**🏢 Real Life Example Compare**

Think pannunga neenga oru **College Hostel In-Charge** nu.

* Hostel la **students list** irukum → (FastAPI la /api/v1/admin/users)
* Oru student rule break panna → **suspend panna vendiyathu** → (API /suspend)
* Avan behavior correct aagiduchuna → **restore panna vendiyathu** → (API /restore)
* In-Charge edutha action ellam **register notebook la log** pannuvanga → (Database la admin\_actions)

**🔹 How it works (Step by Step – Thunglish)**

**1. Purpose**

**Admin → In-Charge**

* Hostel in-charge oda duty: student accounts (users) manage pannradhu.
* Profile view, suspend, restore → app la actions.

**2. Scope**

* Student list show panna vendiyathu.
* Suspend / restore option kuduka vendiyathu.
* Every action record pannanum (log book madhiri).

**3. Functional Requirements**

* All students details view panna.
* Suspend → student hostel la stay panna koodadhu (status = suspended).
* Restore → suspended student ku hostel la permit kuduthu (status = active).
* Log all actions (like warden notebook la entry).

**4. Data Models**

* **users table** → student details notebook.
* **admin\_actions table** → warden actions notebook (suspend/restore record).

**5. API List**

* GET /api/v1/admin/users → all students list check pannradhu.
* POST /api/v1/admin/users/{id}/suspend → specific student suspend panna.
* POST /api/v1/admin/users/{id}/restore → suspended student restore panna.

**6. Sequence Flow**

* Warden (Admin) → student list check pannuvanga.
* Suspend/Restore option choose pannuvanga.
* System (college record) update agum.
* Log book la entry add agum.

**7. Testing Plan**

* Unit test → suspend/restore logic correct ah work aagutha check pannum.
* Integration → view + suspend + restore full flow work aagutha check pannum.
* Edge case → already suspended student ku suspend again panna try pannum → correct error varanuma nu check.

**8. Deliverables**

* Endpoints (warden action buttons).
* DB migrations (student notebook design).
* Swagger docs (warden ku instruction manual).
* Tests (check list).

**9. Timeline**

* 1st week → notebook schema design.
* 2nd-3rd week → student list endpoint.
* 4th week → suspend/restore API.
* 5th week → log integration.
* 6th week → security check.
* 7th week → tests.
* 8th week → final demo.

👉 In simple words:  
**Admin Dashboard (FastAPI) = Hostel In-Charge system.**

* **Users = Students**
* **Suspend = Hostel suspension**
* **Restore = Re-entry**
* **Admin Actions = Warden notebook**

?