

Whizlabs Inventory Management System

-Santhoshkumar J V

Table of Contents

1. Project Setup
2. System Architecture
3. Api Explanation
4. Instructions to use
5. Result

1. Project Setup

The Simple Inventory Management system is a web application that allows users to manage inventory items. It provides a single collection for managing items with the following fields:

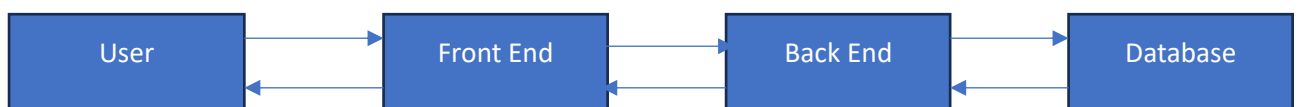
- id (Unique Identifier)
- name (Name of the Item)
- description (Details about the Item)
- category (Category to which the Item belongs)
- price (Price of the Item)
- quantity (Number of Items in Stock)

The application exposes RESTful APIs for CRUD operations and includes a React-based frontend for user interaction.

2. System Architecture

The system follows the Client – Server architecture with RESTful Api design. The key components are:

- Frontend (React JS)
- Backend (Express)
- Database (MongoDB)



3. Api Explanation

API	Description	
Method: Get Path: /items	Purpose	To get all the items from the database
	Request Body	None
	Expected Response	All the items from the database
Method: Get Path: /items/:id	Purpose	To get the particular item with the given :id from the database
	Request Body	None
	Expected Response	If(id present in database) -> Details of the particular Item Else -> 404 Error
Method: Post Path: /items	Purpose	To add new item to the database
	Request Body	<pre>{ itemName : String, description : String, category : String, Quantity : Number, Price: Decimal }</pre>
	Expected Response	If(Validated) -> Adds item to the database with statusCode 201 Else -> Request Error with 400 statusCode
Method: Put Path: /items/:id	Purpose	To edit the existing item to the database
	Request Body	Request Body: { <pre>itemName : String, description : String, category : String, Quantity : Number, Price: Decimal }</pre>

	Expected Response	If(id present && Validated) -> Edit item to the database with statusCode 200 If(id not present) -> 404 error with Item not found response Else -> Request Error with 400 statusCode
Method: Delete Path: /items/:id	Purpose	To delete the existing item from the database
	Request Body	None
	Expected Response	If(id present) -> Deletes the item from the database with statusCode 200 Else -> Item Not found Error with 404 statusCode

4. Instructions to use

Prerequisites

Before setting up the project, ensure the following software is installed:

- Node.js (v20 or later)
- npm (for dependency management)
- Git (for cloning the repository)

Git Clone

Download or clone the following repository and store it in your system.

Github Link: <https://github.com/Santho257/whizlabs-assessment>

Database Setup

Install MongoDB

Start Mongo Server

No manual setup is required for database since backend creates necessary collections.

