Knowledge Transfer Document

Project Title	**:** Comprehensive Database Management project
Version	**:** v1.0
Document Type	**:** Knowledge Transfer (KT) Document
Date of Preparation:	04-05-2025
Client / Stakeholder:	CASFOS Coimbatore

Team members:

Methilesh K/22i228

Vaishnav Ram K/21i269

Nirmal Kumar M R/21i339

Abilash A/22i462

Akash Aravind S/22i202

Praveenkumar A V/22i241

Contents

1. Project Overview

Purpose / Objective:
The project is designed to manage institutional assets (permanent and consumable) and faculty records. It provides functionalities for asset tracking, issuing/returning assets, maintenance, disposal, and building upgrades, as well as faculty data management, including verification, approval, and notification workflows. The system aims to streamline administrative processes, ensure accountability, and provide reporting capabilities.

Tech Stack:

- **Backend**: Node.js, Express.js
- **Database**: MongoDB
- **Frontend**: React.js
- **File Uploads**: Multer (for images and PDFs)
- **Middleware**: Custom middleware for authentication/authorization (assumed), Multer for file uploads
- **Others**: dotenv for environment variables, Mongoose for MongoDB ORM

Audience:

- New developers joining the project
- Support and maintenance team
- QA team for testing APIs and frontend
- Administrators managing assets and faculty data

---

2. Architecture Diagram & Description

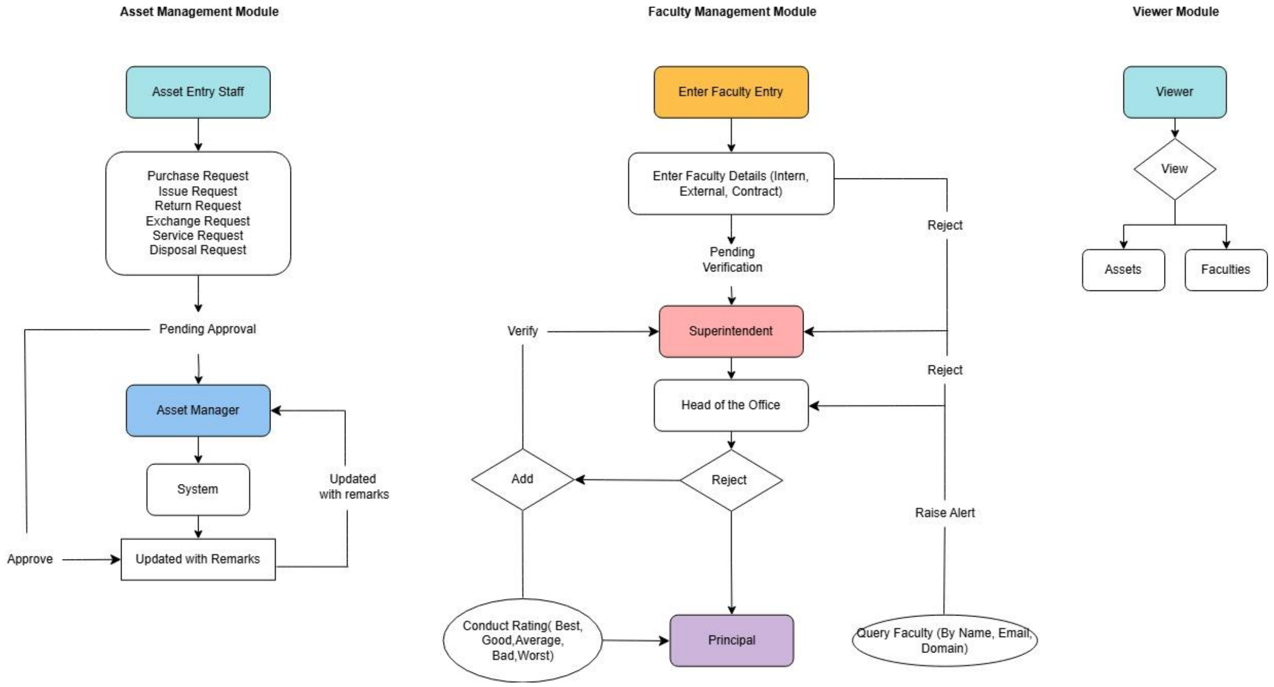
High-Level System Architecture:
The system follows a client-server architecture with a React frontend communicating with a Node.js/Express backend via RESTful APIs. The backend interacts with a MongoDB database to store asset and faculty data. File uploads (e.g., photographs, invoices, PDFs) are handled using Multer and stored on the server or a cloud storage service (to be specified).

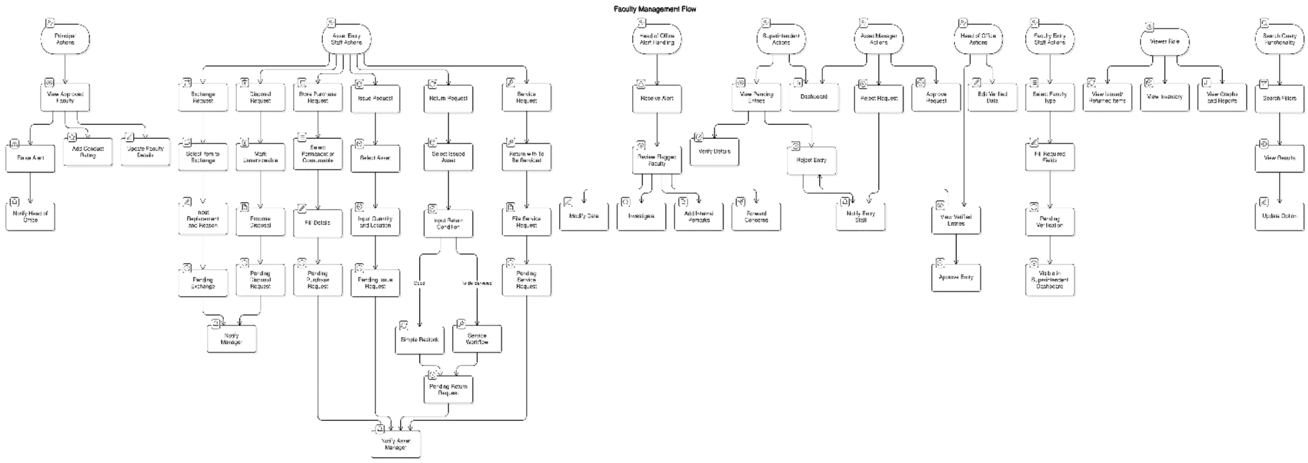
Component Diagram:

- **Frontend (React)**: User interface for asset management, faculty management, and dashboards.
- **Backend (Node.js/Express)**: Handles API requests, business logic, and database operations.
- **Database (MongoDB)**: Stores collections for assets (permanent, consumable, rejected), faculty (unverified, confirmed, rejected, notified), and temporary data (auto-saved, temp issues).
- **File Storage**: Local server or cloud storage for uploaded files (e.g., photographs, invoices, PDFs).

API Flow:

1. User interacts with the React frontend (e.g., submits faculty data or requests asset disposal).
2. Frontend sends HTTP requests to Express backend APIs (e.g., POST /api/faculty/save, POST /api/assets/requestForDisposal).
3. Backend validates requests, processes data (e.g., saves to MongoDB, handles file uploads), and returns responses.
4. Frontend updates the UI based on API responses.





3. Environment Setup

Pre-requisites:

- **Node.js**: v16.x or higher
- **npm**: v8.x or higher (or yarn)
- **MongoDB**: v5.x or higher (local or MongoDB Atlas)
- **GUI Tools**: MongoDB Compass (for database management), Postman (for API testing)
- **Code Editor**: VS Code or similar

Frontend Setup (React):

```
cd client

npm install

npm start
```

- Default port: 3050
- Ensure the backend API URL is configured in the frontend (e.g., in src/services/api.js).

Backend Setup (Node.js):

```
cd server

npm install

npm run dev
```

- Default port: 5000 (or as specified in .env)
- Uses nodemon for development (assumed).

Environment Variables (.env Example):

```
# Backend

PORT=5000

MONGO_URI=mongodb://localhost:27017/asset_faculty_db

JWT_SECRET=your_jwt_secret_key

UPLOAD_PATH=./uploads


# Frontend (if needed)

REACT_APP_API_URL=http://localhost:5000/api
```

Database Setup:

- Install MongoDB locally or use MongoDB Atlas.
- Create a database named asset_faculty_db (or as specified in MONGO_URI).

- Ensure Mongoose models (server/models/) are synced with the database schema.

