Telecom Mission Control Saas Platform - Design Document

Objective:

- Architect a Telecom Mission Control Saas Platform that has high-assurance as a11y-friendly, scalable FRONTEND system for web and mobile users used globally in both online and offline modes on devices of different form factors.
- Define and illustrate high-level architecture.
- Design the data flow models for each feature.
- Define the API model and design a performant API.
- Model the entities and their relationships.

Components Involved:

1. Telecom Mission Control Saas Platform:

Telecom Mission Control (TMC) Saas Platform is a centralized platform to operate, manage, monitor several aspects of telecommunication networks and services. It provides a centralized view and control of telecom network components and features. It offers several features like monitoring, reporting, user management, security management and several others. In simple words, it's a self service portal for Enterprise, Carriers and real-estate developers.

2. Enterprise:

Enterprises are businesses or organisations that subscribe to TMC platforms to enforce the boundaries to their employees/clients and auto discovery of new devices. They create user roles and map the actions that can be allowed and denied using policies. These policies are configured, monitored and managed by TMC platforms and enforced through equipment in cell towers.

3. Carriers:

Telecom providers like AT&T, Verizon, T-Mobile etc determine the devices supported at the towers and the device's operating system (iOS, Android, Windows, etc.). For example, some towers will only allow iOS-based AT&T devices, while others will allow Android and Windows-based AT&T and T-Mobile devices, as well as additional unique combinations. When devices try to connect to a carrier's network, the device OS along with carrier type is sent to the cell towers.

4. Cell Tower:

Physical infrastructure where the edge unit/equipment is installed and enforces the policies configured on the TMC platform by Enterprises. This equipment detects app usage and applies rules based on carrier & device type.

Understanding the Workflow:

Multiple carriers often share cell towers, but not all towers support all carriers. The equipment at the cell tower detects the apps used by the user on their device and captures the actions performed on the app by the user. The system captures the policy determined by the enterprise to enforce the rule, allowing or denying the list of actions permitted on the app based on the user's role. The system provides value-added services such as a real-time dashboard to monitor the security health. A higher subscription plan provides auto-remedial action enforced at the cell towers through enterprise-defined policies to secure the edge. The system integrates with various communications real estate developers to ingest (register) the information related to the towers present nationwide and the carriers they serve. The system allows connecting to different sources for onboarding existing users and their devices. Additionally, the system enables auto-discovery of new enterprise devices through the cell towers.

