

SEAAL File Manager

Installation & Setup Guide

Developed by Rabhi Slimane

August 11, 2025

Overview

The SEAAL File Manager is a complete file management system consisting of:

- **Node.js Backend (Express)** — Handles file creation, loading, unloading, and status retrieval.
- **Angular Web Console** — Displays real-time file statuses with automatic refresh.
- **Flutter Mobile App** — Allows loading/unloading files directly from a smart-phone.

Project Structure

```
SEAAL_File_Manager_Project/  
  backend/           # Node.js server  
  web_console/       # Angular web console  
  mobile_app/        # Flutter mobile app
```

Requirements

Before starting, ensure you have installed:

- Node.js (v14+) — Download [here](#).
- npm (comes with Node.js).
- Flutter SDK — Installation [guide](#).
- Angular CLI — Install globally:

```
npm install -g @angular/cli
```

Backend Setup (Node.js)

1. Open a terminal and navigate to:

```
cd backend
```

2. Install dependencies:

```
npm install express cors
```

3. Start the server:

```
node server.js
```

4. The server runs at: `http://localhost:3000`.

Testing Endpoints:

```
# Status list
curl http://localhost:3000/status

# Load file #1
curl -X POST http://localhost:3000/load/1

# Unload file #1
curl -X POST http://localhost:3000/unload/1
```

For Real Phone Testing:

- Find your PC IPv4:

```
ipconfig
```

- Replace `localhost` with your IP in Flutter and Angular configs.
- Allow port 3000 in firewall:

```
New-NetFirewallRule -DisplayName "Allow Node 3000" -Direction
  Inbound -LocalPort 3000 -Protocol TCP -Action Allow
```

Web Console Setup (Angular)

1. Navigate to:

```
cd web_console
```

2. Install dependencies:

```
npm install
```

3. Update `apiUrl` in `src/app/app.ts`:

```
apiUrl = 'http://localhost:3000';  
// OR http://<PC_IP>:3000 for real phone
```

4. Start Angular dev server:

```
ng serve
```

5. Open: <http://localhost:4200>.

Mobile App Setup (Flutter)

1. Navigate to:

```
cd mobile_app
```

2. Install Flutter packages:

```
flutter pub get
```

3. Update BASE_URL in `lib/services/api_service.dart`:

- For Android emulator:

```
const String BASE_URL = "http://10.0.2.2:3000";
```

- For real phone:

```
const String BASE_URL = "http://192.168.1.15:3000";
```

4. Run:

```
flutter run
```

Testing Workflow

1. Start backend:

```
cd backend  
node server.js
```

2. Start web console:

```
cd web_console  
ng serve
```

3. Run mobile app:

```
cd mobile_app  
flutter run
```

4. On mobile app:

- Enter a file number (1–10).
- Tap **Load File** or **Unload File**.

5. Verify web console:

- Status updates every 6 seconds.

6. Check backend folder:

```
4-default.txt -> 4-charge.txt -> 4-decharge.txt
```

Notes

- Default files (1-default.txt ... 10-default.txt) are auto-created.
- Statuses:
 - **default** — Initial state.
 - **charge** — File loaded.
 - **decharge** — File unloaded.
- Angular console auto-refreshes every 6 seconds.
- Mobile app uses Provider for state management.