---

4. Folder Structure

Frontend (client):

```
client/
├── public/
│   └── (public assets, e.g., index.html, favicon.ico, not listed in detail)
├── src/
│   ├── assets/
│   │   └── (static assets like images, not listed in detail)
│   ├── components/
│   │   ├── addnew.jsx
│   │   ├── adminassetapproval.jsx
│   │   ├── adminassetdashboard.jsx
│   │   ├── assetissue.jsx
│   │   ├── assetmanagerdashboard.jsx
│   │   ├── AssetReturn.jsx
│   │   ├── AssetStore.jsx
│   │   ├── AssetUpdation.jsx
│   │   ├── facultyentry.jsx
│   │   ├── facultyentrydashboard.jsx
│   │   ├── facultyverifierdashboard.jsx
│   │   ├── facultyverifierview.jsx
│   │   ├── facultyview.jsx
│   │   ├── headnotificationdashboard.jsx
│ │ ├── hooassetapproval.jsx
│ │ ├── hooassetview.jsx
│ │ ├── hooreturnapproval.jsx
│ │ ├── hooview.jsx
│ │ ├── login.jsx
│ │ ├── managepage.jsx
│ │ ├── manageassertview.jsx
│ │ ├── PrincipalAssetUpdation.jsx
│ │ ├── principaldashboard.jsx
│ │ ├── principalnotification.jsx
│ │ ├── principalfacultyview.jsx
│ │ ├── register.jsx
│ │ ├── storekeeperassetupdation.jsx
│ │ ├── storekeeperdashboard.jsx
│ │ ├── updatefacultyentry.jsx
│ │ ├── updatefacultyentryprincipal.jsx
│ │ ├── usermanagement.jsx
│ │ ├── viewassert.jsx
│ │ ├── ViewerAssetview.jsx
│ │ ├── viewerDashboard.jsx
│ │ └── ViewerFacultyView.jsx
│ ├── fonts/
│ │ └── (font files, not listed in detail)
│ ├── styles/
│ │ └── (CSS or SCSS files, not listed in detail)
│   ├── App.jsx

│   ├── main.jsx

│   ├── .eslintrc.cjs

│   ├── index.html

│   ├── package.json

│   ├── README.md

│   ├── tailwind.config.js

│   └── vite.config.js

├── node_modules/

│   └── (dependencies, not listed in detail)

├── .gitignore
```

Backend (server):

```
server/
├── config/
│   └── (configuration files, e.g., multerconfig.js,etc…)
├── controllers/
│   ├── assetController.js
│   └── facultyController.js
├── middleware/
│   ├── AssetNotification.js
│   ├── autoValidation.js
│   ├── AutoSaveFaculty.js
│   ├── BuildingUpgrades.js
│   ├── ConfirmedFaculty.js
│   ├── DataEntryModel.js
│   ├── DeadStock.js
│   ├── DeposedAssets.js
│   ├── ExchangeConsumables.js
│   ├── faculty.js
│   ├── FacultyEntryModel.js
│   ├── FacultyVerifierModel.js
│   ├── IssueConsumables.js
│   ├── Lands.js
│   ├── NotifyFaculty.js
│   ├── PendingAssetUpdates.js
│   ├── PermanentAsset.js
│   ├── PrincipalModel.js
│   ├── RejectedFaculty.js
│   ├── RejectedAsset.js
│   ├── ReturnedConsumable.js
│   ├── ReturnedAsset.js
│   ├── ServiceAsset.js
│   ├── StoreConsumables.js
│   ├── StorePermanent.js
│   ├── StoreReturns.js
│   ├── TempAsset.js
│   ├── TempBuildingMaintenance.js
│   ├── TempBuildingUpgrade.js
│   ├── TempDispose.js
│   ├── TempService.js
│   ├── UserModel.js
│   └── ViewModel.js
├── node_modules/
│   └── (dependencies, not listed in detail)
├── routes/
│   ├── assetRoutes.js
│   └── facultyRoutes.js
├── package-lock.json
├── package.json
├── .gitignore
└── LICENSE
└── README.md
```

---

5. Core Functionalities

Login/Registration:

- Users (e.g., admins, HOO, SI) log in to access the system.
- Used by: All users.
- *Note*: Authentication details not provided in code; assumed JWT-based.

Dashboard:

- Displays overview of assets (e.g., issued, returned, pending maintenance) and faculty (e.g., verified, pending approval).
- Used by: Admins, HOO.

Asset Management:

- Manage permanent and consumable assets (create, update, delete).
- Issue/return assets, track stock, and manage dead stock.
- Service, maintenance, and disposal workflows.
- Used by: Store managers, admins.

Faculty Management:

- Save, update, and delete faculty records.
- Auto-save drafts for incomplete faculty data.
- Verify and approve faculty records, reject if necessary.
- Used by: Admins, HR, principal, HOO, SI.

Approval Workflows:

- Approve/reject assets (e.g., issues, returns, disposals, building upgrades).
- Approve/reject faculty verification and approval.
- Used by: Admins, HOO, SI, principal.

Search and Filtering:

- Filter assets by type, status, or location.
- Filter faculty by name, faculty type, domains, or status.
- Used by: Admins, managers.

Notifications:

- Notify faculty (e.g., principal, HOO, SI) for actions.
- Track acknowledgment status of notifications.
- Used by: Admins, principal, HOO, SI.

---

6. API Endpoints

Below is a table of major REST API endpoints based on the provided router code:

Method	Endpoint	Description	Auth Required
POST	/api/assets/storeTempAsset	Store temporary asset	Yes
GET	/api/assets/getAllAssets	Fetch all assets	Yes

Method	Endpoint	Description	Auth Required
GET	/api/assets/permanent	Fetch all permanent assets	Yes
POST	/api/assets/uploadInvoice	Upload invoice image	Yes
POST	/api/assets/requestForDisposal	Request asset disposal	Yes
POST	/api/assets/approve/:id	Approve an asset	Yes
POST	/api/faculty/save	Save or update faculty with photo	Yes
GET	/api/faculty/getAllFaculties	Fetch all faculty records	Yes
GET	/api/faculty/search/:id	Fetch faculty by ID	Yes
POST	/api/faculty/filterFaculties	Filter faculty by criteria	Yes
PUT	/api/faculty/verifyFaculty/:id	Verify a faculty member	Yes
POST	/api/faculty/approve/:id	Approve a verified faculty	Yes
POST	/api/faculty/notify/:id	Notify faculty and move to notification	Yes
GET	/api/faculty/notify-all-true	Fetch fully acknowledged notifications	Yes

Note: Authentication is assumed to be required for most endpoints, as they involve sensitive operations. A complete list can be generated by reviewing all routes.

---

7. Database Schema

Database: MongoDB
Collections:
The following collections are based on the provided Mongoose model definitions:

- **assetmanagerlogin** (AssetManagerModel): Stores asset manager user data for authentication and role-based access.
  - **Fields**:
    - name: String, required, trimmed, max length 50 (e.g., "John Doe").
    - password: String, required, min length 6 (e.g., "secure123").
    - role: String, required, default "assetmanager" (e.g., "assetmanager").
    - dob: Date, optional (e.g., "1990-01-01").
    - designation: String, optional, trimmed (e.g., "Asset Manager").
    - phone: String, optional, must match 10-digit format (e.g., "9876543210").
    - organization: String, optional, trimmed (e.g., "CASFOS").
    - ministry: String, optional, trimmed (e.g., "Ministry of Environment").
    - joined: Date, optional, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- **AssetNotification**: Stores notifications related to asset-related actions (e.g., approvals, rejections).
  - **Fields**:
    - assetType: String, required, enum ["Permanent", "Consumable"] (e.g., "Permanent").
    - assetCategory: String, required (e.g., "Furniture").

        - action: String, required, enum of various actions like "asset approved", "issue rejected", "building upgrade approved", "return approved by HOO" (e.g., "asset approved").
        - actionTime: Date, required (e.g., "2025-05-04T12:00:00Z").
        - supplierName: String, optional (e.g., "ABC Suppliers").
        - purchaseDate: Date, optional (e.g., "2024-01-15").
        - billNo: String, optional (e.g., "INV123").
        - receivedBy: String, optional (e.g., "John Doe").
        - itemNames: Array of Strings, optional (e.g., ["Chair", "Table"]).
        - subCategory: String, optional (e.g., "Wood").
        - quantity: Number, optional (e.g., 10).
        - location: String, optional (e.g., "Main Office").
        - rejectionRemarks: String, optional (e.g., "Incomplete documentation").
        - condition: String, optional (e.g., "Good").
        - changedCondition: String, optional (e.g., "Damaged").
        - rejectedAssetId: ObjectId, references RejectedAsset collection (e.g., "60d5ec49e3b3e4a1c8b4567").
        - timestamps: Automatically adds createdAt and updatedAt fields.
- **AutoSaveFaculty**: Stores temporary drafts of faculty data for later completion.
    - **Fields**:
        - staffid: String, required, trimmed (e.g., "FAC1234567890").
        - facultyType: String, required, enum ["internal", "external", "contract"] (e.g., "internal").
        - name: String, trimmed, default "" (e.g., "Dr. Smith").
        - cadre: String, trimmed, default "" (e.g., "IFS").
        - yearOfAllotment: String, must match 4-digit year format, default "" (e.g., "2020").
        - rrSfsDate: Date, default null (e.g., "2020-01-01").
        - dateOfJoining: Date, default null (e.g., "2021-06-15").
        - dateOfRelieve: Date, default null (e.g., "2024-12-31").
        - dateOfBirth: Date, default null (e.g., "1980-05-10").
        - mobileNumber: String, must match 10-digit format starting with 6-9, default "" (e.g., "9876543210").
        - communicationAddress: String, trimmed, default "" (e.g., "123 Main St").
        - permanentAddress: String, trimmed, default "" (e.g., "456 Oak Ave").
        - email: String, trimmed, lowercase, must match email format, default "" (e.g., "john.doe@example.com").
        - photograph: String, default null (e.g., "/uploads/photo123.jpg").
        - presentPlaceOfWorking: String, trimmed, default "" (e.g., "CASFOS Dehradun").
        - status: String, enum ["retired", "serving"], default "serving" (e.g., "serving").
        - conduct: String, trimmed, default "" (e.g., "Good").
        - modulesHandled: Array of Strings, default [] (e.g., ["Silviculture", "Forest Policy"]).
        - majorDomains: Array of Strings, enum of major domain categories (e.g., ["Forest & Wildlife"]).
        - minorDomains: Array of Strings, enum of minor domain values (e.g., ["Silviculture", "Wildlife Conservation"]).
        - areasOfExpertise: String, trimmed, default "" (e.g., "Forest Management").
        - awardsReceived: String, trimmed, default "" (e.g., "Best Faculty 2023").

