

Estructura del Proyecto - API Gestión de Proyectos y Tareas

Estructura de Directorios

```
project-management-api/  
├── src/  
│   ├── config/  
│   │   ├── database.js  
│   │   ├── jwt.js  
│   │   └── multer.js  
│   ├── controllers/  
│   │   ├── auth.controller.js  
│   │   ├── user.controller.js  
│   │   ├── role.controller.js  
│   │   ├── category.controller.js  
│   │   ├── project.controller.js  
│   │   ├── task.controller.js  
│   │   ├── comment.controller.js  
│   │   ├── state.controller.js  
│   │   ├── upload.controller.js  
│   │   └── ai.controller.js  
│   ├── middlewares/  
│   │   ├── auth.middleware.js  
│   │   ├── role.middleware.js  
│   │   ├── validation.middleware.js  
│   │   ├── error.middleware.js  
│   │   └── upload.middleware.js  
│   ├── models/  
│   │   ├── User.model.js  
│   │   ├── Role.model.js  
│   │   ├── Category.model.js  
│   │   ├── Project.model.js  
│   │   ├── Task.model.js  
│   │   ├── Comment.model.js  
│   │   └── State.model.js  
│   ├── routes/  
│   │   ├── auth.routes.js  
│   │   ├── user.routes.js  
│   │   ├── role.routes.js  
│   │   ├── category.routes.js  
│   │   ├── project.routes.js  
│   │   ├── task.routes.js  
│   │   ├── comment.routes.js  
│   │   ├── state.routes.js  
│   │   ├── upload.routes.js  
│   │   └── ai.routes.js  
│   └── services/  
│       ├── auth.service.js  
│       ├── email.service.js  
│       ├── ai.service.js  
│       └── upload.service.js
```

```

├── utils/
│   ├── response.util.js
│   ├── encryption.util.js
│   ├── validation.util.js
│   └── constants.js
├── validators/
│   ├── auth.validator.js
│   ├── user.validator.js
│   ├── project.validator.js
│   ├── task.validator.js
│   └── common.validator.js
├── app.js
├── uploads/
│   ├── avatars/
│   └── documents/
├── tests/
│   ├── auth.test.js
│   ├── users.test.js
│   ├── projects.test.js
│   └── tasks.test.js
├── .env.example
├── .gitignore
├── package.json
├── README.md
└── server.js

```



Tecnologías Base

- **Backend:** Node.js + Express.js
- **Base de Datos:** MongoDB + Mongoose
- **Autenticación:** JWT + bcryptjs
- **Validación:** express-validator
- **Upload:** multer
- **IA:** Integración con Gemini API
- **Email:** nodemailer
- **Testing:** Jest + supertest



Dependencias Principales

```

{
  "dependencies": {
    "express": "^4.18.2",
    "mongoose": "^7.0.0",
    "bcryptjs": "^2.4.3",
    "jsonwebtoken": "^9.0.0",
    "express-validator": "^6.14.0",
    "multer": "^1.4.5",
    "nodemailer": "^6.9.0",
    "openai": "^4.0.0",
    "cors": "^2.8.5",
    "helmet": "^6.0.0",
    "express-rate-limit": "^6.7.0",

```

```
    "dotenv": "^16.0.0"
  },
  "devDependencies": {
    "jest": "^29.0.0",
    "supertest": "^6.3.0",
    "nodemon": "^2.0.20"
  }
}
```

Variables de Entorno (.env)

```
# Server
PORT=3000
NODE_ENV=development

# Database
MONGODB_URI=mongodb://localhost:27017/project management

# JWT
JWT_SECRET=your super secret key
JWT_EXPIRE=7d
JWT_REFRESH_SECRET=your refresh secret
JWT_REFRESH_EXPIRE=30d

# Email
EMAIL_HOST=smtp.gmail.com
EMAIL_PORT=587
EMAIL_USER=your email@gmail.com
EMAIL_PASS=your app password

# AI
OPENAI_API_KEY=your openai api key

# Upload
MAX_FILE_SIZE=5000000
ALLOWED_FILE_TYPES=image/jpeg,image/png,image/gif

# Rate Limiting
RATE_LIMIT_WINDOW=15
RATE_LIMIT_MAX_REQUESTS=100
```