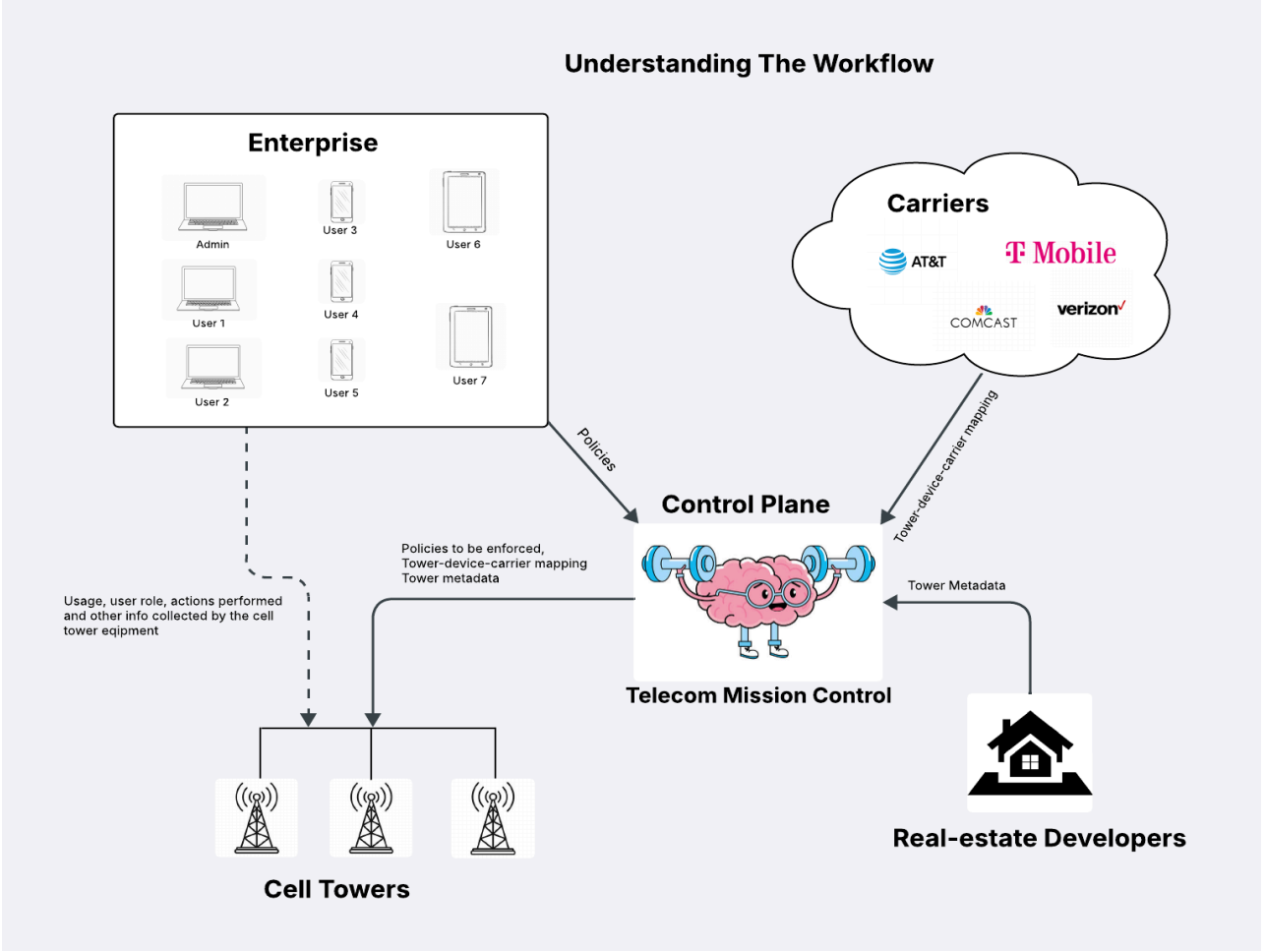


Fig 1. Understanding the workflow

High-Level Architecture with Dataflow:

1. **Communications Real Estate Developers**: When a new tower is physically constructed and the TMC edge device is installed, the unique ID given to the tower, location details, equipment supporting carriers and user device OS that the device can accommodate is all ingested through the TMC frontend application after real estate developer login is successful. This calls the backend endpoint to store the

tower metadata in the DB or update if it's already stored. Tower metadata is stored in the Tower table of the DB.

2. **Enterprise**: Enterprise administrator configures the user profile in the Enterprise module of the TMC Application with all the user details and assigning a user role along with permissible and deniable actions called policies. User details are stored in the User table of the DB and policies in the Policies table of the DB. When the enterprise user logs into the device, the cell tower notifies the application when the deniable actions are requested. With higher subscriptions, they can even block those actions on the device.

3. **Carriers**: Carriers are backbone for this workflow but do not send data to TMC directly. When an enterprise device tries to connect through a carrier, they provide the carrier and device metadata to the cell tower equipment through TMC platform which then enforces the policies.