- inServiceTrainingHandled: String, trimmed, default "" (e.g., "IFS Training 2022").
- publications: Array of Objects:
  - typeOfPublication: String, enum ["Books Published", "Articles Published", "Manuals", "Others"], default "Others" (e.g., "Books Published").
  - title: String, trimmed, default "" (e.g., "Forest Ecology").
  - dateOfPublication: Date, default null (e.g., "2023-01-01").
  - additionalDetails: String, trimmed, default "" (e.g., "Published by Springer").
- educationDetails: Array of Objects:
  - degree: String, trimmed, default "" (e.g., "PhD").
  - specialization: String, trimmed, default "" (e.g., "Forestry").
  - institutionName: String, trimmed, default "" (e.g., "FRI Dehradun").
- coursesHandled: Array of Objects:
  - courseType: String, enum ["InductionTraining", "InserviceTraining", "OtherStakeholder", "SpecialTraining", "SpecialLecture", "Tours", "Others", ""], default "" (e.g., "InductionTraining").
  - batchno: Number, default null (e.g., 202301).
  - title: String, trimmed, default "" (e.g., "Forest Management 101").
  - feedbackRating: Number, min 1, max 10, default null (e.g., 8).
  - feedbackRatings: String, enum ["poor", "good", "verygood", "excellent", ""], default "" (e.g., "excellent").
  - otherCourseType: String, trimmed, default "" (e.g., "Workshop").
- toursAttended: Array of Objects:
  - activityType: String, enum ["FieldExercise", "StudyTour"], default null (e.g., "FieldExercise").
  - days: Number, default null (e.g., 5).
  - startDate: Date, default null (e.g., "2023-06-01").
  - endDate: Date, default null (e.g., "2023-06-05").
  - fieldExerciseName: String, trimmed, default "" (e.g., "Forest Survey").
  - batchNumber: Number, default null (e.g., 202301).
  - location: String, trimmed, default "" (e.g., "Jim Corbett").
  - remarks: String, trimmed, default "" (e.g., "Successful").
  - region: String, trimmed, default "" (e.g., "Uttarakhand").
- examiner: Array of Objects:
  - batchNo: Number, default null (e.g., 202301).
  - year: Number, default null (e.g., 2023).
  - batchType: String, trimmed, default "" (e.g., "IFS Batch").
  - date: Date, default null (e.g., "2023-07-01").
  - paperCorrected: String, trimmed, default "" (e.g., "Silviculture Paper").
- specialSessions: Array of Objects:
  - institutionname: String, trimmed, default "" (e.g., "IGNFA").
  - topic: String, trimmed, default "" (e.g., "Climate Change").
  - date: Date, default null (e.g., "2023-08-01").
  - feedbackRating: Number, default null (e.g., 9).
  - feedbackRating1: String, trimmed, default "" (e.g., "Excellent").
- institution: String, trimmed, default "" (e.g., "CASFOS").
- otherResponsibilities: Array of Objects:

        - responsibility: String, trimmed, default "" (e.g., "Event Coordinator").
    - joined: Date, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- **Building** (BuildingUpgrades): Stores building asset details, including upgrades.
  - **Fields**:
    - assetType: String, required, enum ["Permanent"] (e.g., "Permanent").
    - assetCategory: String, required, default "Building" (e.g., "Building").
    - entryDate: Date, required (e.g., "2024-01-01").
    - subCategory: String, required (e.g., "Residential Quarters").
    - location: String, optional (e.g., "Main Campus").
    - type: String, optional (e.g., "Hostel").
    - buildingNo: String, optional (e.g., "Bldg-001").
    - approvedEstimate: String, optional (e.g., "5000000").
    - plinthArea: String, optional (e.g., "1000 sqm").
    - status: String, optional (e.g., "Active").
    - dateOfConstruction: Date, optional (e.g., "2010-01-01").
    - costOfConstruction: Number, optional (e.g., 5000000).
    - remarks: String, optional (e.g., "Renovated in 2020").
    - upgrades: Array of Objects:
      - year: Number, optional (e.g., 2020).
      - estimate: Number, optional (e.g., 1000000).
      - approvedEstimate: Number, optional (e.g., 900000).
      - dateOfCompletion: Date, optional (e.g., "2020-12-31").
      - defectliabiltyPeriod: String, optional (e.g., "1 year").
      - executionAgency: String, optional (e.g., "ABC Construction").
      - dateOfHandover: Date, optional (e.g., "2021-01-15").
      - documentUrl: String, optional (e.g., "/uploads/doc123.pdf").
    - approvedBuildingPlanUrl: String, optional (e.g., "/uploads/plan123.pdf").
    - kmzOrkmlFileUrl: String, optional (e.g., "/uploads/map123.kmz").
    - timestamps: Automatically adds createdAt and updatedAt fields.
- **BuildingMaintenance**: Stores maintenance records for building assets.
  - **Fields**:
    - assetType: String, required, enum ["Permanent"] (e.g., "Permanent").
    - assetCategory: String, required, default "Building" (e.g., "Building").
    - subCategory: String, optional (e.g., "Residential Quarters").
    - buildingNo: String, required (e.g., "Bldg-001").
    - yearOfMaintenance: Date, required (e.g., "2023-01-01").
    - cost: Number, required (e.g., 500000).
    - description: String, required (e.g., "Roof repair").
    - custody: String, required (e.g., "Maintenance Dept").
    - agency: String, required (e.g., "XYZ Contractors").
    - timestamps: Automatically adds createdAt and updatedAt fields.
- **ConfirmedFaculty**: Stores verified and approved faculty data.
  - **Fields**: (Similar to AutoSaveFaculty, with some differences in required fields and defaults)
    - facultyType: String, required, enum ["internal", "external", "contract"] (e.g., "internal").
    - name: String, required, trimmed (e.g., "Dr. Smith").
    - cadre: String, trimmed (e.g., "IFS").
    - yearOfAllotment: String, must match 4-digit year format (e.g., "2020").
    - rrSfsDate: Date (e.g., "2020-01-01").

        - dateOfJoining: Date (e.g., "2021-06-15").
        - dateOfRelieve: Date (e.g., "2024-12-31").
        - dateOfBirth: Date (e.g., "1980-05-10").
        - mobileNumber: String, must match 10-digit format starting with 6-9 (e.g., "9876543210").
        - communicationAddress: String, trimmed (e.g., "123 Main St").
        - permanentAddress: String, trimmed (e.g., "456 Oak Ave").
        - email: String, trimmed, lowercase, must match email format (e.g., "john.doe@example.com").
        - photograph: String (e.g., "/uploads/photo123.jpg").
        - presentPlaceOfWorking: String, trimmed (e.g., "CASFOS Dehradun").
        - status: String, enum ["retired", "serving"], default "serving" (e.g., "serving").
        - conduct: String, trimmed, default "" (e.g., "Good").
        - remarks: String, trimmed, default "" (e.g., "Approved by Principal").
        - modulesHandled: Array of Strings (e.g., ["Silviculture", "Forest Policy"]).
        - majorDomains: Array of Strings, enum of major domain categories (e.g., ["Forest & Wildlife"]).
        - minorDomains: Array of Strings, enum of minor domain values (e.g., ["Silviculture", "Wildlife Conservation"]).
        - areasOfExpertise: String, trimmed (e.g., "Forest Management").
        - awardsReceived: String, trimmed (e.g., "Best Faculty 2023").
        - inServiceTrainingHandled: String, trimmed (e.g., "IFS Training 2022").
        - publications: Array of Objects (same as AutoSaveFaculty).
        - educationDetails: Array of Objects (same as AutoSaveFaculty, but degree, specialization, and institutionName are required).
        - coursesHandled: Array of Objects (same as AutoSaveFaculty).
        - toursAttended: Array of Objects (same as AutoSaveFaculty, but batchNumber is required).
        - examiner: Array of Objects (same as AutoSaveFaculty).
        - specialSessions: Array of Objects (same as AutoSaveFaculty).
        - institution: String, trimmed (e.g., "CASFOS").
        - otherResponsibilities: Array of Objects (same as AutoSaveFaculty).
        - joined: Date, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
        - staffid: String, required, trimmed, default "FAC" + timestamp (e.g., "FAC1234567890").
- **Consumable**: Stores consumable asset details with validation for categories and items.
    - **Fields**:
        - assetType: String, required, enum ["Consumable"] (e.g., "Consumable").
        - assetCategory: String, required, validated against consumableAssetOptions (e.g., "Stationery").
        - entryDate: Date, required (e.g., "2024-01-01").
        - purchaseDate: Date, required (e.g., "2023-12-15").
        - supplierName: String, optional (e.g., "ABC Suppliers").
        - supplierAddress: String, optional (e.g., "123 Supplier St").
        - source: String, optional (e.g., "Vendor").
        - modeOfPurchase: String, optional (e.g., "Online").
        - billNo: String, optional (e.g., "INV123").
        - receivedBy: String, optional (e.g., "John Doe").
        - billPhotoUrl: String, optional (e.g., "/uploads/bill123.jpg").
        - items: Array of Objects:

- itemName: String, required, validated against itemNameOptions based on subCategory (e.g., "Pen").
- subCategory: String, validated against subCategoryOptions based on assetCategory (e.g., "Stationery").
- itemDescription: String, optional (e.g., "Blue ink pen").
- quantityReceived: Number, required (e.g., 100).
- unitPrice: Number, required (e.g., 10).
- totalPrice: Number, optional (e.g., 1000).
- amcFromDate: Date, optional (e.g., "2024-01-01").
- amcToDate: Date, optional (e.g., "2025-01-01").
- amcCost: Number, optional (e.g., 5000).
- amcPhotoUrl: String, optional (e.g., "/uploads/amc123.jpg").
- itemPhotoUrl: String, optional (e.g., "/uploads/item123.jpg").
- warrantyNumber: String, optional (e.g., "WARR123").
- warrantyValidUpto: Date, optional (e.g., "2025-01-01").
- warrantyPhotoUrl: String, optional (e.g., "/uploads/warranty123.jpg").

