

## EXPERIENCE

---

- **SparkHabits** Dallas, TX (Remote)  
Sept 2021 - Present
  - **Senior Frontend React JS, React Native Engineer**
  - **Dashboard made with React JS:**
    1. Made a customised dashboard where coaches of SparkHabits can create programs, upload resources like Video, PDF resources.
    2. Made Authenticated Routes with React router library with Firebase authentication and provided facility for Multi Authentication login with Email & OTP.
    3. Implemented multi file upload feature in the resources section, where coaches can upload multiple file types like PDFs, PNGs, Videos which will be uploaded to Firebase storage and mux.com with respective to their file types.
    4. Utilized Unsplash API to pick program images for the coaches and implemented a feature where they can record their own intro videos which will be uploaded to mux.com
    5. Utilized MaterialUI and Bootstrap for styling. Implemented Lazy loading of Images and pagination.
    6. Utilized proper optimization techniques to prevent unnecessary re-renders which reduces the performance of the dashboard and utilized hooks like useMemo and useCallback wherever necessary.
    7. Used Redux for global state management and react-table for managing users who have signed up to the program. Used formik for form handling.
    8. Utilized proper loading states so that when a program/resource is created/updated/deleted user will be properly notified what is exactly happening and properly made error messages so that it informs the user of what went wrong so that they can correct them.
  - **Landing page and Sign up Links with Next JS:**
    1. Made a customised Sign up page with Next JS where users can sign up to the program created by the coach with their details. Once they have signed up, they can complete their purchase with a stripe powered checkout links. Once successfully paid, they will be automatically added to the program with help of Webhook events subscribed to Stripe in the Express JS backend.
    2. Developed the landing page of SparkHabits, the program details screen where we show key details like resources uploaded to the program, courses which the user can learn as in a structured way and the sign up links which the user can utilise to join the program as well as the coach profile page.
    3. After the user has successfully registered, a SMS will be sent from Plivo to the end user's phone number as well as a welcome email.
    4. Ensured proper SEO techniques are followed as when the links are shared the open graph protocol will scrape the links which should properly retrieve the program image, the title and the description as well, which will provide a rich experience for the end user
  - **React Native:**
    1. Made the SparkHabits app from scratch by implementing plethora of features ranging from OTP authentication, Proper Handling of Navigation Stacks by separating them into Auth and Unauthenticated routes, Uploading Images, Videos and PDF's via Gallery, Camera, Permissions handling like Camera, Notification with help of react-native-permissions library, properly implemented Splash screen for both platforms with react-native-bootsplash, used react-native-track-player to build a Timer and Breath Meditation feature for the end users.
    2. Implementing caching for images with react-native-fast-image, Map feature to showcase charity locations. Utilized redux for global state management with offline caching of data with react-native-mmkv.
    3. Utilised Flipper to detect re-renders and highly optimized the app so that it can be performant in lower end Android and iOS Devices. Utilised Hermes engine to improve the performance of the android part of the app.
    4. Gave attention to detail to design standards of both Android and iOS side like the Status Bar, Bottom bar in android devices, the top notch in iOS, SafeAreaViews in iPhone 12 and higher end iOS devices.
    5. Implemented Notification feature with help of Firebase Cloud messaging a react native library called Notifee where users can reply to the notification just like how it happens in Whatsapp. Implemented foreground and background notifications and implemented deep linking feature where when user presses on the notification, they will be navigated to the proper screen just like how when we share a instagram post it navigates to that particular post irrespective of the screen the user is in. In the iOS side, implemented APNS notifications which are be powered by the cloud functions backend.

6. Used Sentry for Error Monitoring and Microsoft Code Push to implement over the air (OTA) updates when an error is reported. Implemented Fastlane for faster deployment of both the versions of the app to both testers in Testflight and Firebase App distribution and the app stores.
- **Node JS/Express JS with Google cloud:**
  1. Utilized Google Cloud functions to serve Express JS APIs to the three front end applications - The coach dashboard, the Sign up links and mobile application.
  2. Made multiple routes to retrieve data from Unsplash API, sending FCM notifications to the mobile app users etc.
  3. Made a pub sub function to send notifications to users at a timely interval.
  4. Rate Limited each and every routes to prevent DDOS with help of Redis cache and made sure proper security rules are implemented in the Firebase Console to prevent data leaks.
  5. Provided a proxy express route behind each third party application to prevent any abuse of API keys.
  6. Utilised Google Cloud logs to detect any errors and anomalies happening while the API is being called.
  7. Used proper Status Codes and Logging whenever an error happens.

## EDUCATION

---

- **NIT Trichy** Trichy, TN  
*Bachelors in Metallurgical Engineering.*

## PROJECTS

---

- **ApolloShot:** An app to take HD photos, videos and save them to gallery. I implemented wide range of features like compressing the images, videos, resizing the images, geotagging the images with location permissions, controlling exposure of the camera, in-app purchases with Revenue cat so that users can get the life time access to my app upon completion of payment. I made the application with help of the React Native Vision Camera library and the app is currently available in the Apple Appstore.
- **Links To my Projects:** APOLLOSHOT

## PROGRAMMING SKILLS

---

- **Languages:** Javascript
- **Technologies, Libraries, Frameworks:** React, Next JS, React Native, TypeScript Node JS, Express, Firebase, Bootstrap, Tailwind CSS, Material UI, Google Cloud Platform (GCP), GitHub Actions for CI/CD, Day.js, Moment.js, Razorpay, Stripe, i18n.
- **Tools:** Fastlane, Sentry, Appcenter, Supabase, Postman, Ngrok for webhooks.
- **GitHub:**<https://github.com/Gautham495>
- **LinkedIn:**<https://www.linkedin.com/in/gautham-vijayan/>