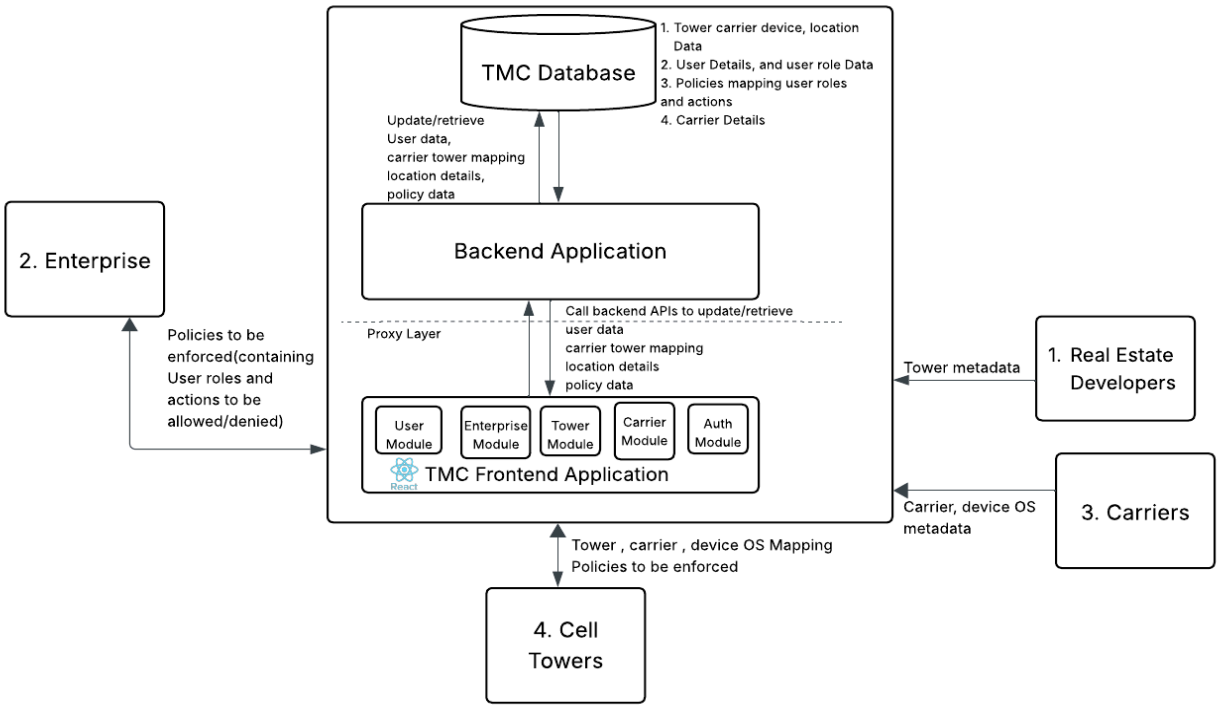


Fig 2. High Level Architecture

4. **Cell Towers:** Cell Tower extracts the user details, device type, carrier details from TMC platform. Based on the user role, action requested is checked against the enterprise policies. With higher subscription, deniable actions are blocked otherwise the application is notified of the action requested. Along with this, the user details are used for monitoring and logging the actions performed. Traffic is monitored and real-time events are streamed to the TMC platform application to update the backend and also the dashboard. As a value added service, the tower equipment health data is also streamed and monitored by the application in real-time.
When an unknown device/user is connected in the enterprise network, equipment streams this data to the TMC application and a notification is sent to the appropriate enterprise administrator who will then issue registration details of the user using that device through the TMC.

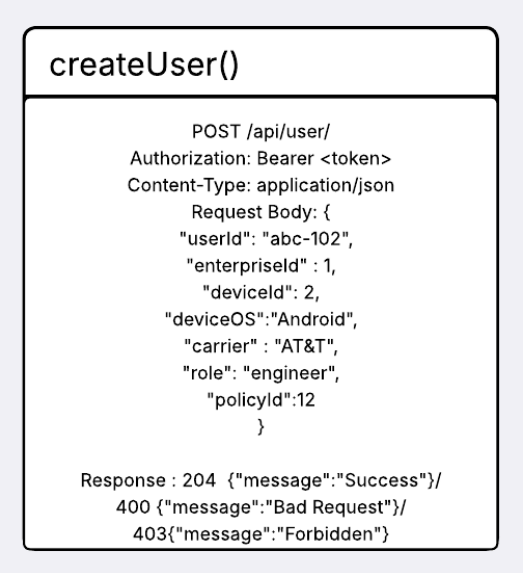
Telecom Mission Control Application: This platform is the central brain of this operational flow since it manages and controls all the aspects like enterprise, carriers (indirectly), cell tower equipment and real-estate developers. All the data relating to the operations are stored in the TMC DB. TMC DB

contains tables like User, Tower, Enterprise, Policies and so on to store corresponding data. Backend application APIs are used to do all the crud operations on the DB tables. These APIs are called from the frontend application through the proxy layer to prevent any attacks on the server. Different roles are authorised and authenticated to provide different options and views for different roles. The data from the equipment in the towers are all streamed in real-time and are all asynchronously stored/modified/updated in the DB tables. The changes to the policies are again asynchronously sent to the equipment in real-time. Tower Health is monitored as a value added service. Monitoring and logging modules are designed and implemented for the admin roles to monitor traffic, violated and blocked actions, cost trend and other details as requested.