- timestamps: Automatically adds createdAt and updatedAt fields.

- **dataentrylogin** (DataEntryModel): Stores data entry user data for authentication and role-based access.
  - **Fields**:
    - name: String, required, trimmed (e.g., "Jane Doe").
    - dob: Date, required (e.g., "1992-03-15").
    - designation: String, required, trimmed (e.g., "Data Entry Operator").
    - phone: String, required, must match 10-digit format (e.g., "9876543210").
    - access: Array of Strings, required, default [] (e.g., ["assets", "faculty"]).
    - organization: String, required, trimmed (e.g., "CASFOS").
    - ministry: String, required, trimmed (e.g., "Ministry of Environment").
    - role: String, default "dataentry" (e.g., "dataentry").
    - password: String, required, min length 6 (e.g., "secure123").
    - joined: Date, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- **DeadStockRegister**: Stores records of dead stock assets (e.g., condemned or unserviceable items).
  - **Fields**:
    - assetType: String, required, enum ["Permanent", "Consumable"] (e.g., "Permanent").
    - assetCategory: String, required (e.g., "Furniture").
    - assetSubCategory: String, required (e.g., "Wood").
    - itemName: String, required (e.g., "Chair").
    - itemDescription: String, required (e.g., "Wooden chair").
    - servicableQuantity: Number, default 0 (e.g., 5, from returned assets).
    - condemnedQuantity: Number, default 0 (e.g., 10, from disposed assets).
    - overallQuantity: Number, required (e.g., 15, from Permanent/Consumable collections).
    - methodOfDisposal: String, required, enum ["Sold", "Auctioned", "Destroyed", "Other"] (e.g., "Auctioned").
    - remarks: String, optional (e.g., "Auctioned in 2024").
    - timestamps: Automatically adds createdAt and updatedAt fields.
- **DisposedAsset**: Stores records of disposed assets, including buildings and non-building assets.
  - **Fields**:

- assetType: String, required (e.g., "Permanent").
- assetCategory: String, required (e.g., "Building").
- itemName: String, optional (e.g., "Fan").
- subCategory: String, optional (e.g., "Appliance").
- itemDescription: String, optional (e.g., "Ceiling fan").
- quantity: Number, optional (e.g., 5).
- itemIds: Array of Strings, optional (e.g., ["FAN001", "FAN002"]).
- purchaseValue: Number, optional (e.g., 5000).
- bookValue: Number, optional (e.g., 2000).
- inspectionDate: Date, optional (e.g., "2024-01-01").
- condemnationDate: Date, optional (e.g., "2024-02-01").
- remark: String, optional (e.g., "Beyond repair").
- disposalValue: Number, optional (e.g., 500).
- methodOfDisposal: String, optional (e.g., "Sold").
- condemnationYear: Number, optional (e.g., 2024, for buildings).
- certificateObtained: String, optional, enum ["Yes", "No"] (e.g., "Yes").
- authority: String, optional (e.g., "PWD").
- dateOfReferenceUrl: String, optional (e.g., "/uploads/ref123.pdf").
- agency: String, optional (e.g., "XYZ Demolition").
- agencyReferenceNumberUrl: String, optional (e.g., "/uploads/agency123.pdf").
- date: Date, optional (e.g., "2024-03-01").
- demolitionPeriod: String, optional (e.g., "3 months").
- demolitionEstimate: Number, optional (e.g., 1000000).

- **ExchangedConsumable**: Stores records of exchanged consumable assets.
  - **Fields**:
    - assetType: String, required, enum ["Consumable"], default "Consumable" (e.g., "Consumable").
    - assetCategory: String, required (e.g., "Stationery").
    - itemName: String, required (e.g., "Pen").
    - subCategory: String, optional (e.g., "Stationery").
    - itemDescription: String, required (e.g., "Blue ink pen").
    - returnedQuantity: Number, required (e.g., 50).
    - exchangeDate: Date, default current time (e.g., "2025-05-04T12:00:00Z").
    - remark: String, optional (e.g., "Exchanged for new stock").
    - pdfUrl: String, optional (e.g., "/uploads/exchange123.pdf").
    - signedPdfUrl: String, optional (e.g., "/uploads/signed123.pdf").
    - originalReturnedAssetId: ObjectId, required, references ReturnedConsumable (e.g., "60d5ec49e3b3e4a1c8b4567").
    - approved: String, enum [null, "yes", "no", "rejected"], default null (e.g., "yes").
    - timestamps: Automatically adds createdAt and updatedAt fields.
- **Faculty**: Stores unverified faculty data before approval.
  - **Fields**: (Similar to ConfirmedFaculty, with additional verified and reentered fields)
    - verified: Boolean, default false (e.g., false).
    - facultyType: String, required, enum ["internal", "external", "contract"] (e.g., "internal").
    - name: String, required, trimmed (e.g., "Dr. Smith").
    - cadre: String, trimmed (e.g., "IFS").
    - yearOfAllotment: String, must match 4-digit year format (e.g., "2020").
    - rrSfsDate: Date (e.g., "2020-01-01").

- dateOfJoining: Date (e.g., "2021-06-15").
- dateOfRelieve: Date (e.g., "2024-12-31").
- dateOfBirth: Date (e.g., "1980-05-10").
- mobileNumber: String, must match 10-digit format starting with 6-9 (e.g., "9876543210").
- communicationAddress: String, trimmed (e.g., "123 Main St").
- permanentAddress: String, trimmed (e.g., "456 Oak Ave").
- email: String, trimmed, lowercase, must match email format (e.g., "john.doe@example.com").
- photograph: String (e.g., "/uploads/photo123.jpg").
- presentPlaceOfWorking: String, trimmed (e.g., "CASFOS Dehradun").
- status: String, enum ["retired", "serving"], default "serving" (e.g., "serving").
- conduct: String, trimmed, default "" (e.g., "Good").
- modulesHandled: Array of Strings (e.g., ["Silviculture", "Forest Policy"]).
- majorDomains: Array of Strings, enum of major domain categories (e.g., ["Forest & Wildlife"]).
- minorDomains: Array of Strings, enum of minor domain values (e.g., ["Silviculture", "Wildlife Conservation"]).
- areasOfExpertise: String, trimmed (e.g., "Forest Management").
- awardsReceived: String, trimmed (e.g., "Best Faculty 2023").
- inServiceTrainingHandled: String, trimmed (e.g., "IFS Training 2022").
- publications: Array of Objects (same as AutoSaveFaculty).
- educationDetails: Array of Objects (same as AutoSaveFaculty, but degree, specialization, and institutionName are required).
- coursesHandled: Array of Objects (same as AutoSaveFaculty).
- toursAttended: Array of Objects (same as AutoSaveFaculty, but batchNumber is required).
- examiner: Array of Objects (same as AutoSaveFaculty).
- specialSessions: Array of Objects (same as AutoSaveFaculty).
- institution: String, trimmed (e.g., "CASFOS").
- otherResponsibilities: Array of Objects (same as AutoSaveFaculty).
- joined: Date, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- staffid: String, required, trimmed, default "FAC" + timestamp (e.g., "FAC1234567890").
- reentered: Boolean, default false (e.g., false).

- **facultyentrystafflogin** (FacultyEntryStaffModel): Stores faculty entry staff user data for authentication and role-based access.
    - **Fields**: (Similar to assetmanagerlogin)
        - name: String, required, trimmed, max length 50 (e.g., "Alice Smith").
        - password: String, required, min length 6 (e.g., "secure123").
        - role: String, required, default "facultyentrystaff" (e.g., "facultyentrystaff").
        - dob: Date, optional (e.g., "1990-01-01").
        - designation: String, optional, trimmed (e.g., "Faculty Entry Staff").
        - phone: String, optional, must match 10-digit format (e.g., "9876543210").
        - organization: String, optional, trimmed (e.g., "CASFOS").
        - ministry: String, optional, trimmed (e.g., "Ministry of Environment").
        - joined: Date, optional, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- **facultyverifierlogin** (FacultyVerifierModel): Stores faculty verifier user data for authentication and role-based access.
    - **Fields**: (Similar to assetmanagerlogin)

    - name: String, required, trimmed, max length 50 (e.g., "Bob Johnson").
    - password: String, required, min length 6 (e.g., "secure123").
    - role: String, required, default "facultyverifier" (e.g., "facultyverifier").
    - dob: Date, optional (e.g., "1990-01-01").
    - designation: String, optional, trimmed (e.g., "Faculty Verifier").
    - phone: String, optional, must match 10-digit format (e.g., "9876543210").
    - organization: String, optional, trimmed (e.g., "CASFOS").
    - ministry: String, optional, trimmed (e.g., "Ministry of Environment").
    - joined: Date, optional, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").