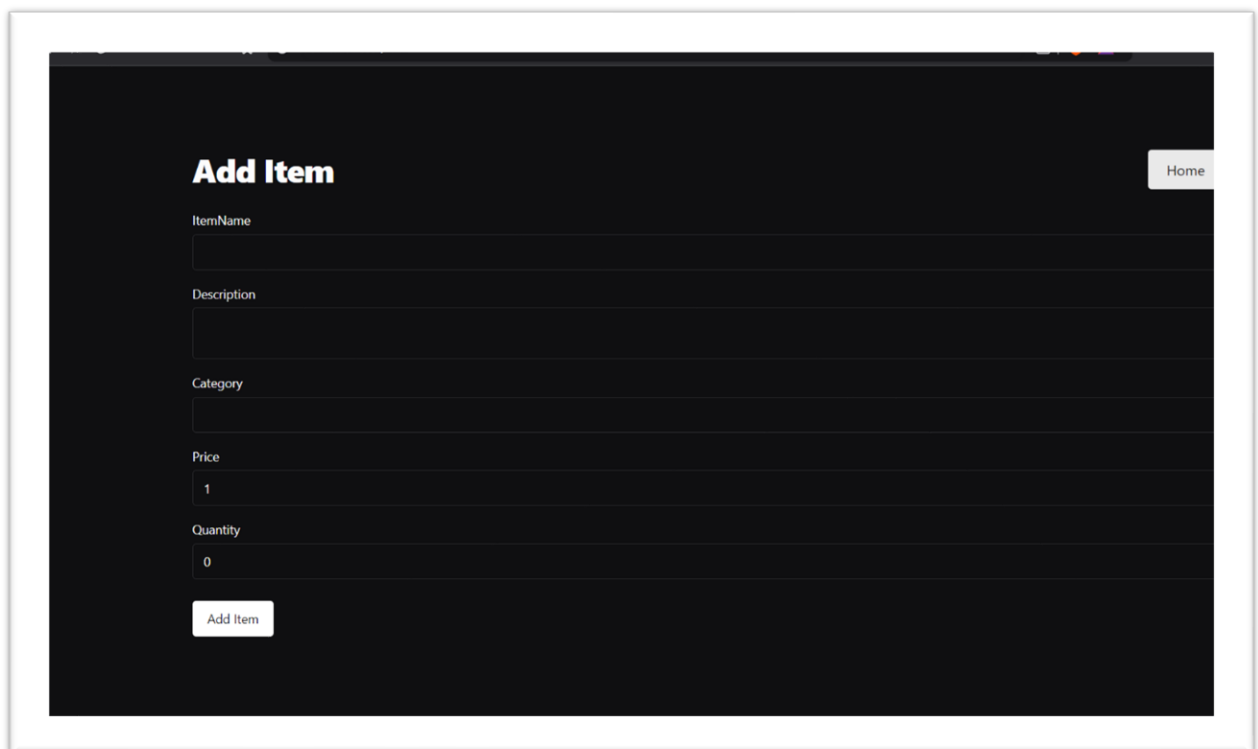
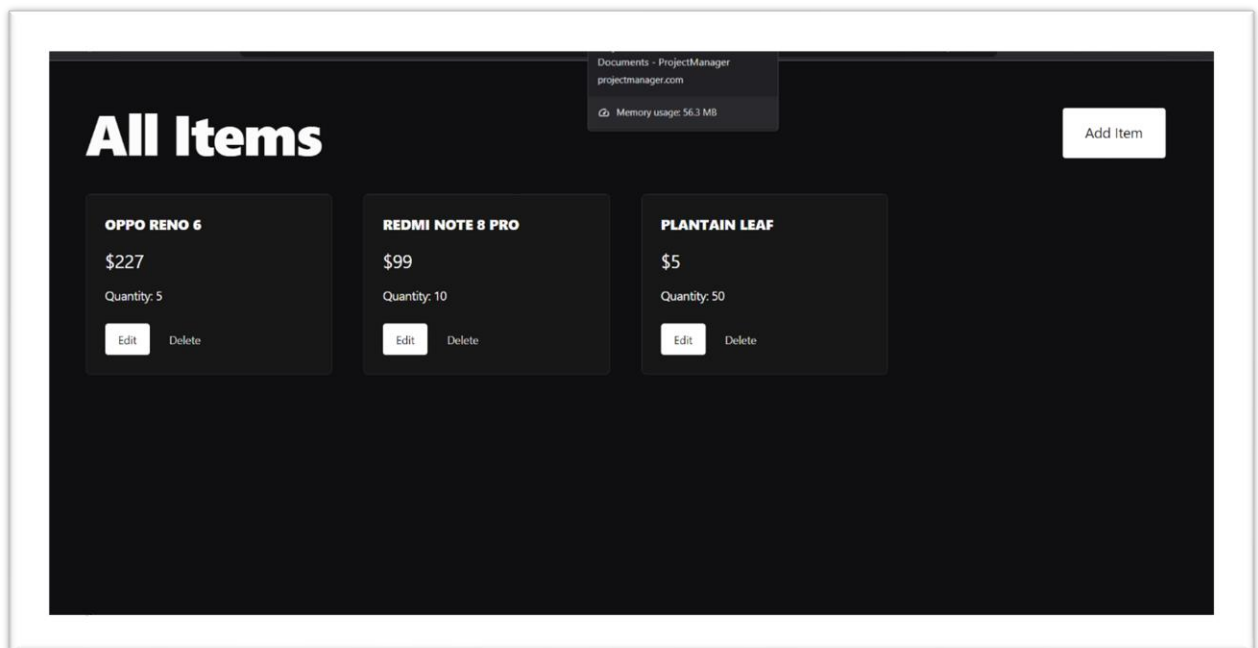
Backend Setup

- Open the project in terminal.
- Change current dir to backend
cd ./whizlabs-inventory-back-end
- Install dependencies
npm install
- Copy env files
cp .env.dev .env
- Change the .env as needed, setup MongoDB URL to your local database
- Run the back end
npm run dev

Front End Setup

- Open the project in terminal.
- Change current dir to frontend
cd ./whizlabs-inventory-front-end
- Install The Dependencies
npm install
- Change the constants.js BACKEND values as needed
- Install Chakra Snippets
npx @chakra-ui/cli add snippet
- Run the front end
npm run dev

5. Result



localhost:5173/add

Add Item

Home

ItemName

Description

Category

Price

Quantity

Add Item

localhost:5173/edit/6731799073b44a4e485b3e0

Edit Item

Home

ItemName

REDMI NOTE 8 PRO

Description

A mobile phone

Category

ELECTRONICS

Price

99

Quantity

10

Edit Item