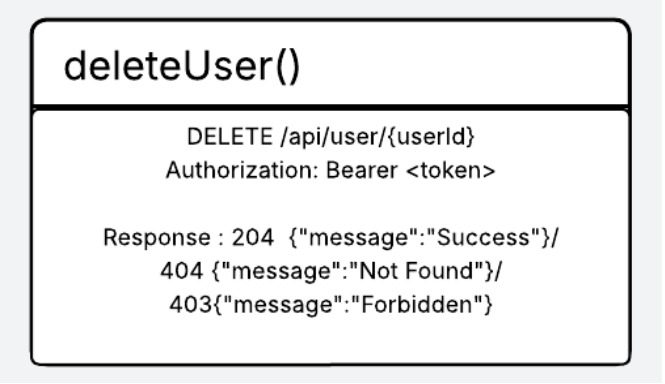
Features of the Telecom Mission Control Web Application for Enterprise, Tower device and Real-estate Developers:

1. Enterprise:

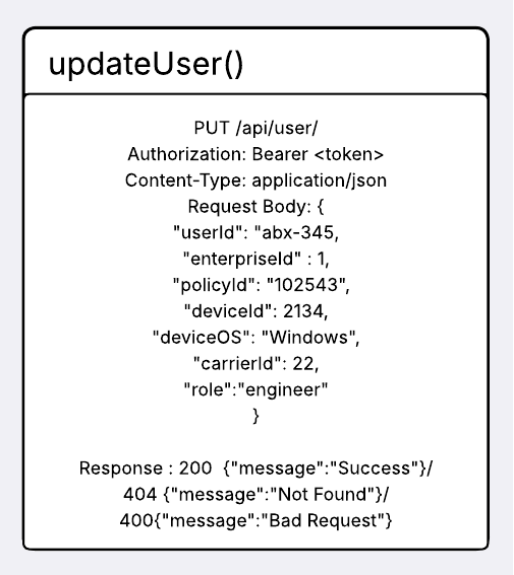
- Admin/User Authentication : Authenticates enterprise admins and users to use the application
- Authorization: Provides particular webpages based on the user roles after authentication
- User creation: Allows enterprise admins to create user profiles entering all user details and assigning a role.



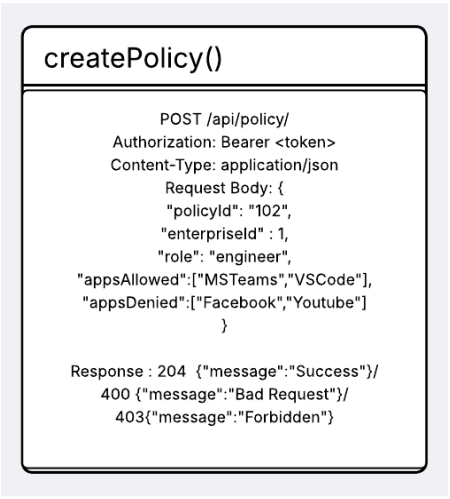
- Delete user: Delete users from Users tables once the user leaves the enterprise.



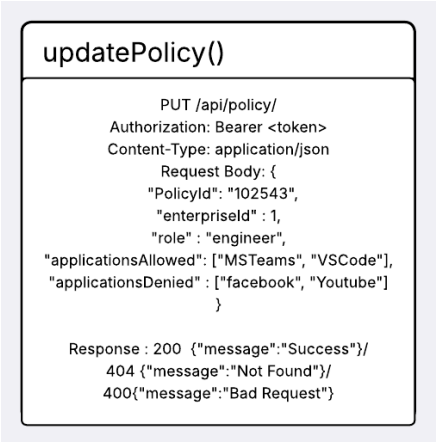
- Update user: Modify user details in the DB