- **headofficelogin** (HeadOfOfficeModel): Stores head of office user data for authentication and role-based access.
  - **Fields**: (Similar to assetmanagerlogin)
    - name: String, required, trimmed, max length 50 (e.g., "Carol Williams").
    - password: String, required, min length 6 (e.g., "secure123").
    - role: String, required, default "headofoffice" (e.g., "headofoffice").
    - dob: Date, optional (e.g., "1990-01-01").
    - designation: String, optional, trimmed (e.g., "Head of Office").
    - phone: String, optional, must match 10-digit format (e.g., "9876543210").
    - organization: String, optional, trimmed (e.g., "CASFOS").
    - ministry: String, optional, trimmed (e.g., "Ministry of Environment").
    - joined: Date, optional, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- **IssuedConsumable**: Stores records of issued consumable assets.
  - **Fields**:
    - assetType: String, required, enum ["Consumable"], default "Consumable" (e.g., "Consumable").
    - assetCategory: String, required, min length 2, max length 50 (e.g., "Stationery").
    - itemName: String, required, min length 2, max length 100 (e.g., "Pen").
    - subCategory: String, optional (e.g., "Stationery").
    - itemDescription: String, required, min length 5, max length 500 (e.g., "Blue ink pen").
    - issues: Array of Objects:
      - issuedTo: String, required, min length 2, max length 100 (e.g., "John Doe").
      - location: String, optional (e.g., "Main Office").
      - quantity: Number, required, min 1, max 10000 (e.g., 50).
      - issuedDate: Date, default current time, max current time (e.g., "2025-05-04T12:00:00Z").
- **IssuedPermanent**: Stores records of issued permanent assets.
  - **Fields**:
    - assetType: String, required, enum ["Permanent"], default "Permanent" (e.g., "Permanent").
    - assetCategory: String, required, min length 2, max length 50 (e.g., "Furniture").
    - itemName: String, required, min length 2, max length 100 (e.g., "Chair").
    - subCategory: String, optional (e.g., "Wood").
    - itemDescription: String, required, min length 5, max length 500 (e.g., "Wooden chair").
    - issues: Array of Objects:

        - issuedTo: String, required, min length 2, max length 100 (e.g., "John Doe").
        - location: String, optional (e.g., "Main Office").
        - quantity: Number, required, min 1, max 10000 (e.g., 5).
        - issuedDate: Date, default current time, max current time (e.g., "2025-05-04T12:00:00Z").
        - issuedIds: Array of Strings (e.g., ["CHAIR001", "CHAIR002"]).
- **Land**: Stores land asset details.
    - **Fields**:
        - assetType: String, required, enum ["Permanent"] (e.g., "Permanent").
        - assetCategory: String, required, default "Land" (e.g., "Land").
        - entryDate: Date, required (e.g., "2024-01-01").
        - subCategory: String, required (e.g., "Land with Building").
        - location: String, required (e.g., "Main Campus").
        - status: String, required (e.g., "Active").
        - dateOfPossession: Date, required (e.g., "2010-01-01").
        - controllerOrCustody: String, required (e.g., "Estate Dept").
        - details: String, optional (e.g., "50 acres").
        - timestamps: Automatically adds createdAt and updatedAt fields.
- **NotifyFaculty**: Stores faculty notification records with acknowledgment flags, inheriting fields from ConfirmedFaculty.
    - **Fields**: (Includes all fields from ConfirmedFaculty, plus the following)
        - notifyprincipal: Boolean, default false (e.g., false).
        - notifyhoo: Boolean, default false (e.g., false).
        - notifysi: Boolean, default false (e.g., false).
        - notifyremarks: String, trimmed, default "" (e.g., "Pending HOO approval").
        - notificationDate: Date, default current time (e.g., "2025-05-04T12:00:00Z").
- **PendingAssetUpdate**: Stores pending updates for assets awaiting approval.
    - **Fields**:
        - assetId: ObjectId, required (e.g., "60d5ec49e3b3e4a1c8b4567").
        - assetType: String, required, enum ["Permanent", "Consumable"] (e.g., "Permanent").
        - originalData: Object, required (e.g., { itemName: "Chair", quantity: 5 }).
        - updatedData: Object, required (e.g., { itemName: "Chair", quantity: 10 }).
        - status: String, enum ["pending", "approved", "rejected"], default "pending" (e.g., "pending").
        - rejectionRemarks: String, optional (e.g., "Invalid quantity").
        - timestamps: Automatically adds createdAt and updatedAt fields.
- **Permanent**: Stores permanent asset details with validation for categories and items.
    - **Fields**:
        - assetType: String, required, enum ["Permanent"] (e.g., "Permanent").
        - assetCategory: String, required, validated against permanentAssetOptions (e.g., "Furniture").
        - entryDate: Date, required (e.g., "2024-01-01").
        - purchaseDate: Date, required (e.g., "2023-12-15").
        - supplierName: String, optional (e.g., "ABC Suppliers").
        - supplierAddress: String, optional (e.g., "123 Supplier St").
        - source: String, optional (e.g., "Vendor").
        - modeOfPurchase: String, optional (e.g., "Online").
        - billNo: String, optional (e.g., "INV123").
        - receivedBy: String, optional (e.g., "John Doe").

- billPhotoUrl: String, optional (e.g., "/uploads/bill123.jpg").
- items: Array of Objects:
    - itemName: String, required, validated against itemNameOptions based on subCategory (e.g., "Chair").
    - subCategory: String, validated against subCategoryOptions based on assetCategory (e.g., "Wood").
    - itemDescription: String, required (e.g., "Wooden chair").
    - quantityReceived: Number, required (e.g., 10).
    - unitPrice: Number, required (e.g., 500).
    - totalPrice: Number, optional (e.g., 5000).
    - amcFromDate: Date, optional (e.g., "2024-01-01").
    - amcToDate: Date, optional (e.g., "2025-01-01").
    - amcCost: Number, optional (e.g., 10000).
    - amcPhotoUrl: String, optional (e.g., "/uploads/amc123.jpg").
    - itemPhotoUrl: String, optional (e.g., "/uploads/item123.jpg").
    - warrantyNumber: String, optional (e.g., "WARR123").
    - warrantyValidUpto: Date, optional (e.g., "2025-01-01").
    - warrantyPhotoUrl: String, optional (e.g., "/uploads/warranty123.jpg").
    - itemIds: Array of Strings (e.g., ["CHAIR001", "CHAIR002"]).
- timestamps: Automatically adds createdAt and updatedAt fields.

- **principallogin** (PrincipalModel): Stores principal user data for authentication and role-based access.
    - **Fields**: (Similar to assetmanagerlogin)
        - name: String, required, trimmed, max length 50 (e.g., "David Brown").
        - password: String, required, min length 6 (e.g., "secure123").
        - role: String, required, default "principal" (e.g., "principal").
        - dob: Date, optional (e.g., "1990-01-01").
        - designation: String, optional, trimmed (e.g., "Principal").
        - phone: String, optional, must match 10-digit format (e.g., "9876543210").
        - organization: String, optional, trimmed (e.g., "CASFOS").
        - ministry: String, optional, trimmed (e.g., "Ministry of Environment").
        - joined: Date, optional, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- **RejectedAsset**: Stores records of rejected assets, including buildings, issues, returns, services, and disposals.
    - **Fields**:
        - assetType: String, required, enum ["Permanent", "Consumable"] (e.g., "Permanent").
        - assetCategory: String, required (e.g., "Furniture").
        - subCategory: String, optional (e.g., "Wood").
        - location: String, optional (e.g., "Main Office").
        - entryDate: Date, optional (e.g., "2024-01-01").
        - remarks: String, optional (e.g., "Rejected due to incomplete documentation").
        - rejectionRemarks: String, required (e.g., "Missing bill").
        - purchaseDate: Date, optional (e.g., "2023-12-15").
        - supplierName: String, optional (e.g., "ABC Suppliers").
        - supplierAddress: String, optional (e.g., "123 Supplier St").
        - source: String, optional (e.g., "Vendor").
        - modeOfPurchase: String, optional (e.g., "Online").

- billNo: String, optional (e.g., "INV123").
- receivedBy: String, optional (e.g., "John Doe").
- billPhotoUrl: String, optional (e.g., "/uploads/bill123.jpg").
- items: Array of Objects:
    - itemName: String, optional (e.g., "Chair").
    - subCategory: String, optional (e.g., "Wood").
    - itemDescription: String, optional (e.g., "Wooden chair").
    - quantityReceived: Number, optional (e.g., 10).
    - unitPrice: Number, optional (e.g., 500).
    - totalPrice: Number, optional (e.g., 5000).
    - amcFromDate: Date, optional (e.g., "2024-01-01").
    - amcToDate: Date, optional (e.g., "2025-01-01").
    - amcCost: Number, optional (e.g., 10000).
    - amcPhotoUrl: String, optional (e.g., "/uploads/amc123.jpg").
    - warrantyNumber: String, optional (e.g., "WARR123").
    - warrantyValidUpto: Date, optional (e.g., "2025-01-01").
    - warrantyPhotoUrl: String, optional (e.g., "/uploads/warranty123.jpg").
    - itemIds: Array of Strings, optional (e.g., ["CHAIR001", "CHAIR002"]).
    - itemPhotoUrl: String, optional (e.g., "/uploads/item123.jpg").
- type: String, optional (e.g., "Hostel", for buildings).
- buildingNo: String, optional (e.g., "Bldg-001").
- plinthArea: String, optional (e.g., "1000 sqm").
- dateOfConstruction: Date, optional (e.g., "2010-01-01").
- costOfConstruction: Number, optional (e.g., 5000000).
- approvedEstimate: String, optional (e.g., "5000000").
- approvedBuildingPlanUrl: String, optional (e.g., "/uploads/plan123.pdf").
- kmzOrkmlFileUrl: String, optional (e.g., "/uploads/map123.kmz").
- upgrades: Array of Objects:
    - year: Number, optional (e.g., 2020).
    - estimate: Number, optional (e.g., 1000000).
    - approvedEstimate: Number, optional (e.g., 900000).
    - dateOfCompletion: Date, optional (e.g., "2020-12-31").
    - defectliabiltyPeriod: String, optional (e.g., "1 year").
    - executionAgency: String, optional (e.g., "ABC Construction").
- itemName: String, optional (e.g., "Chair", from TempIssueSchema).
- itemDescription: String, optional (e.g., "Wooden chair").
- issuedTo: String, optional (e.g., "John Doe").
- quantity: Number, optional (e.g., 5).
- issuedIds: Array of Strings, optional (e.g., ["CHAIR001", "CHAIR002"]).
- pdfUrl: String, optional (e.g., "/uploads/issue123.pdf").
- signedPdfUrl: String, optional (e.g., "/uploads/signed123.pdf").
- acknowledged: String, enum ["yes", "no"], default "no" (e.g., "no").
- rejected: String, enum ["yes", "no"], default "yes" (e.g., "yes").
- status: String, optional (e.g., "Good", from returns).
- returnQuantity: Number, optional (e.g., 5).
- itemId: String, optional (e.g., "CHAIR001").
- returnIds: Array of Strings, optional (e.g., ["CHAIR001", "CHAIR002"]).
- approved: String, enum ["yes", "no"], default "no" (e.g., "no").
- serviceNo: String, optional (e.g., "SERV123", from TempServicedSchema).
- serviceDate: Date, optional (e.g., "2024-01-01").

