# **Inventory Management System API Project Documentation**

## 1. Project Overview

* **Project Name**: Develop a Basic Backend API for the Inventory Management System
* **Current Phase/Status**: Development Completed by Shubham Umbarkar

## 2. Handover Summary

* **Handover Date**:04/10/2024
* **Resource(s) Handing Over**: Shubham Umbarkar
* **Resource(s) Taking Over**:

## 3. Key Project Information

**Project Goals**: Create a backend API to support basic CRUD operations for an Inventory Management System with authentication and caching capabilities.

**Objectives**:

* + Implement basic inventory operations including Create, Read, Update, Delete (CRUD).
  + Ensure secure access using JWT-based authentication.
  + Use Redis for caching frequently accessed items.
  + Provide logging and error handling for monitoring.
  + API includes unit tests to verify functionality.

**Current Status**: All core features completed; JWT authentication integrated.

## 4. Requirements

* **Django** == 4.2.4
* **djangorestframework** == 3.15.0
* **djangorestframework-simplejwt** == 5.1.0
* **django-cors-headers** == 4.2.0
* **django-redis** == 5.3.0
* **psycopg2-binary** == 2.9.6
* **Create Virtual Environment**: python -m venv (name)

## 5. Development Details

* **Codebase Overview**:
  + **Repository Location**: [Insert repository URL]
* **Branch Information**: Key branch is master.

### Key Components and Features:

**User Registration**

* + **Objective**: Allow users to register with username, password, and email.
  + **Endpoint**: /register/
  + **Method**: POST
  + **Logic**:
    - Create a new user instance.
    - Save the user with a hashed password.
    - Return success and JWT tokens.
  + **Dependencies**: Django REST Framework, Simple JWT.

**User Login**

* + **Objective**: Authenticate users and return a JWT token.
  + **Endpoint**: /login/
  + **Method**: POST
  + **Logic**:
    - Verify the user credentials.
    - Return JWT tokens upon successful authentication.
  + **Dependencies**: Django REST Framework, Simple JWT.

**Create Inventory Item**

* + **Objective**: Allow authenticated users to add inventory items.
  + **Endpoint**: /items/
  + **Method**: POST
  + **Logic**:
    - Ensure the user is authenticated.
    - Save the item to the database.
    - Return the item details.
  + **Dependencies**: Django, Django REST Framework.

**Read Inventory Item**

* + **Objective**: Retrieve a specific inventory item by its ID.
  + **Endpoint**: /items/<item\_id>/
  + **Method**: GET
  + **Logic**:
    - Fetch the item by ID from the database.
    - If not found, return a 404 response.
  + **Dependencies**: Django, Django REST Framework.

**Update Inventory Item**

* + **Objective**: Allow users to update existing inventory items.
  + **Endpoint**: /items/<item\_id>/update/
  + **Method**: PUT
  + **Logic**:
    - Ensure the item exists.
    - Update the fields and save to the database.
    - Return updated item details.
  + **Dependencies**: Django, Django REST Framework.

**Delete Inventory Item**

* + **Objective**: Delete an inventory item by ID.
  + **Endpoint**: /items/<item\_id>/delete/
  + **Method**: DELETE
  + **Logic**:
    - Find the item by ID.
    - If found, delete the item from the database.
    - Return a success message.
  + **Dependencies**: Django, Django REST Framework.

**Read All Inventory Items**

* + **Objective**: Retrieve a list of all inventory items.
  + **Endpoint**: /items/
  + **Method**: GET
  + **Logic**:
    - Fetch all items from the database.
    - Return a list of item details.
  + **Dependencies**: Django, Django REST Framework.

**Caching with Redis**

* + **Objective**: Cache frequently accessed items using Redis.
  + **Logic**:
    - Cache the item on the first access and return from cache for subsequent requests.
    - Redis configured to cache items for 5 minutes.

## 6. Documentation and Resources

* **Repository**:https://github.com/Shubumbarkar/DRF\_assignment
* **API Documentation**: https://documenter.getpostman.com/view/37753310/2sAXxMeY7Q

## 7. Sign-Off

* **Prepared By**:  
  Name: Shubham Umbarkar  
  Position: SDE  
  Signature: Shubham Umbarkar