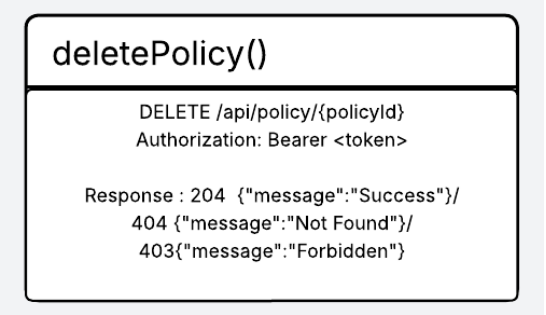
- Policy creation : Maps user roles, actions allowed, policy unique ID to update policy table



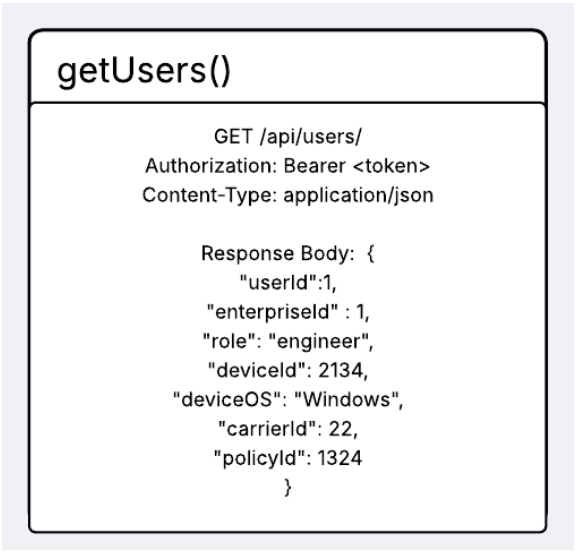
- Update policy: Updating user ID, policy IDs or actions allowed in the DB



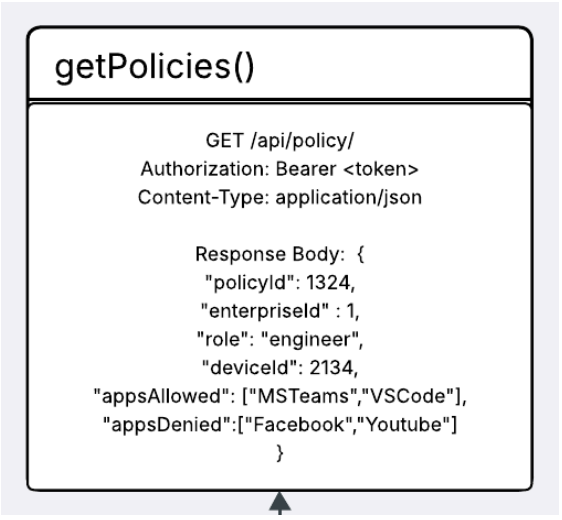
- Delete policy: This is called either by delete user endpoint or by enterprise admins to delete policies for a particular User.



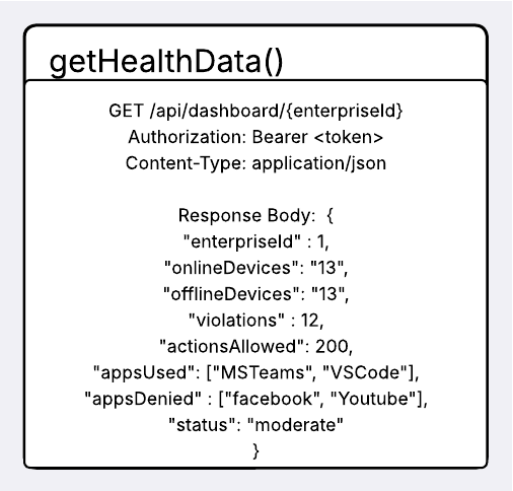
- Get all users: This feature enables the admin profile holder to see all the existing users and their details.



- **Get all policies**: This feature enables the admin profile holder to see all the created policies and their details.



- **Get security health data** : This endpoint returns information to update the dashboard like online and offline devices , allowed and violated actions and so on.



