**Warehouse Management System – Documentation**

**1. Project Introduction**

**Project Name:** Warehouse Management Application

**Purpose:**

* Track warehouse stocks and work orders
* Ensure two-way data synchronization with Zoho Creator
* Manage data with a local database and user interface

**Target Users:** Warehouse managers, technicians

**2. System Architecture**

**Components:**

1. **Zoho Creator** → Data entry via forms and reports
2. **Zoho Creator API** → Data exchange with PyQt5 application and Flask listener
3. **Flask Webhook Listener (Python)** → Receives JSON payload and processes data
4. **Ngrok** → Provides external access to the local server
5. **SQLite** → Local database; parts, stock movements, and work orders
6. **PyQt5 GUI** → User interface; Stock Movements, Inventory Count, Work Orders, Synchronization

**3. User Guide**

**Tabs and Functions**

1. **Stock Movements:** Stock in/out, Deletion
2. **Inventory Count:** Current stock counting, Difference, and Update
3. **Work Orders:** Add, View, Delete work orders
4. **Synchronization:** Send all data to Creator, Receive updates from Creator (Webhook + API)

**Quick Usage Notes:**

* Quantity and code entry must be correct
* Each transaction can be automatically sent to Creator

**4. Data Flow / API Operation**

* **Webhook:** Captures changes in Creator
* **Ngrok:** Provides external access to the localhost server
* **Flask Listener:** Receives JSON payload, saves to SQLite, sends back to Creator via API
* **send\_to\_creator function:** Adds or updates data in Creator
* **SQLite and PyQt GUI:** Creator ID and stock status are updated

**5. Database Design**

**parts**

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| id | INTEGER PK | Auto-increment ID |
| code | TEXT | Part code |
| description | TEXT | Part description |
| quantity | INTEGER | Stock quantity |
| shelf | TEXT | Shelf / location |
| creator\_id | TEXT | Zoho Creator ID |

**stock\_movements**

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| id | INTEGER PK | Auto-increment ID |
| part\_id | INTEGER FK | Reference to parts table |
| movement\_type | TEXT | In / Out / Inventory Count |
| quantity | INTEGER | Movement quantity |
| shelf | TEXT | Shelf / location |
| creator\_id | TEXT | Zoho Creator ID |
| date | TIMESTAMP | Movement date |

**work\_orders**

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| id | INTEGER PK | Auto-increment ID |
| records | TEXT | Maintenance / repair log |
| required\_parts | TEXT | Required parts |
| status | TEXT | Status information |
| date | TIMESTAMP | Creation date |
| creator\_id | TEXT | Zoho Creator ID |

**6. Installation and Running**

1. Python 3.11 or higher must be installed
2. Required libraries: PyQt5, Flask, Requests, JSON
3. Zoho API token and client details (CLIENT\_ID, CLIENT\_SECRET, REFRESH\_TOKEN) must be configured
4. Initialize SQLite database: init\_db()
5. Start Flask server: run\_server()
6. Open localhost tunnel with Ngrok: ngrok http 5000
7. Run PyQt5 GUI: python main.py