Automated Palletizer Cell Control

Background

You are tasked with planning and programming the control of a Robotic Cell for palletizing application at a warehouse.
A robotic arm operates inside a secure cell, stacking boxes on pallets. The cell has doors to allow human access, a stack-light to indicate system status, and a touchscreen HMI for workers.
Communication with the Warehouse Management System (WMS) is done via API calls.
You will simulate this cell in software - no real robot is required.

Tasks

1. API Call Handler

Warehouse Management System (WMS) sends palletizing requests to the Robotic Cell.

Request format:
HTTP POST http://<ip>:8080/palletize

```
{
  "palletId": 456,
  "boxCount": 10
}
```

The Robotic Cell responds after processing the request:

HTTP POST http://<ip>:8081/confirmPalletize

```
{
```

"palletId": 456,
"palletizeSuccessful": true,
"errorMessage": null,
"lastBoxBarcode": 98765
}

If the door is open or emergency button pressed, respond with palletizeSuccessful: false.

You may simulate errors if desired.

2. Scanner ROS 2 Node

Simulate a barcode scanner that publishes a 5-digit random barcode.

Provide a service to return the latest scanned barcode.

3. Door Handle ROS 2 Node

Publishes boolean:

- true: door closed
- false: door open

Provide a service to toggle door state.

If door is open, no palletizing should occur.

4. Emergency Button ROS 2 Node

Publishes boolean:

- true: emergency button pressed
- false: normal

Automated Palletizer Cell Control

Provide two services:

- Press emergency button
- Reset emergency button

When pressed, palletizing must stop.

5. Stack-Light ROS 2 Node

Publishes int state:

- 0: operational (door closed, no emergency, ready to palletize)
- 1: paused (door open)
- -1: emergency

6. HMI

Implement a simple real-time HMI (HTML/React, Qt, Tkinter... your choice).

Display:

- Current palletize request (palletId, boxCount)
- Response status
- Emergency button state
- Door state
- Stack-light indicator:
    - 0: green
    - 1: yellow
    - -1: red

Requirements

Automated Palletizer Cell Control

Requirements

- Submit full project as ZIP
- Include Git version control + README (dependencies, install, run instructions)
- Provide screen-recorded video demo (no face, no private data)
- Mention any open-source code used
- Use ROS2 Humble if possible
- Docker support is optional
- If unclear decisions arise, document them in README