- serviceAmount: Number, optional (e.g., 5000).
- itemIds: Array of Strings, optional (e.g., ["CHAIR001", "CHAIR002"]).
- purchaseValue: Number, optional (e.g., 5000, from TempDisposeAssetSchema).
- bookValue: Number, optional (e.g., 2000).
- inspectionDate: Date, optional (e.g., "2024-01-01").
- condemnationDate: Date, optional (e.g., "2024-02-01").
- disposalValue: Number, optional (e.g., 500).
- methodOfDisposal: String, optional (e.g., "Sold").
- condemnationYear: String, optional (e.g., "2024").
- certificateObtained: String, optional (e.g., "Yes").
- authority: String, optional (e.g., "PWD").
- dateOfReferenceUrl: String, optional (e.g., "/uploads/ref123.pdf").
- agency: String, optional (e.g., "XYZ Demolition").
- agencyReferenceNumberUrl: String, optional (e.g., "/uploads/agency123.pdf").
- date: Date, optional (e.g., "2024-03-01").
- demolitionPeriod: String, optional (e.g., "3 months").
- demolitionEstimate: Number, optional (e.g., 1000000).
- yearOfMaintenance: Date, optional (e.g., "2023-01-01").
- cost: Number, optional (e.g., 500000).
- description: String, optional (e.g., "Roof repair").
- custody: String, optional (e.g., "Maintenance Dept").
- agency: String, optional (e.g., "XYZ Contractors").
- remark: String, optional (e.g., "Beyond repair").
- assetId: ObjectId, optional (e.g., "60d5ec49e3b3e4a1c8b4567").
- updatedData: Object, optional (e.g., { itemName: "Chair", quantity: 10 }).
- timestamps: Automatically adds createdAt and updatedAt fields.

- **RejectedFaculty**: Stores rejected faculty data with rejection details.
  - **Fields**: (Similar to Faculty, with additional rejection fields)
    - verified: Boolean, default false (e.g., false).
    - facultyType: String, required, enum ["internal", "external", "contract"] (e.g., "internal").
    - name: String, required, trimmed (e.g., "Dr. Smith").
    - cadre: String, trimmed (e.g., "IFS").
    - yearOfAllotment: String, must match 4-digit year format (e.g., "2020").
    - rrSfsDate: Date (e.g., "2020-01-01").
    - dateOfJoining: Date (e.g., "2021-06-15").
    - dateOfRelieve: Date (e.g., "2024-12-31").
    - dateOfBirth: Date (e.g., "1980-05-10").
    - mobileNumber: String, must match 10-digit format starting with 6-9 (e.g., "9876543210").
    - communicationAddress: String, trimmed (e.g., "123 Main St").
    - permanentAddress: String, trimmed (e.g., "456 Oak Ave").
    - email: String, trimmed, lowercase, must match email format (e.g., "john.doe@example.com").
    - photograph: String (e.g., "/uploads/photo123.jpg").
    - presentPlaceOfWorking: String, trimmed (e.g., "CASFOS Dehradun").
    - status: String, enum ["retired", "serving"], default "serving" (e.g., "serving").
    - conduct: String, trimmed, default "" (e.g., "Good").
    - modulesHandled: Array of Strings (e.g., ["Silviculture", "Forest Policy"]).

        - majorDomains: Array of Strings, enum of major domain categories (e.g., ["Forest & Wildlife"]).
        - minorDomains: Array of Strings, enum of minor domain values (e.g., ["Silviculture", "Wildlife Conservation"]).
        - areasOfExpertise: String, trimmed (e.g., "Forest Management").
        - awardsReceived: String, trimmed (e.g., "Best Faculty 2023").
        - inServiceTrainingHandled: String, trimmed (e.g., "IFS Training 2022").
        - publications: Array of Objects (same as AutoSaveFaculty).
        - educationDetails: Array of Objects (same as AutoSaveFaculty, but degree, specialization, and institutionName are required).
        - coursesHandled: Array of Objects (same as AutoSaveFaculty).
        - toursAttended: Array of Objects (same as AutoSaveFaculty, but batchNumber is required).
        - examiner: Array of Objects (same as AutoSaveFaculty).
        - specialSessions: Array of Objects (same as AutoSaveFaculty).
        - institution: String, trimmed (e.g., "CASFOS").
        - otherResponsibilities: Array of Objects (same as AutoSaveFaculty).
        - joined: Date, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
        - staffid: String, required, trimmed, default "FAC" + timestamp (e.g., "FAC1234567890").
        - verificationRejection: Boolean, default false (e.g., false).
        - approvalRejection: Boolean, default false (e.g., false).
        - rejectionRemarks: String, trimmed, default "" (e.g., "Incomplete education details").
        - timestamps: Automatically adds createdAt and updatedAt fields.
- **RejectedUser**: Stores rejected user data with rejection details.
    - **Fields**:
        - name: String, required, trimmed, max length 50 (e.g., "Eve Adams").
        - password: String, required, min length 6 (e.g., "secure123").
        - role: String, required (e.g., "assetmanager").
        - dob: Date, optional (e.g., "1990-01-01").
        - designation: String, optional, trimmed (e.g., "Asset Manager").
        - phone: String, optional, must match 10-digit format (e.g., "9876543210").
        - organization: String, optional, trimmed (e.g., "CASFOS").
        - ministry: String, optional, trimmed (e.g., "Ministry of Environment").
        - remark: String, required, trimmed (e.g., "Invalid phone number").
        - rejectedAt: Date, optional, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- **ReturnedConsumable**: Stores records of returned consumable assets.
    - **Fields**:
        - assetType: String, required, enum ["Consumable"] (e.g., "Consumable").
        - assetCategory: String, optional (e.g., "Stationery").
        - itemName: String, required (e.g., "Pen").
        - subCategory: String, optional (e.g., "Stationery").
        - itemDescription: String, optional (e.g., "Blue ink pen").
        - location: String, optional (e.g., "Main Office").
        - status: String, enum ["Good", "exchange", "dispose", "returned"], default "returned" (e.g., "returned").
        - pdfUrl: String, optional (e.g., "/uploads/return123.pdf").
        - signedPdfUrl: String, optional (e.g., "/uploads/signed123.pdf").
        - remark: String, optional (e.g., "Returned due to excess stock").

        - returnQuantity: Number, required (e.g., 50).
        - approved: String, enum ["yes", "no"], default null (e.g., "yes").
        - rejectionRemarks: String, default null (e.g., "Invalid return reason").
        - disposedEntry: String, enum ["yes", "no", null], default null (e.g., "yes").
        - hooapproval: String, enum ["waiting", null], default null (e.g., "waiting").
        - rejectedDisposal: String, enum ["yes", "no", null], default null (e.g., "no").
        - disposalRemarks: String, default null (e.g., "Disposal rejected due to policy").
- **ReturnedPermanent**: Stores records of returned permanent assets.
    - **Fields**:
        - assetType: String, required, enum ["Permanent"], default "Permanent" (e.g., "Permanent").
        - assetCategory: String, optional (e.g., "Furniture").
        - itemName: String, optional (e.g., "Chair").
        - subCategory: String, optional (e.g., "Wood").
        - itemDescription: String, optional (e.g., "Wooden chair").
        - location: String, optional (e.g., "Main Office").
        - itemId: String, optional (e.g., "CHAIR001").
        - status: String, enum ["Good", "service", "dispose", "returned"], default "returned" (e.g., "returned").
        - remark: String, optional (e.g., "Returned due to replacement").
        - pdfUrl: String, optional (e.g., "/uploads/return123.pdf").
        - signedPdfUrl: String, optional (e.g., "/uploads/signed123.pdf").
        - approved: String, enum ["yes", "no"], default null (e.g., "yes").
        - servicedEntry: String, enum ["yes", "no", null], default null (e.g., "yes").
        - hooapproval: String, enum ["waiting", null], default null (e.g., "waiting").
        - servicedRejection: String, enum ["yes", "no", null], default null (e.g., "no").
        - servicedRejectionRemarks: String, default null (e.g., "Service not required").
        - rejectionRemarks: String, default null (e.g., "Invalid return reason").
        - disposedEntry: String, enum ["yes", "no", null], default null (e.g., "yes").
        - rejectedDisposal: String, enum ["yes", "no", null], default null (e.g., "no").
        - disposalRemarks: String, default null (e.g., "Disposal rejected due to policy").
- **ServicedAsset**: Stores records of serviced assets.
    - **Fields**:
        - assetType: String, required (e.g., "Permanent").
        - assetCategory: String, required (e.g., "Furniture").
        - itemName: String, required (e.g., "Chair").
        - subCategory: String, optional (e.g., "Wood").
        - itemDescription: String, required (e.g., "Wooden chair").
        - itemIds: Array of Strings, optional (e.g., ["CHAIR001", "CHAIR002"]).
        - serviceNo: String, required (e.g., "SERV123").
        - serviceDate: Date, required (e.g., "2024-01-01").
        - serviceAmount: Number, required (e.g., 5000).
- **StoreConsumable**: Stores stock details of consumable assets.
    - **Fields**:
        - assetCategory: String, required, min length 2, max length 50 (e.g., "Stationery").
        - itemName: String, required, min length 2, max length 100, validated against itemNameOptions based on subCategory (e.g., "Pen").
        - subCategory: String, optional (e.g., "Stationery").
        - itemDescription: String, required, min length 5, max length 500 (e.g., "Blue ink pen").

        - inStock: Number, default 0, min 0, max 100000 (e.g., 100).
- **storekeeperlogin** (StorekeeperModel): Stores storekeeper user data for authentication and role-based access.
    - **Fields**: (Similar to assetmanagerlogin)
        - name: String, required, trimmed, max length 50 (e.g., "Frank Wilson").
        - password: String, required, min length 6 (e.g., "secure123").
        - role: String, required, default "storekeeper" (e.g., "storekeeper").
        - dob: Date, optional (e.g., "1990-01-01").
        - designation: String, optional, trimmed (e.g., "Storekeeper").
        - phone: String, optional, must match 10-digit format (e.g., "9876543210").
        - organization: String, optional, trimmed (e.g., "CASFOS").
        - ministry: String, optional, trimmed (e.g., "Ministry of Environment").
        - joined: Date, optional, defaults to current IST time (e.g., "2025-05-04T12:00:00Z").
- **StorePermanent**: Stores stock details of permanent assets.
    - **Fields**:
        - assetCategory: String, required, min length 2, max length 50, validated against permanentAssetOptions (e.g., "Furniture").
        - itemName: String, required, min length 2, max length 100, validated against itemNameOptions based on subCategory (e.g., "Chair").
        - subCategory: String, optional, validated against subCategoryOptions based on assetCategory (e.g., "Wood").
        - itemDescription: String, required, min length 1, max length 500 (e.g., "Wooden chair").
        - inStock: Number, default 0, min 0, max 100000 (e.g., 10).
        - itemIds: Array of Strings, optional (e.g., ["CHAIR001", "CHAIR002"]).
- **StoreReturn**: Stores records of assets returned to the store.
    - **Fields**:
        - assetType: String, required, enum ["Permanent", "Consumable"] (e.g., "Permanent").
        - assetCategory: String, required, min length 2, max length 50 (e.g., "Furniture").
        - itemName: String, required, min length 2, max length 100 (e.g., "Chair").
        - subCategory: String, optional, max length 50 (e.g., "Wood").
        - itemDescription: String, required, min length 1, max length 500 (e.g., "Wooden chair").
        - location: String, required, default "Store" (e.g., "Store").
        - itemIds: Array of Strings, default [] (e.g., ["CHAIR001", "CHAIR002"]).
        - returnedQuantity: Number, min 0 (e.g., 5).
        - pdfUrl: String, optional (e.g., "/uploads/return123.pdf").
        - signedPdfUrl: String, optional (e.g., "/uploads/signed123.pdf").
        - status: String, default "pending" (e.g., "pending").
        - source: String, required, default "store" (e.g., "store").
        - originalStoreId: ObjectId, required (e.g., "60d5ec49e3b3e4a1c8b4567").
        - createdAt: Date, default current time (e.g., "2025-05-04T12:00:00Z").
- **TempAsset**: Stores temporary asset data (e.g., pending approval) for Permanent, Consumable, Land, and Building assets.
    - **Fields**:
        - assetType: String, optional, enum ["Permanent", "Consumable"] (e.g., "Permanent").

    - assetCategory: String, optional, validated against permanentAssetOptions or consumableAssetOptions based on assetType (e.g., "Furniture").
    - entryDate: Date, optional (e.g., "2024-01-01").
    - subCategory: String, optional, validated against subCategoryOptions based on assetCategory (e.g., "Wood").
    - location: String, optional (e.g., "Main Office").
    - status: String, optional (e.g., "Pending").
    - purchaseDate: Date, optional (e.g., "2023-12-15").
    - supplierName: String, optional (e.g., "ABC Suppliers").
    - supplierAddress: String, optional (e.g., "123 Supplier St").
    - source: String, optional (e.g., "Vendor").
    - modeOfPurchase: String, optional (e.g., "Online").
    - billNo: String, optional (e.g., "INV123").
    - receivedBy: String, optional (e.g., "John Doe").
    - billPhotoUrl: String, optional (e.g., "/uploads/bill123.jpg").
    - items: Array of Objects:
        - itemName: String, optional, validated against itemNameOptions based on subCategory (e.g., "Chair").
        - subCategory: String, optional, validated against subCategoryOptions based on assetCategory (e.g., "Wood").
        - itemDescription: String, optional (e.g., "Wooden chair").
        - quantityReceived: Number, optional (e.g., 10).
        - unitPrice: Number, optional (e.g., 500).
        - totalPrice: Number, optional (e.g., 5000).
        - amcFromDate: Date, optional (e.g., "2024-01-01").
        - amcToDate: Date, optional (e.g., "2025-01-01").
        - amcCost: Number, optional (e.g., 10000).
        - amcPhotoUrl: String, optional (e.g., "/uploads/amc123.jpg").
        - itemPhotoUrl: String, optional (e.g., "/uploads/item123.jpg").
        - warrantyNumber: String, optional (e.g., "WARR123").
        - warrantyValidUpto: Date, optional (e.g., "2025-01-01").
        - warrantyPhotoUrl: String, optional (e.g., "/uploads/warranty123.jpg").
        - itemIds: Array of Strings, optional (e.g., ["CHAIR001", "CHAIR002"]).
    - dateOfPossession: Date, optional (e.g., "2010-01-01", for Land).
    - controllerOrCustody: String, optional (e.g., "Estate Dept").
    - details: String, optional (e.g., "50 acres").
    - type: String, optional (e.g., "Hostel", for Building).
    - buildingNo: String, optional (e.g., "Bldg-001").
    - approvedEstimate: String, optional (e.g., "5000000").
    - plinthArea: String, optional (e.g., "1000 sqm").
    - dateOfConstruction: Date, optional (e.g., "2010-01-01").
    - costOfConstruction: Number, optional (e.g., 5000000).
    - remarks: String, optional (e.g., "Renovated in 2020").
    - approvedBuildingPlanUrl: String, optional (e.g., "/uploads/plan123.pdf").
    - kmzOrkmlFileUrl: String, optional (e.g., "/uploads/map123.kmz").
    - timestamps: Automatically adds createdAt and updatedAt fields.

- **TempBuildingMaintenance**: Stores temporary building maintenance records pending approval.
  - **Fields**:
    - assetType: String, required, enum ["Permanent"] (e.g., "Permanent").

    - subCategory: String, optional (e.g., "Residential Quarters").
    - assetCategory: String, required, default "Building" (e.g., "Building").
    - buildingNo: String, required (e.g., "Bldg-001").
    - yearOfMaintenance: Date, required (e.g., "2023-01-01").
    - cost: Number, required (e.g., 500000).
    - description: String, required (e.g., "Roof repair").
    - custody: String, required (e.g., "Maintenance Dept").
    - agency: String, required (e.g., "XYZ Contractors").
    - status: String, default "pending", enum ["pending", "approved", "rejected"] (e.g., "pending").
    - timestamps: Automatically adds createdAt and updatedAt fields.
- **TempBuildingUpgrade**: Stores temporary building upgrade records pending approval.
  - **Fields**:
    - subCategory: String, required (e.g., "Residential Quarters").
    - upgrades: Array of Objects:
      - year: Number, required (e.g., 2020).
      - estimate: Number, required, min 0 (e.g., 1000000).
      - approvedEstimate: Number, required, min 0 (e.g., 900000).
      - dateOfCompletion: Date, required (e.g., "2020-12-31").
      - defectliabiltyPeriod: String, required (e.g., "1 year").
      - executionAgency: String, required (e.g., "ABC Construction").
      - dateOfHandover: Date, required (e.g., "2021-01-15").
      - documentUrl: String, optional (e.g., "/uploads/doc123.pdf").
    - timestamps: Automatically adds createdAt and updatedAt fields.
- **tempDispose**: Stores temporary disposal records for assets pending approval.
  - **Fields**:
    - assetType: String, required (e.g., "Permanent").
    - assetCategory: String, required (e.g., "Furniture").
    - itemName: String, optional (e.g., "Chair").
    - subCategory: String, optional (e.g., "Wood").
    - itemDescription: String, optional (e.g., "Wooden chair").
    - quantity: Number, optional (e.g., 5).
    - itemIds: Array of Strings, optional (e.g., ["CHAIR001", "CHAIR002"]).
    - purchaseValue: Number, optional (e.g., 5000).
    - bookValue: Number, optional (e.g., 2000).
    - inspectionDate: Date, optional (e.g., "2024-01-01").
    - condemnationDate: Date, optional (e.g., "2024-02-01").
    - remark: String, optional (e.g., "Beyond repair").
    - disposalValue: Number, optional (e.g., 500).
    - methodOfDisposal: String, optional (e.g., "Sold").
    - condemnationYear: Number, optional (e.g., 2024, for buildings).
    - certificateObtained: String, optional, enum ["Yes", "No"] (e.g., "Yes").
    - authority: String, optional (e.g., "PWD").
    - dateOfReferenceUrl: String, optional (e.g., "/uploads/ref123.pdf").
    - agency: String, optional (e.g., "XYZ Demolition").
    - agencyReferenceNumberUrl: String, optional (e.g., "/uploads/agency123.pdf").
    - date: Date, optional (e.g., "2024-03-01").
    - demolitionPeriod: String, optional (e.g., "3 months").
    - demolitionEstimate: Number, optional (e.g., 1000000).

8. Frontend Components

The below are some of the essential frontend components.

Login.jsx

- **Role Selection Panel:**
    - Choose from seven roles (e.g., Head of Office, Principal, Asset Manager) using radio buttons.
    - Default role is Head of Office.
- **Login Form:**
    - Username field with a user icon and placeholder text.
    - Password field with a lock icon and placeholder text.
    - Password visibility toggle (eye icon) to show or hide the password.
    - Login button to submit credentials.
- **Visual Elements:**
    - Displays the CASFOS logo above the login form.
    - Modern design with fade-in animation on page load.
    - Split layout: role selection on the left, login form on the right.
- **User Feedback:**
    - Loading indicator ("Logging in...") on the login button during submission.
    - Error messages (e.g., "Invalid login credentials", "Enter username and password") below the form.
- **Navigation Links:**
    - "Create your Account" link with an arrow icon, directing to the registration page.

Admindashboard.jsx

- **Sidebar Navigation**:
    - Displays "ADMIN" branding at the top.
    - Menu options: Home (active), Registration Approval, User Management, Asset Approval, Faculty Approval, Asset View, Faculty View.
    - Logout link at the bottom.
- **Header Section**:
    - Shows "Dashboard" title.
    - Displays the logged-in username (or "Guest" if not provided) with a user icon.
- **User Summary Section**:

    - Title: "Total Registered Users" with a subtitle "User's Summary".
    - Three cards displaying counts for:
        - Admin (with shield icon).
        - Data Entry Staff (with keyboard icon).
        - Data Viewer (with eye icon).
    - Each card has a distinct background color and icon color.
- **Upcoming AMC Expiry Table**:
    - Title: "Upcoming AMC Expiry" with a red exclamation icon.
    - Table columns: Asset Type, Location, Accessory Name, AMC Date.
    - Displays assets with AMC dates within the next month.

Assetstore.jsx

Core Functionalities

- **Store/Receipt Entry**
    - Add new Permanent/Consumable assets.
    - Include purchase details, bills, and item-specific documents.
    - Assign unique IDs for Permanent assets.
- **Returned Assets**
    - View and manage returned assets.
    - Set condition: To Be Serviced, Disposed, or Exchanged.
    - Generate PDF receipt and upload signed copy.
- **Serviced/Maintenance**
    - Log service info (AMC, warranty).
    - Submit building maintenance requests.
- **Condemnation**
    - Request asset disposal or building condemnation.
    - Provide justification and related documents.
- **Building Upgrade**
    - Record upgrades with year, cost, agency.
    - Submit upgrade entries in bulk.

---

2. User Flow

- **Store/Receipt Entry**

    - Select asset type and category.
    - Fill in purchase details and upload supporting files.
    - Assign IDs for Permanent items.
- **Returned Assets**
    - Access list of returned items.
    - Set asset condition, generate & upload receipt.
    - Mark item as “Done”.
- **Building Upgrade**
    - Choose building subcategory.
    - Add upgrade entries (year, cost, etc.).
    - Submit all entries together.

Headofofficedashboard.jsx

- **Login / Access Dashboard**
    - User arrives at the HOO dashboard with a username in URL.
- **View Notifications**
    - Bell icon shows pending approvals.
    - Clicking opens expandable panel.
- **Manage Notifications**
    - Expand → View Details → Acknowledge / Clear.
- **Use Sidebar**
    - Navigate to approvals or view/updation pages.
- **Explore Info Sections**
    - Scroll down to About, History, Location & Contact.

9. Authentication & Authorization

Mechanism:

- Assumed to use JWT (JSON Web Tokens) for authentication.
- Role-based access control (RBAC) for different users (e.g., admin, HOO, SI, principal).
- Middleware (auth.js) likely checks JWT tokens and user roles.

---

10. Deployment Details

Environments:

- **Development**: Local setup with npm run dev (backend) and npm start (frontend).
- **UAT**: Hosted on a local intranet server for testing.
- **Production**: Hosted on a local intranet server for production use.

Hosting:

- **Frontend**: Deployed on a local intranet server (e.g., using a web server like Nginx or Apache to serve the React build).
- **Backend**: Deployed on a local intranet server with Node.js runtime, managed using PM2 for process management.
- **Database**: MongoDB hosted locally on the intranet server or on a dedicated database server within the intranet (alternatively, MongoDB Atlas could be used if cloud access is permitted).
- **File Storage**: Local server storage within the intranet (e.g., a dedicated folder like server/uploads/ for uploaded files such as photographs, invoices, and PDFs).

Build & Deploy Commands:

- **Frontend**:
  The frontend (React) is built and served on the intranet server.

```
cd client

npm run build
```

  - This generates a dist/ folder (since the project uses Vite, as seen in vite.config.js) containing the production-ready static files.
  - Copy the contents of the dist/ folder to the web server's directory (e.g., /var/www/html/asset-faculty-frontend for Nginx or Apache).
  - Configure the web server (e.g., Nginx) to serve the static files:

```
server {

    listen 80;
```

```
        server_name intranet-frontend.example.com;

        root /var/www/html/asset-faculty-frontend;

        index index.html;

        location / {

            try_files $uri $uri/ /index.html;

        }

        location /api {

            proxy_pass http://intranet-backend.example.com:5000;

            proxy_set_header Host $host;

            proxy_set_header X-Real-IP $remote_addr;

        }

    }
```

   - Restart the web server (e.g., sudo systemctl restart nginx).

- **Backend**:
  The backend (Node.js/Express) is deployed on the intranet server and managed using PM2.

```
cd server

npm install

pm2 start server.js --name asset-faculty-backend
```

   - Ensure the backend server is running on a specific port (e.g., 5000, as specified in your .env file).
   - Configure the intranet server to allow access to this port, potentially behind a reverse proxy (e.g., Nginx) if needed:

```
server {

    listen 80;

    server_name intranet-backend.example.com;
```

```
        location / {

            proxy_pass http://localhost:5000;

            proxy_set_header Host $host;

            proxy_set_header X-Real-IP $remote_addr;

        }

    }
```

  - Restart the proxy server if applicable (e.g., sudo systemctl restart nginx).

Additional Notes:

- **MongoDB Setup**: Ensure MongoDB is installed and running on the intranet server (e.g., mongod --dbpath /path/to/db). Update the MONGO_URI in the backend .env file to point to the local MongoDB instance (e.g., MONGO_URI=mongodb://localhost:27017/asset_faculty_db).
- **File Storage**: Uploaded files (e.g., faculty photographs, asset invoices) are stored locally on the intranet server in the server/uploads/ directory. Ensure the directory has appropriate read/write permissions (e.g., chmod -R 755 uploads).
- **Network Access**: The intranet server should be accessible within the organization's network (e.g., via http://intranet-frontend.example.com for the frontend and http://intranet-backend.example.com:5000 for the backend).
- **Environment Variables**: Copy the .env file to the intranet server and update variables as needed (e.g., PORT, MONGO_URI, UPLOAD_PATH).
- **PM2 Management**: Use PM2 commands to manage the backend process:
  - Check status: pm2 list
  - Restart: pm2 restart asset-faculty-backend
  - Logs: pm2 logs asset-faculty-backend

---

11. Testing

Unit Testing:

- **Frontend**: Jest and React Testing Library for component testing.
- **Backend**: Mocha/Chai or Jest for testing controllers and middleware.
- *Note*: Testing setup not provided in code; assumed to be implemented.

API Testing:

- Postman collection for manual API testing.

- Example endpoints to test:
    - POST /api/faculty/save (with file upload).
    - GET /api/assets/getAllAssets.
    - POST /api/faculty/filterFaculties.

Sample Postman Collection:

- Create a collection in Postman with requests for all endpoints listed in Section 6.
- Include environment variables for baseUrl (e.g., http://localhost:5000/api).

---

12. Common Issues & Fixes

CORS Problems:

- **Issue**: Frontend unable to access backend APIs due to CORS.
- **Fix**: Ensure CORS middleware is enabled in server.js:

```
const cors = require('cors');

app.use(cors({ origin: 'http://localhost:3000' }));
```

MongoDB Connection Errors:

- **Issue**: MongoServerError: connect ECONNREFUSED.
- **Fix**: Verify MongoDB is running locally or update MONGO_URI to point to MongoDB Atlas.

React Refresh Issues:

- **Issue**: Frontend not updating on code changes.
- **Fix**: Restart npm start or clear browser cache.

Deployment-Related Bugs:

- **Issue**: File uploads fail in production.
- **Fix**: Ensure file storage path (UPLOAD_PATH) is correctly configured and accessible.

---

13. Future Scope / TODOs

Features in Backlog:

- Implement advanced reporting dashboards (e.g., asset usage trends).
- Add bulk upload for faculty and asset data via CSV.

- Integrate email notifications for faculty and asset actions.

Areas for Improvement:

- Optimize database queries for large datasets (e.g., indexing for staffid, assetId).
- Refactor repetitive controller logic (e.g., approval/rejection functions).
- Enhance security with rate limiting and input sanitization.

---

14. Contact Information

Developers:

- Name: Nirmal Kumar M R, Email: 21i339@psgtech.ac.in
- Methilesh K, Email: 21i228@psgtech.ac.in
- Vaishnav Ram K, Email: 21i269@psgtech.ac.in
- Nirmal Kumar M R, Email: 21i339@psgtech.ac.in
- Abilash A, Email: 22i462@psgtech.ac.in
- S.Akash Aravind, Email: 22i202@psgtech.ac.in
- Praveenkumar A V, Email: 22i241@psgtech.ac.in

Collaboration Tools:

- **GitHub**: https://github.com/Akash-Aravind/casfos_db_deployed