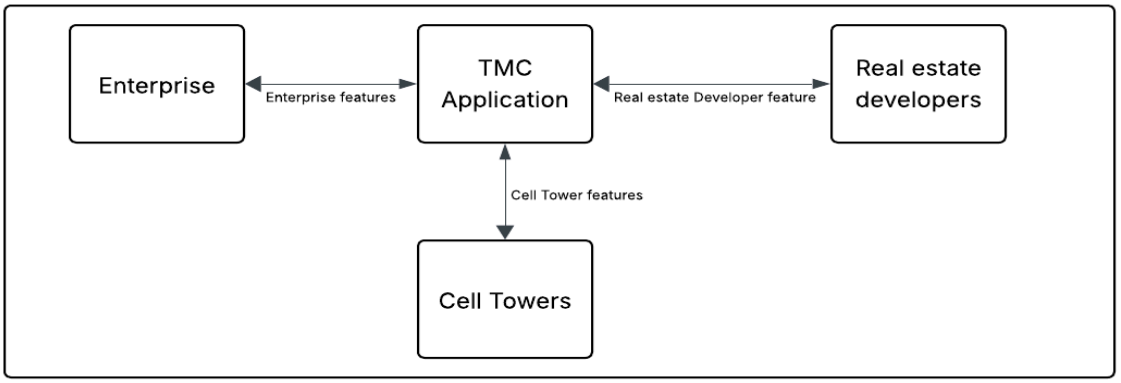
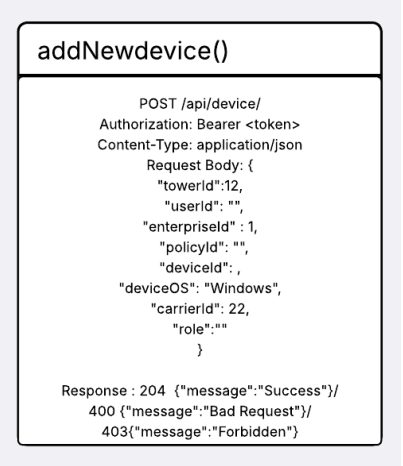


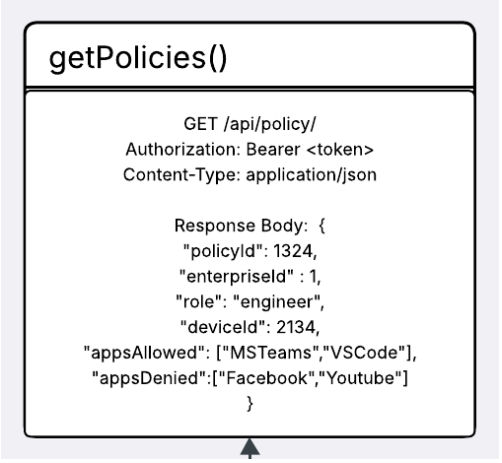
Fig 3. Features with Dataflow

2. Equipment on Cell Tower:

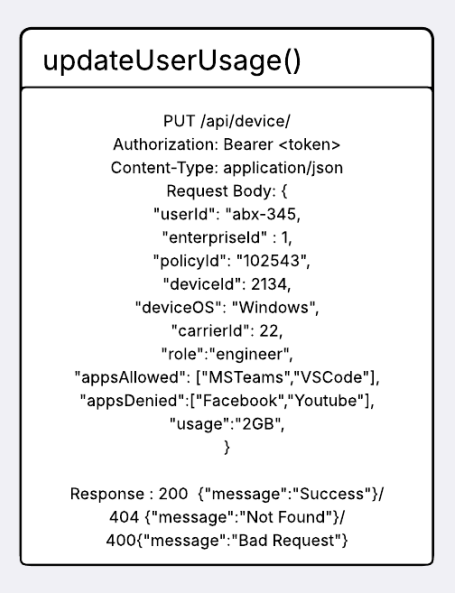
- Auto-discovery: The equipment scans for new unregistered devices and notifies the TMC application with details of the device. This notifies the enterprise admin to register the device and provide the user a role. The equipment later continues to record the app usage, actions requested and other details like any other registered device.



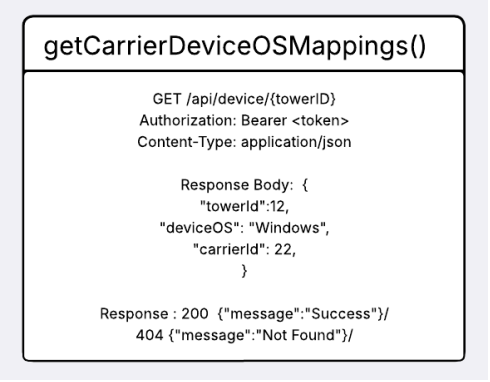
- Get policies to tower equipment: This feature allows the system to update the policy/policy changes in the equipment from the DB.



