Advanced Todo App - Frontend

A modern React-based task management application with user authentication, real-time chat, file uploads, and collaborative features.

Features

- **User Authentication**: Register and login functionality
- **Task Management**: Create, update, delete, and assign tasks
- **Task Filters**: Filter tasks by status, priority, and dates
- **Real-time Chat**: Chat system for task discussions
- **File Uploads**: Upload images and audio files to tasks
- **Responsive Design**: Mobile-friendly interface with Tailwind CSS
- **Role-based Permissions**: Assign tasks with different permission levels

Tech Stack

- **Frontend**: React 19, Vite, React Router
- **Styling**: Tailwind CSS
- **Icons**: Lucide React
- **HTTP Client**: Axios
- **Date Handling**: date-fns
- **Backend**: Express.js (included in server.js)

Prerequisites

- Node.js (v16 or higher)
- npm or yarn

Installation

PROF

1. Clone the repository:

```
git clone <repository-url>
cd advanced-todo-app
```

2. Install dependencies:

```
npm install
```

Running the Application

Development Mode

```
npm run dev
```

Starts the frontend development server on `http://localhost:5173`

Start Backend Server

```
npm run server
```

Starts the backend API server on `http://localhost:4000`

Production Build

```
npm run build
npm run preview
```

Project Structure

```
src/
├── components/
│   ├── auth/          # Authentication components
│   ├── tasks/         # Task-related components
│   ├── chat/          # Chat components
│   └── ui/            # Reusable UI components
├── context/           # React context providers
├── hooks/             # Custom React hooks
├── services/          # API service functions
├── utils/             # Utility functions
└── constants/         # App constants
```

PROF

API Integration

The frontend connects to a backend API running on `http://localhost:4000`. The backend provides:

- User authentication endpoints
- Task CRUD operations
- Chat functionality
- File upload handling

Key Components

- **AuthForm**: User registration and login
- **TaskGrid**: Display tasks in a grid layout

- **TaskCard**: Individual task display
- **TaskForm**: Create/edit task form
- **ChatBox**: Real-time chat interface
- **ImageUpload**: Image upload component
- **AudioRecorder**: Voice recording component

Scripts

- `npm run dev` - Start development server
- `npm run build` - Build for production
- `npm run preview` - Preview production build
- `npm run server` - Start backend server
- `npm run lint` - Run ESLint

Environment Variables

Create a `.env` file in the root directory if you need to configure API endpoints:

```
VITE_API_BASE_URL=http://localhost:4000
```

Contributing

1. Fork the repository
2. Create a feature branch
3. Make your changes
4. Run linting: `npm run lint`
5. Test your changes
6. Submit a pull request

License

PROF

ISC