**Estate Management App Skeletal View**

1. Admin Dashboard:
2. User management (add/delete users, users report)
3. Settings:
   1. Add/delete estate street
   2. Define number of access gates
   3. Add/delete category
   4. Add/delete second principal users **(part of users)……..**prinicipal users both first and second and put them in their category
   5. Delete dependent, but cannot add dependent
   6. Add/Remove Security users and put them in their category **(part of users)……..**
4. Reporting(access report)\*\*\*\*
5. Create/reset permanent access code for dependents (state-location-estate street-category-users-generate access code)
6. Security Dashboard:
   1. Validates code
   2. Validates report
7. Principal User
   1. Add/delete second Principal user ( where required)
   2. Add/delete dependent and put them in their category
   3. Reset personal Access code and reset that of dependants
8. Super admin dashboard:
   1. Admin dashboard
   2. Reporting – Access users report, security report, code status report
   3. Settings – create admins
   4. Register estate name / address/estate official number and estate geographic address
   5. Manage estate start date, estate deactivation due to payment

**Timeline (Total - 21 Days)**

**\*Day 1-2 (9/05 – 10/05):** Project Setup & Planning

-Define API structure, endpoints, and database schema

**\*Day 3-5 (12/05 – 15/05):** Backend Development - Core Functionalities

-Implement Authentication & Role-based Access Control

-Develop Admin functionalities (User management, Gate & Street Management, Reports)

-Set up Principal user & Dependant management

-Implement Visitor code generation & expiration logic

-Implement Security user verification system

-API Testing & Debugging

**\*Day 6-9(16/05 – 19/05):UI Development - Admin & Security Dashboard**

-Develop Admin Panel (User & Estate Management, Reports)

-Develop Security User Interface (Visitor code validation, Logs)

-Develop Principal User Dashboard (Dependant & Visitor Code Management)

-Develop Dependant User Interface

**\*Day 10-12 (19/05 – 21/05):API Integration & Frontend Testing**

-Connect UI with API

-Conduct basic functionality tests

- Full System Testing & Debugging

-Test all user flows

-Fix critical bugs & optimize

**\*Day 13(22/05): Deployment to Staging & User Training**

-Deploy for internal testing

**\*Day 14 (23/05): Final Adjustments & Production Deployment**

-Final security review & optimizations

-Deploy to live production

**\* Day 15-20 (24/05 – 30/05): Replicating the web UI on the Mobile app and connect it to API**

**\* Day 21(31/05): Testing of the Mobile apk**