# 1. To-Do List API

- Users
- Tasks

✅ Practice: Simple CRUD

Model Required

## 📋 User Schema Fields Overview

| Method | Endpoint | Description |
|---|---|---|
| `POST` | `/api/auth/register` | **Register a new user** — Validate name, email, and password. Auto-handle trimming, case formatting, and unique checks. |
| `POST` | `/api/auth/login` | **Login user** — Check email and password, return JWT token |
| `GET` | `/api/auth/me` | **Get current user profile** — Protected route, return user info (except password) |
| `PUT` | `/api/auth/update` | **Update user info** — Allow updating name or email |
| `PUT` | `/api/auth/change-password` | **Change password** — Requires current password, new password |
| `DELETE` | `/api/auth/delete` | **Delete account** — Authenticated user deletes their own account |
| | | |
| | | |

| **Field** | **Type** | **Required** | **Default** | **Validation / Options** |
|---|---|---|---|---|
| `name` | `String` | ✅ Yes | – | - Trimmed- Min length: 2 characters- Error: `"Name is required"` |
| `email` | `String` | ✅ Yes | – | - Unique- Lowercase & Trimmed- Regex validation for format- Error messages |
| `password` | `String` | ✅ Yes | – | - Min length: 6 characters- Error: `"Password is required"` |

| | | | | |
|---|---|---|---|---|
| `createdAt` | `Date` | ✅ Yes | Auto | Added by `timestamps: true` |
| `updatedAt` | `Date` | ✅ Yes | Auto | Added by `timestamps: true` |

Task

| Method | Endpoint | Description |
|---|---|---|
| Method | Endpoint | Description |
| POST | /api/tasks | Create a new task |
| GET | /api/tasks | Get all tasks for the logged-in user |
| GET | /api/tasks/:id | Get a single task by its ID |
| PUT | /api/tasks/:id | Update a task |
| DELETE | /api/tasks/:id | Delete a task by ID |
| GET | /api/tasks/dashboardStats | Get overall task statistics for dashboard |
| GET | /api/tasks/completedTasks | Get list of completed tasks for the logged-in user |

| **Field** | **Type** | **Required** | **Default** | **Validation / Notes** |
|---|---|---|---|---|
| `title` | `String` | ✅ Yes | – | Minimum length: 4 charactersErrors: `"title is required"` , `"minimum 4 character"` |
| `description` | `String` | ✅ Yes | – | ⚠️ You missed `type: String` in your code (needs correction) |
| `completed` | `Boolean` | ❌ No | `false` | Default value is `false` |
| `dueDate` | `Date` | ✅ Yes | – | Inconsistency: Marked optional in TS interface (`?`) but required in schema |
| `user` | `mongoose.Schema.Types.ObjectId` | ✅ Yes | – | References the `"User"` model (foreign key relationship) |

🚀

## Future Enhancements for To-Do App

| Feature | Description | New DB Fields / Models | API Endpoints | Frontend Ideas |
|---|---|---|---|---|
| **Leaderboard (Monthly/Yearly)** | Rank users by tasks completed and productivity score | - Use `Task` aggregation or create `TaskStats` model- Fields: `userId`, `month`, `completedTasks`, `score` | `GET /api/leaderboard/monthlyGET /api/leaderboard/yearly` | Table of top 10 users, filters for global/friends |
| **Progress Tracker** | Visualize monthly and yearly progress | New model `ProgressTracker` Fields: `userId`, `month`, `year`, `completedTasks`, `streak`, `completionRate` | `GET /api/progress/monthlyGET /api/progress/yearly` | Line & bar charts, best/worst months |
| **Public Profile Sharing** | Share productivity profile with others via URL | Extend `User` schema:→ `username` (unique)→ `publicProfile` (boolean) | `GET /api/users/:username/public-profile` | Shareable link showing stats, streaks, rank |
| **Gamification (Badges/XPs)** | Reward users for consistency & productivity | Extend `User` schema:→ `xpPoints`, `badges`, `coins`, `streak` | `GET /api/user/rewards` | Badge icons, XP bar, unlockable themes |
| **Goals & Reminders** | Set monthly/weekly task goals and receive reminders | Add to `User` schema:→ `goals.monthly`, `goals.weekly` Use cron/email system for reminders | `POST /api/user/goalsGET /api/user/goals` | Progress ring UI, reminder popups |