend (Admin Panel)** - ReactJS, Tailwind CSS, TypeScript
- **Frontend (Client Demo)** - React Native (Mobile App), ReactJS (Web App)
- **Cloud Service Providers** - (AWS or DigitalOcean)
- **Database** - MongoDB or Firestore
- **Other** - GitHub, Git, CI/CD

So, how is your solution different?

Our approach provides a no-code solution that is scalable and platform independent. Our implementation is not only limited to BFL apps, rather can be made open-source and implemented across any application.

We are taking inspiration from popular analytics solutions and creating a CDN script which can be easily injected into the document head for browsers. On the app (native) side, we're planning to making a react native library which will wrap the whole app and consume the nudge configuration from a single entrypoint. This solution will depend on a BFS implementation that we'll use to find the screen coordinates of targeted elements.

We have ideated fire-and-forget API calls for tracking the nudge metrics on every interaction with the user, like opening, closing, touch.

Future possible enhancements

Given the stringent duration of the hackathon, we aim to build our solution for React Native applications and Web Applications following along with the stack that Bajaj Finserv recommended (MERN). We would like to make this solution work with multiple platforms, namely, Flutter and Kotlin or Java developers.

Secondly, we aim to add more types of nudges than the three that we have proposed for now. We can also add multiple nudge triggers apart from the onLoad that we're implementing for the hackathon.

Our metrics system is based on the fire-and-forget API calls, which can bode for a loss of data, we can implement a queue based data storage on the local device that can help reduce this data loss.

We need to figure out a way to handle off-screen elements (Example: scrolling to the targeted view before showing the nudge) for extensive customizability.

Risks / Challenges / Dependencies

- Implementing a top level search feature for React Native which works in a similar way to the browser documents' `` `getElementById()` `` method.
- Figuring out a way to handle off-screen elements (Example: scrolling to the targeted view before showing the nudge).
- Assuming that people tasked with nudge creation have access to the IDs of UI elements.
- Figuring out an ID based flow that is intuitive for non-coders while still giving them enough control over choreographing complex nudge sequences.
- Generating different packages or CDN scripts/stylesheets based on the tech stack that the client app is based on.
- Complete configuration and implementation for web as well as mobile applications.

Anything Else ?

We are a team of students who have 4+ years experience in building out full fledged applications as hackers, freelancers and interns. With this project we aim to move forward Bajaj's offering in their health and finance sector whilst working as hackers. Combined, we have an experience of participating in 40+ hackathons, with 10+ wins and category prizes. We hope that we get this opportunity to build out our idea which can be used for the greater good whilst promising us a fun weekend, sipping tea and energy drinks while building in Pune!