

eBuddy Technical Test

About

This is a monorepo built with Turborepo that contains both frontend and backend applications. The project is structured as follows:

Apps

- **frontend-repo**: A Next.js application with Material UI, Redux Toolkit, and Firebase integration
- **backend-repo**: An Express.js API server with Firebase Admin SDK integration

Packages

- **@repo/dto**: Shared Data Transfer Objects between frontend and backend
- **@repo/eslint-config**: Shared ESLint configurations
- **@repo/typescript-config**: Shared TypeScript configurations

How To Run Locally

Prerequisites

- Node.js (version specified in `.nvmrc`)
- pnpm package manager

Setup

1. Clone the repository

```
git clone git@github.com:RofiSyahrul/ebuddy-technical-test.git
cd ebuddy-technical-test
```

2. Install dependencies

```
pnpm install
```

3. Set up environment variables (see Environment Variables section below)

4. Run the development server

```
pnpm dev
```

This will start both the frontend and backend applications in development mode.

Build for Production

```
pnpm build
```

Environment Variables and Secret Files

Backend Application

Create a `.env` file in the `apps/backend-repo` directory with the following variables:

```
CORS_ALLOWED_ORIGIN=localhost  
PORT=8080
```

Additionally, you'll need:

Firestore Admin SDK Service Account:

- The backend uses Firestore Admin SDK with application default credentials
- Place your Firestore service account key file in a location accessible by the application (`./apps/backend-repo/service-account.json`)

Frontend Application

Create a `.env.local` file in the `apps/frontend-repo` directory with the following variables:

```
NEXT_PUBLIC_API_URL=http://localhost:8080  
NEXT_PUBLIC_FIREBASE_API_KEY=your_firebase_api_key  
NEXT_PUBLIC_FIREBASE_APP_ID=your_firebase_app_id  
NEXT_PUBLIC_GCLOUD_PROJECT=your_gcloud_project_id
```

Development Workflow

- **Running specific apps:** You can use the `--filter` flag to run commands for specific apps:

```
pnpm dev --filter frontend-repo  
pnpm dev --filter backend-repo
```

- **Linting:** Run ESLint across all projects:

```
pnpm lint
```

Project Structure

```
.
├── apps/
│   ├── backend-repo/    # Express.js backend
│   └── frontend-repo/  # Next.js frontend
├── packages/
│   ├── dto/             # Shared data transfer objects
│   ├── eslint-config/   # Shared ESLint configuration
│   └── typescript-config/ # Shared TypeScript configuration
└── ...
```

Technologies Used

- **Frontend:** Next.js, React, Material UI, Redux Toolkit, Firebase
- **Backend:** Express.js, Firebase Admin SDK
- **Build Tools:** Turborepo, esbuild
- **Languages:** TypeScript