

## Drone Log Management System - Low Level Design

### API Endpoints

#### Drone APIs:

- POST /api/drones/add: Create drone
- PUT /api/drones/edit/:id: Update drone
- GET /api/drones/getallDrones: Get all drones
- DELETE /api/drones/delete/:id: Delete drone

#### Log APIs:

- POST /api/logs/upload: Upload .tlog file
- GET /api/logs: Get all logs
- GET /api/logs/:id: Get log by ID
- DELETE /api/logs/:id: Delete log

#### Incident APIs:

- GET /api/incidents: Get all incidents
- GET /api/incidents/:id: Get incident by ID
- POST /api/incidents: Create incident
- PUT /api/incidents/:id: Update incident
- DELETE /api/incidents/:id: Delete incident

#### PrePostCheck APIs:

- GET /api/prepostchecks: Get all checks
- GET /api/prepostchecks/:id: Get check by ID
- POST /api/prepostchecks: Create check
- PUT /api/prepostchecks/:id: Update check
- DELETE /api/prepostchecks/:id: Delete check

### Database Schema

Drone: { name, modelName, serial, createdAt }

Log: { drone, filename, path, size, startTime, endTime, durationSeconds, gpsTrack, messagesCount, p

Incident: { drone, log, title, description, severity, createdBy }

PrePostCheck: { drone, log, type, items, performedBy }

### Frontend Components

- Aircraft: Drone CRUD with table and forms
- Logs: Log upload, list, details, delete
- Incidents: Incident CRUD with table and forms
- PrePostCheck: Check CRUD with table and forms