

## Sprint 2: Materials CRUD (Backend + Basic Frontend)

### Goal:

Enable users to create, read, update, and delete inventory materials.

### User Story:

"As a user, I want to manage materials so I can keep track of inventory items."

### Tasks:

#### 1. Create Material Model [x] Completed

- Fields: name, type, quantity, location, notes

#### 2. Create Materials Routes [x] Completed

- POST /api/materials - Add material
- GET /api/materials - Get all materials
- GET /api/materials/:id - Get single material
- PUT /api/materials/:id - Update material
- DELETE /api/materials/:id - Delete material

#### 3. Create Controller Functions [x] Completed

- Handle logic for all routes (CRUD)
- Use protect middleware to restrict access

#### 4. Connect Routes to server.js [x] Completed

- Mount routes under /api/materials

#### 5. Test Routes in Thunder Client [x] Completed

- Confirm full CRUD operations work
- Ensure token is required for protected actions

## 6. Setup Basic Frontend View (Optional)

- Use fetch or Axios to display materials list
- Connect with backend for read operation

### Dependencies Needed:

- (already installed) express, mongoose, dotenv, jsonwebtoken

### Deliverables:

- Fully functional Material model [x]
- RESTful API with CRUD operations [x]
- Protected routes using auth middleware [x]
- (Optional) Frontend display for material list

### Next Sprint:

- Locations CRUD
- User Story: "As a user, I want to manage and assign locations to materials."