- Send usage data: Network and application usage, amount of data sent and received are all streamed to TMC application for reporting and will be part of dashboard metrics if chosen to be displayed.

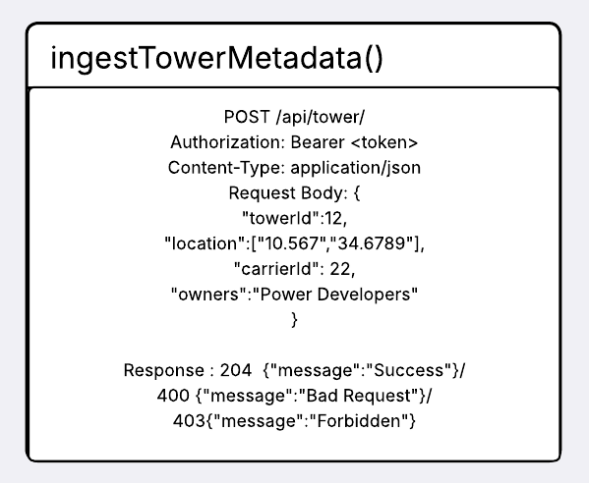


- get carrier device OS mapping: Carrier and device OS mappings are in the TMC DB and are retrieved during the setup of the device in the tower.



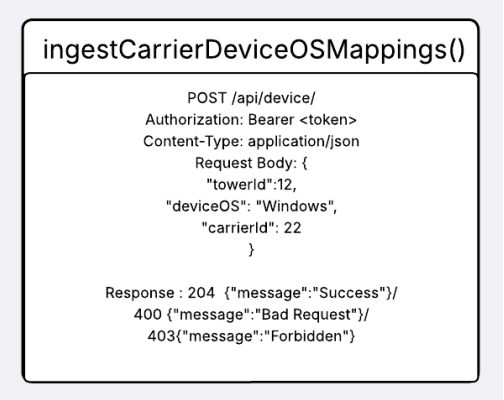
3. Real-estate developers:

- Ingest tower metadata: Tower details like the location, carriers supported, device OS supported, owners and other tower related data will be ingested to the DB using this feature.



4. Carriers:

- Ingest carrier device mapping: Carriers provide details about the towers that can support specific carriers and device OS. This mapping is stored in the DB in Carriers table.



Technical Aspects for Telecom Mission Control Frontend Application:

This can be a responsive React application that works on web browsers, mobile devices of all forms and supports majorly used operating systems that the enterprise workers operate on.

- This can be a modularised application that provides space for enhancements on the application and eases maintainability.
- The major advantages of using React applications are the performance with virtual DOM architecture and a redux store house that stores various states with respect to different components like enterprise, Carriers, real-estate developers and policy related data.
- Typescript with React library increases maintainability and reduces the number of errors and helps with debugging.
- React boilerplate comes with Jest and enzyme for unit testing reducing the initial effort to install and bundle all the libraries together.
- End-to-End Automated Testing can be done using Cypress or Playwright.

To achieve global accessibility and high availability:

- Register the TMC domain name (such as tmc.com) on a domain registrar like godaddy.com or namecheap.com etc.
- Store static content in Cloud storage buckets like AWS S3 or Azure Storage.
- Configure CDNs for faster loading of static content on user side.
- Deploy global load balancers on the SaaS cloud provider and configure SSL based communication.
- Plan to deploy the TMC backend application across multiple regions and across multiple availability zones in each region for high redundancy and availability.
- TMC backend application can be run on Kubernetes or auto-scaling VM groups to handle load variability.

To make the application work in offline mode, following strategies can be used based on the requirement specified:

- We can use caching strategies or use local storage for static content or API responses.
- We can use CDNs for larger content, but need to be cached on the first load when the device is online.
- If the requests made need to be queued in offline mode and then requested after the connectivity resumes, we can also use a service worker based on the requirements.

Entity-Relationship (ER) Diagrams: