

AUT.840
ASSIGNMENT 01: ORCHESTRATION
OF INDUSTRIAL EQUIPMENT

INDUSTRIAL INFORMATICS

Presented by: Stepan Zalud | Nadun Ranasinghe | Ana Carvajal

151062243 | 150906193 | 151021213

CONTENTS

- ✓ Objectives
- ✓ Requirements
- ✓ Loop-Based Method. Event-Based Method
- ✓ Class Diagram
- ✓ Implementation
- ✓ Program description
- ✓ Improvements
- ✓ Demonstration

OBJECTIVES

- **OOP for solving an orchestration task**
- **Use of Python**
- **Use of Web-Services**
- **Control of industrial processes**
- **Problem resolution**

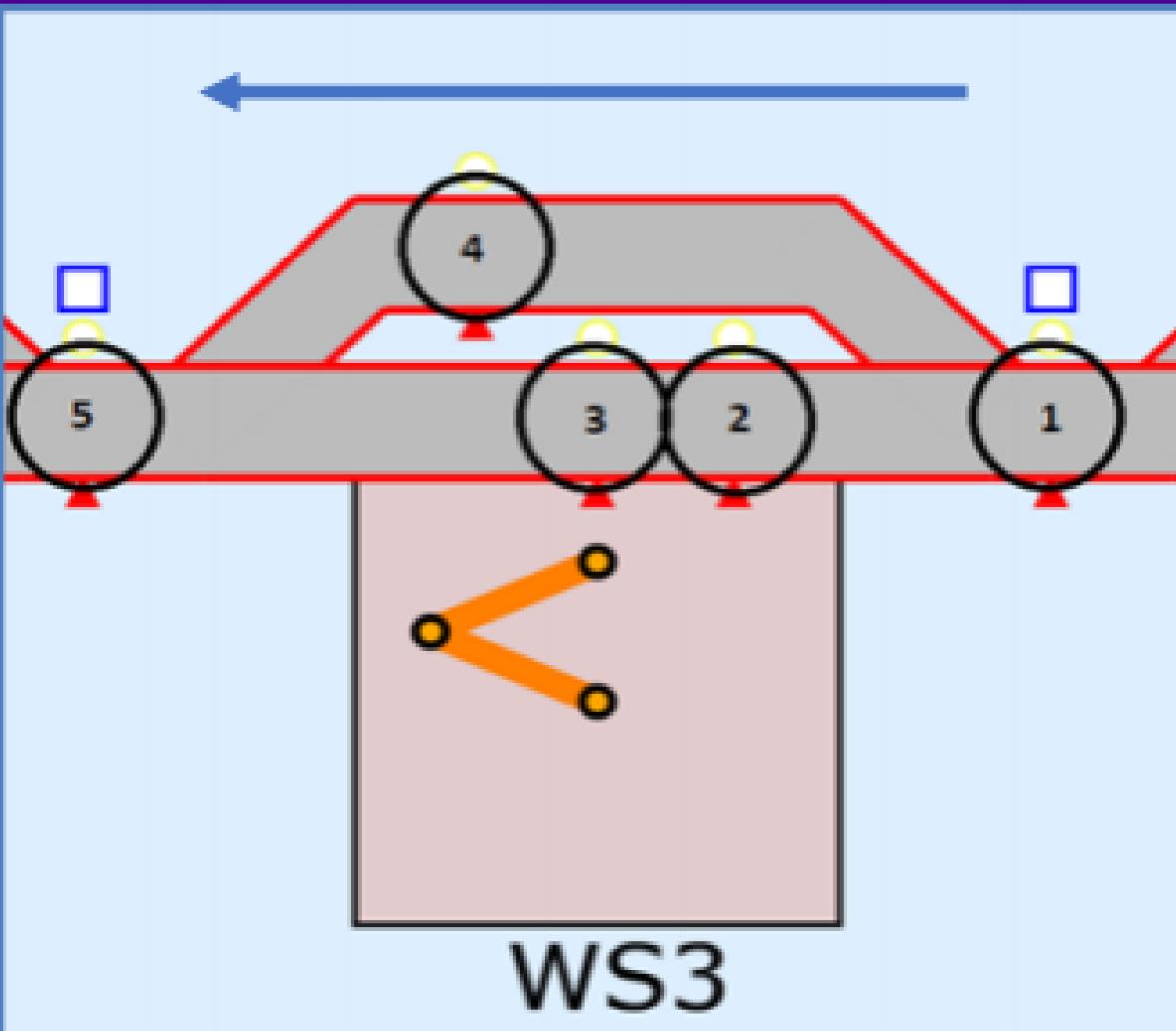
REQUIREMENTS

Z1 -->Z3

- The pallet moves to Z3
- The robot executes the required order
- Pen color won't change.
- Draw recipies

Z1 -->Z2

- Waiting zone
- Z3 is running an order



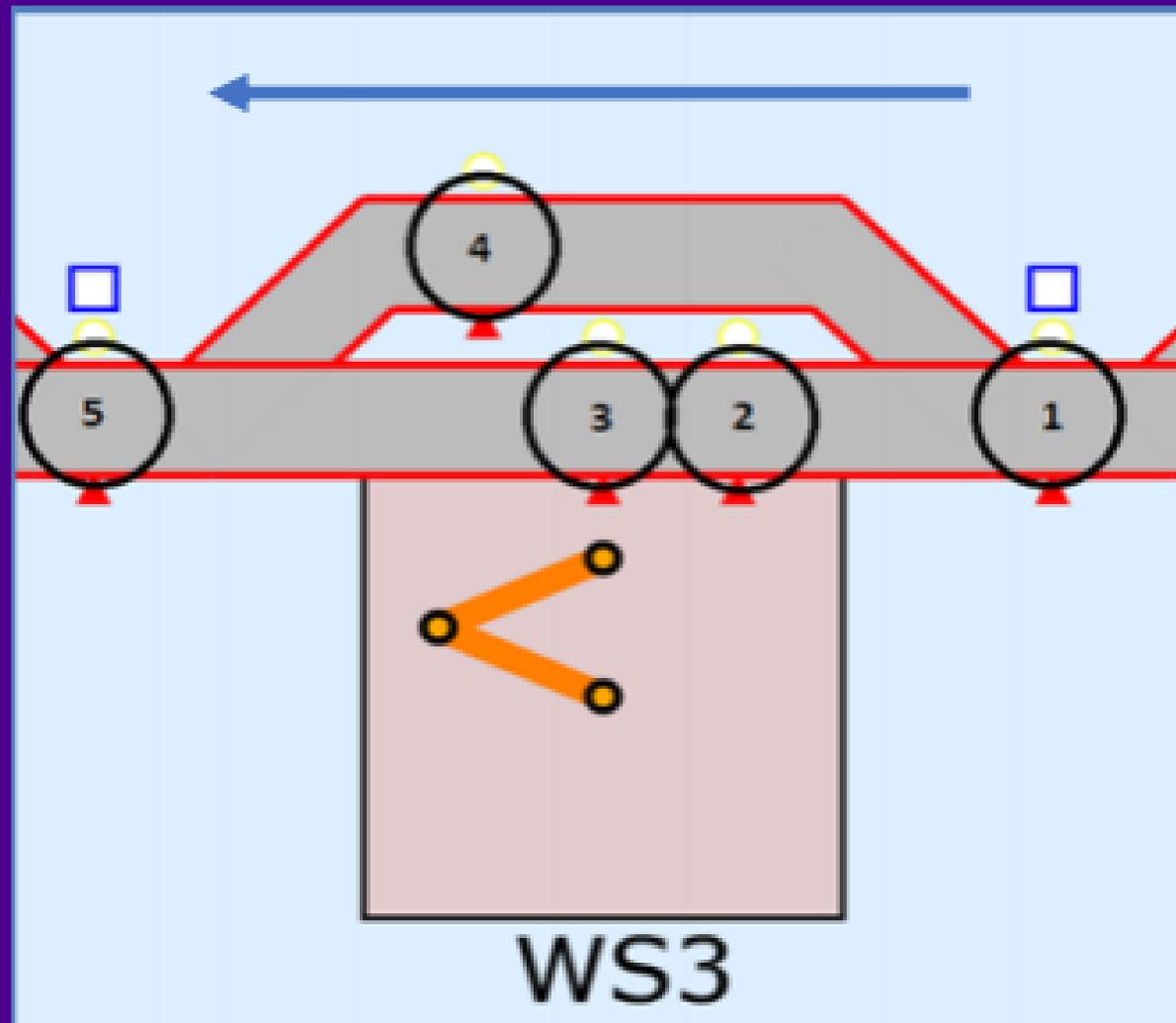
REQUIREMENTS

Z1 -->Z4

- Order discarded
- Z2 and Z3 in use

Z4 -->Z5

- When the robot is working
- Avoid collisions



LOOP-BASED SYSTEM

- Checks the state of the conveyor at each zone and acts accordingly.
- Simpler code.
- No server is required.
- No Flask.
- Not able to use subscribed notifications.
- Not able to identify the state of the robot.

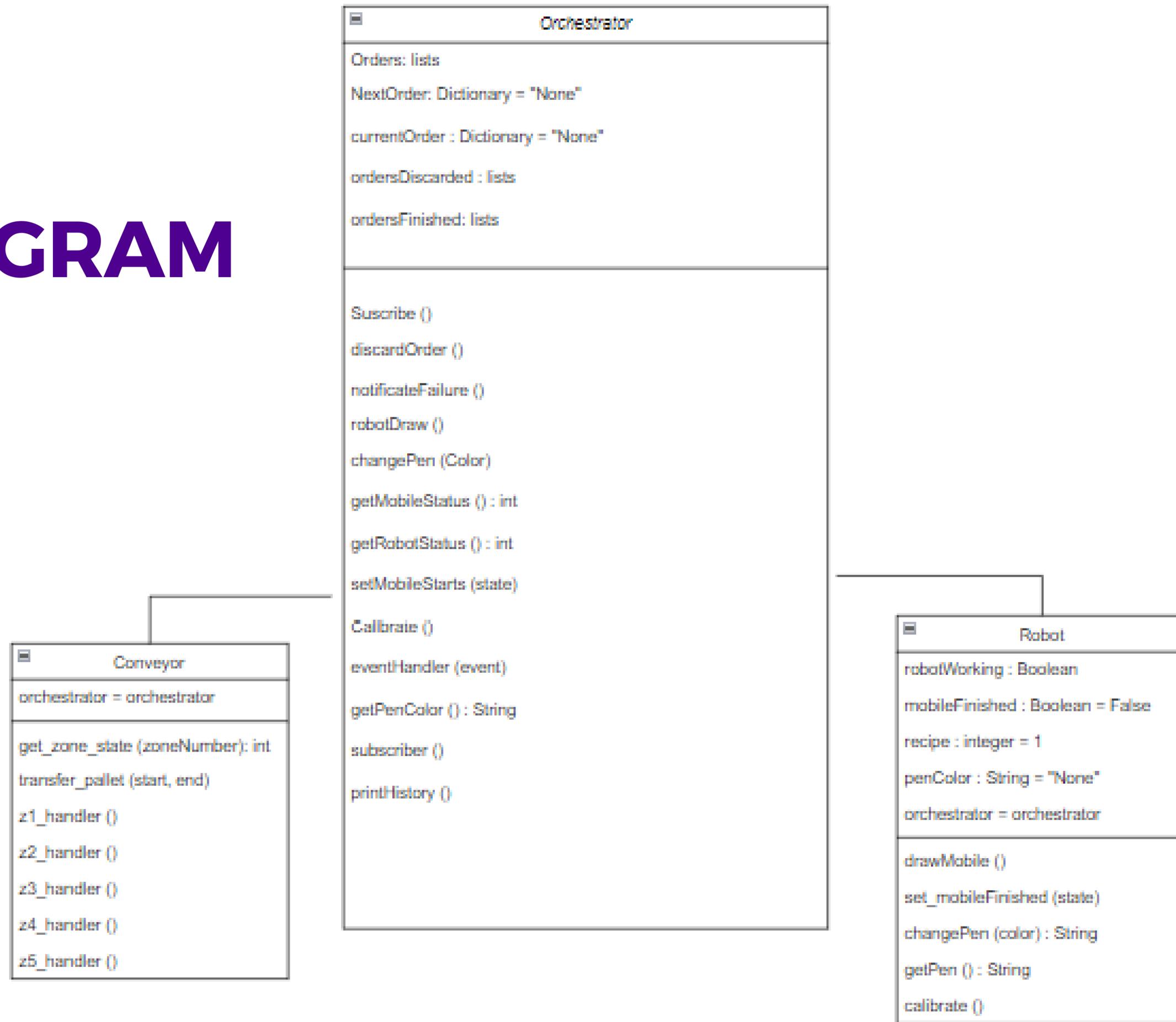
EVENT-BASED SYSTEM

- Works according to the occurring of an event.
- High complexity.
- Works more efficiently.
- Uses Flask server.
- Get robot state notifications.
- Miss messages received from the workstation.

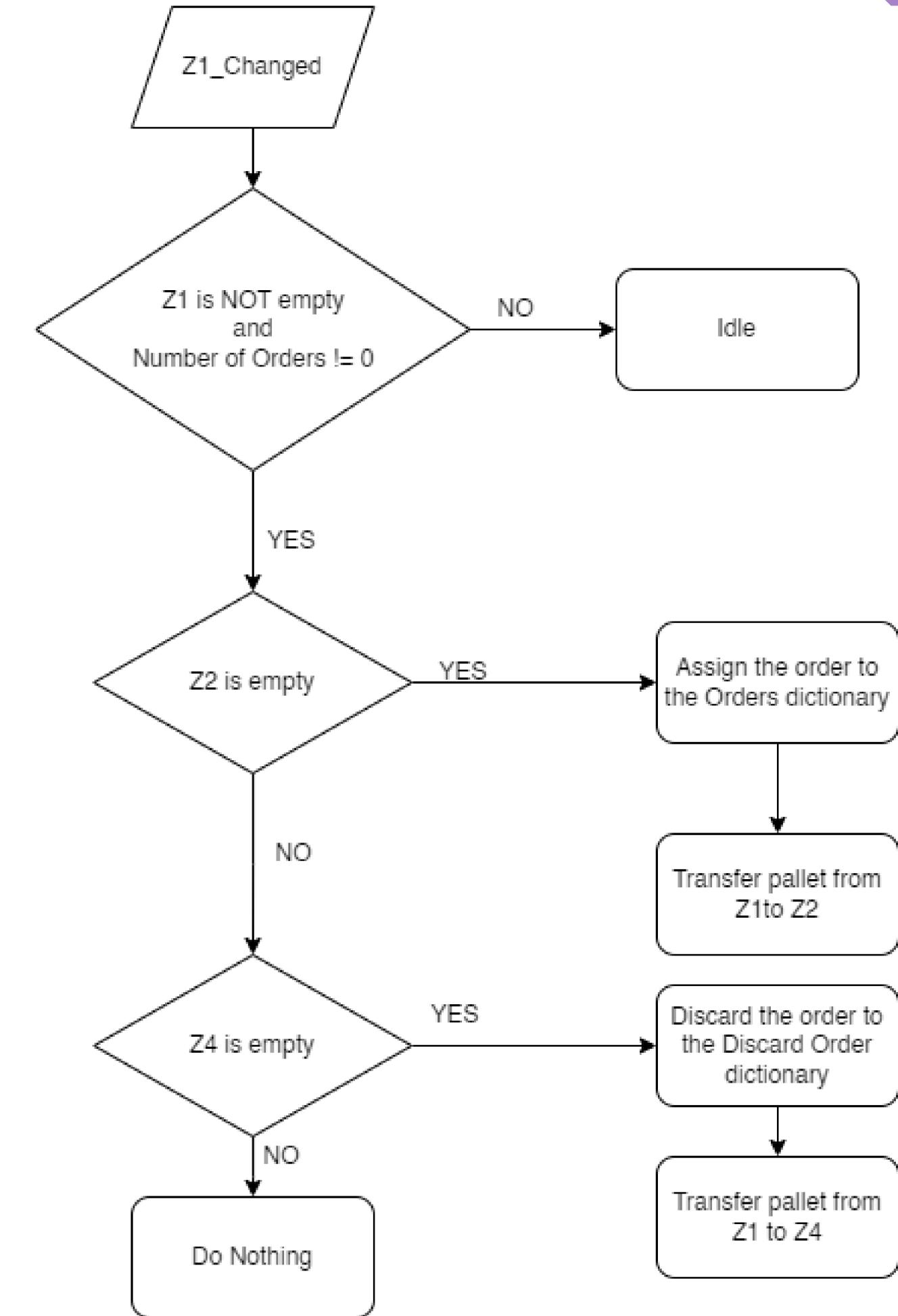
IMPLEMENTATION

- **Event-Based System**
- **Calibration at the initialization of the system.**
- **Implements the system automatically according to the notifications and orders received.**
- **Use of time delays for the efficient functioning of the drawing process.**

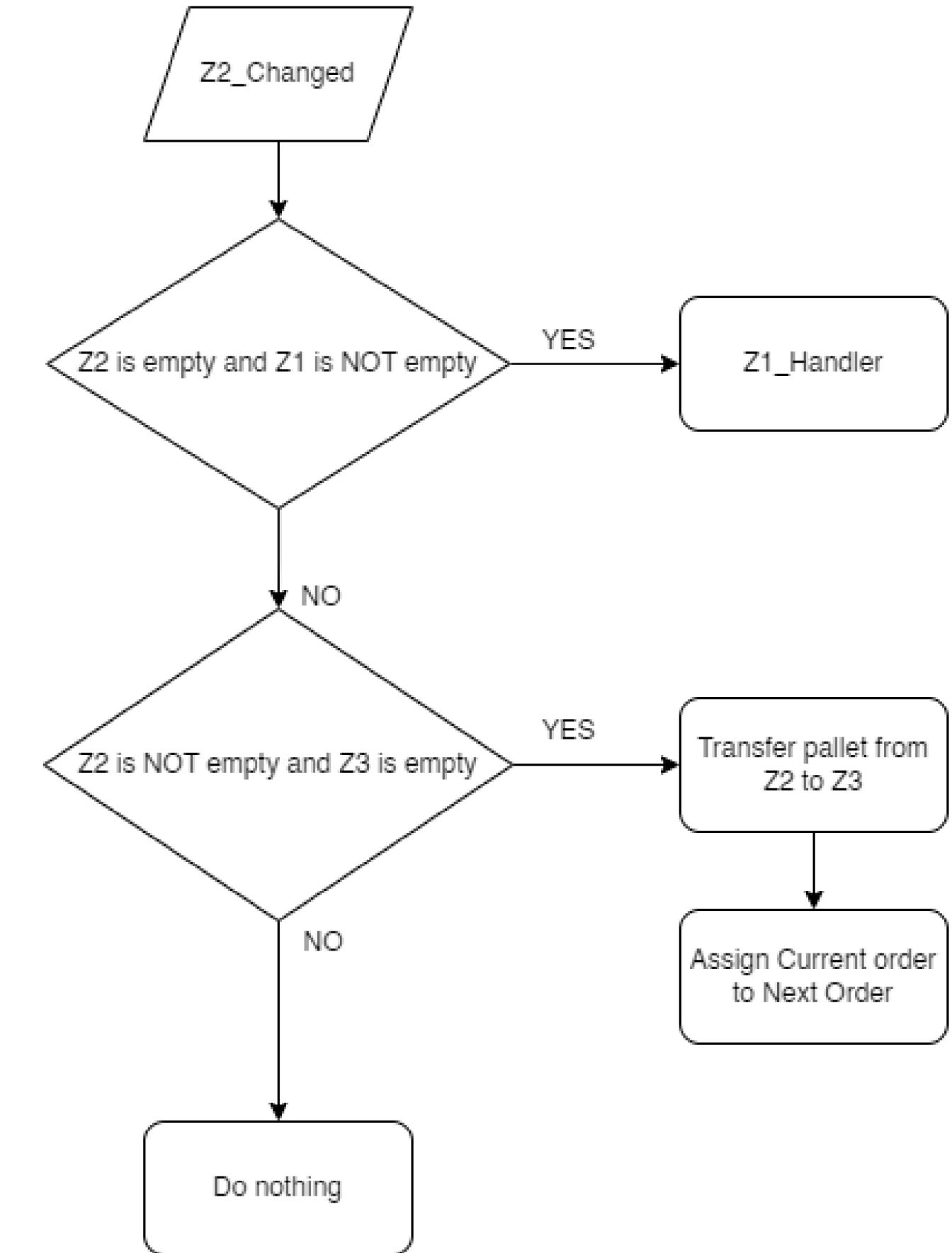
CLASS DIAGRAM



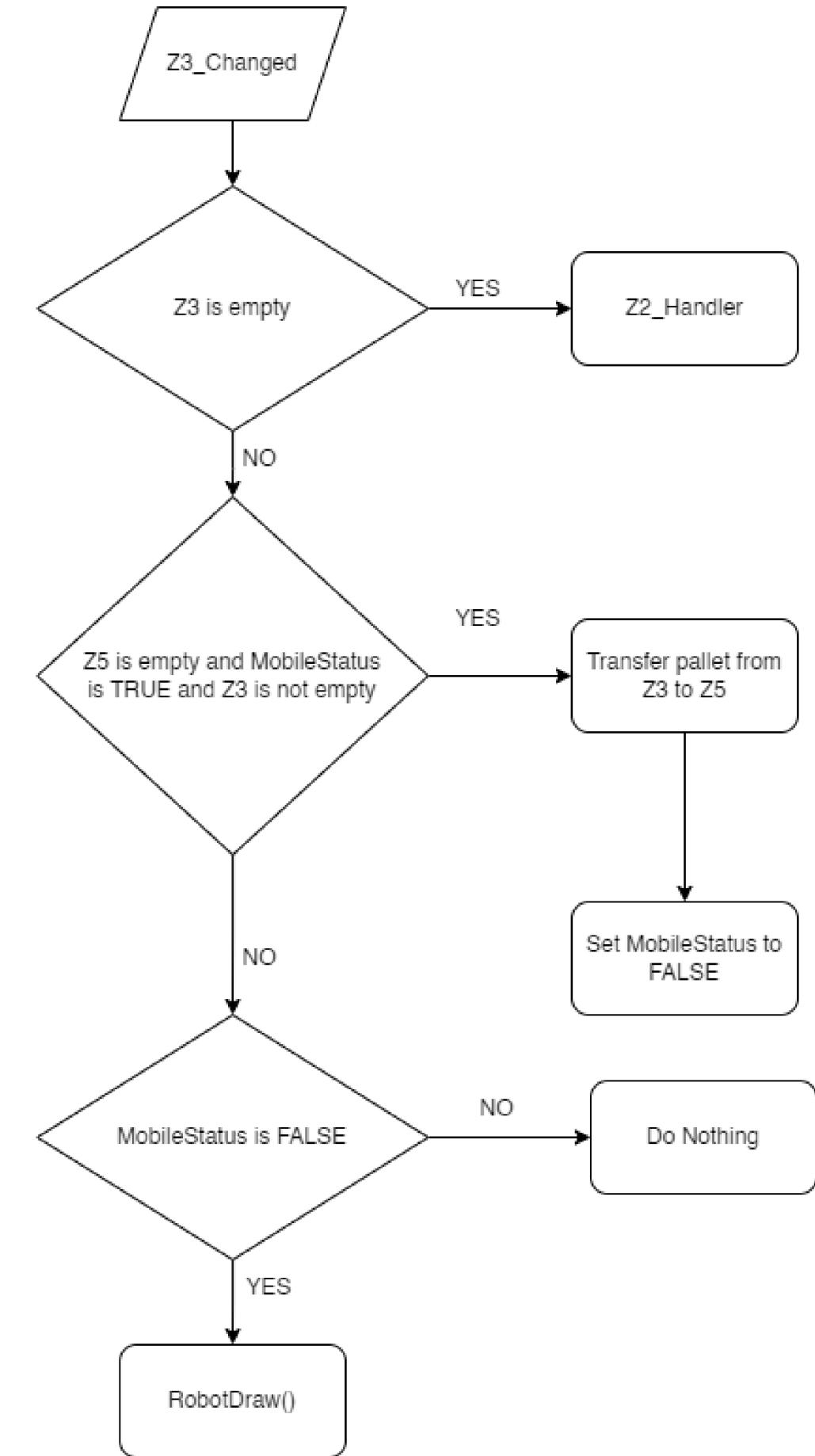
EVENT-BASED SYSTEM ZONE 1



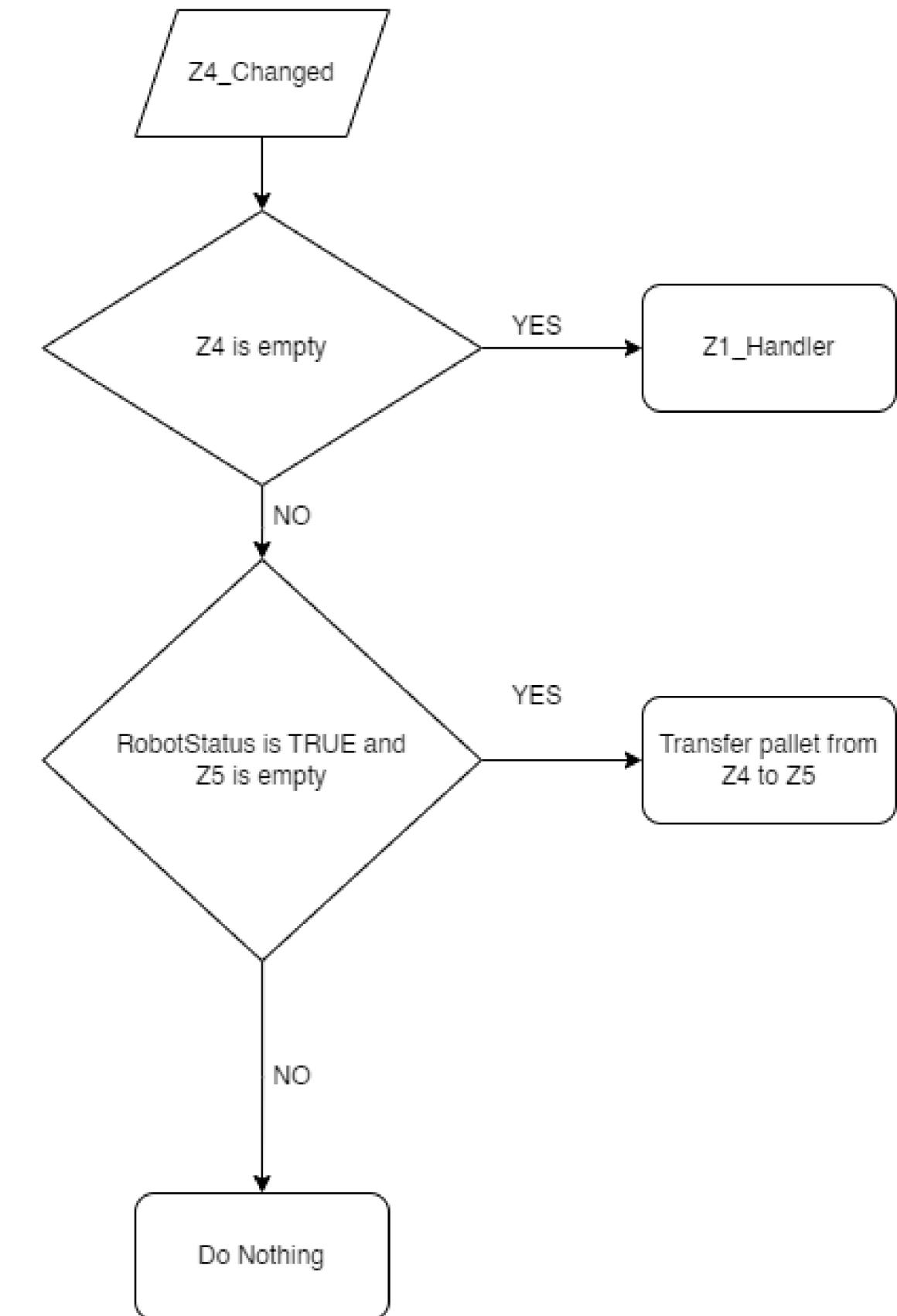
EVENT-BASED SYSTEM ZONE 2



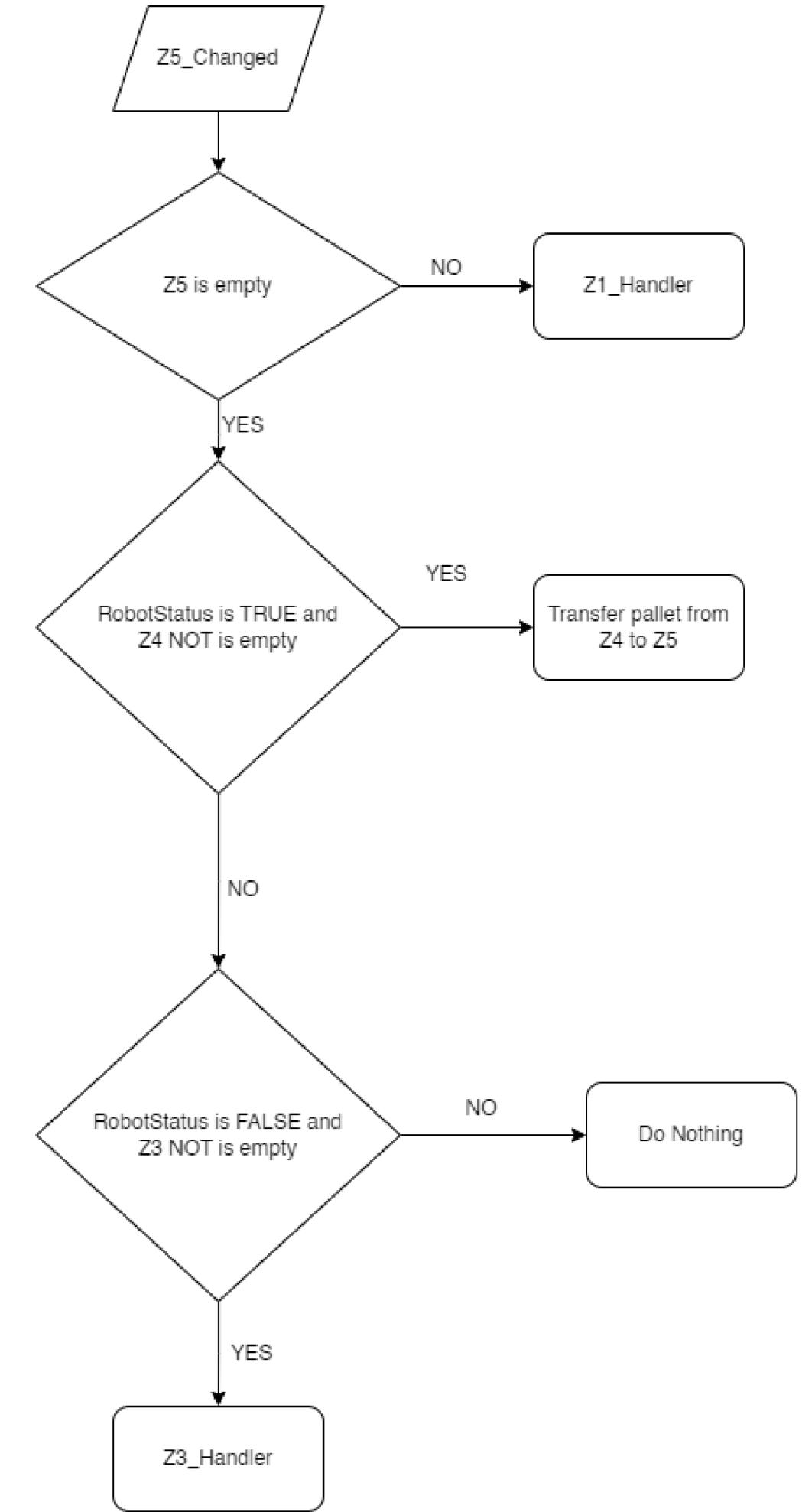
EVENT-BASED SYSTEM ZONE 3



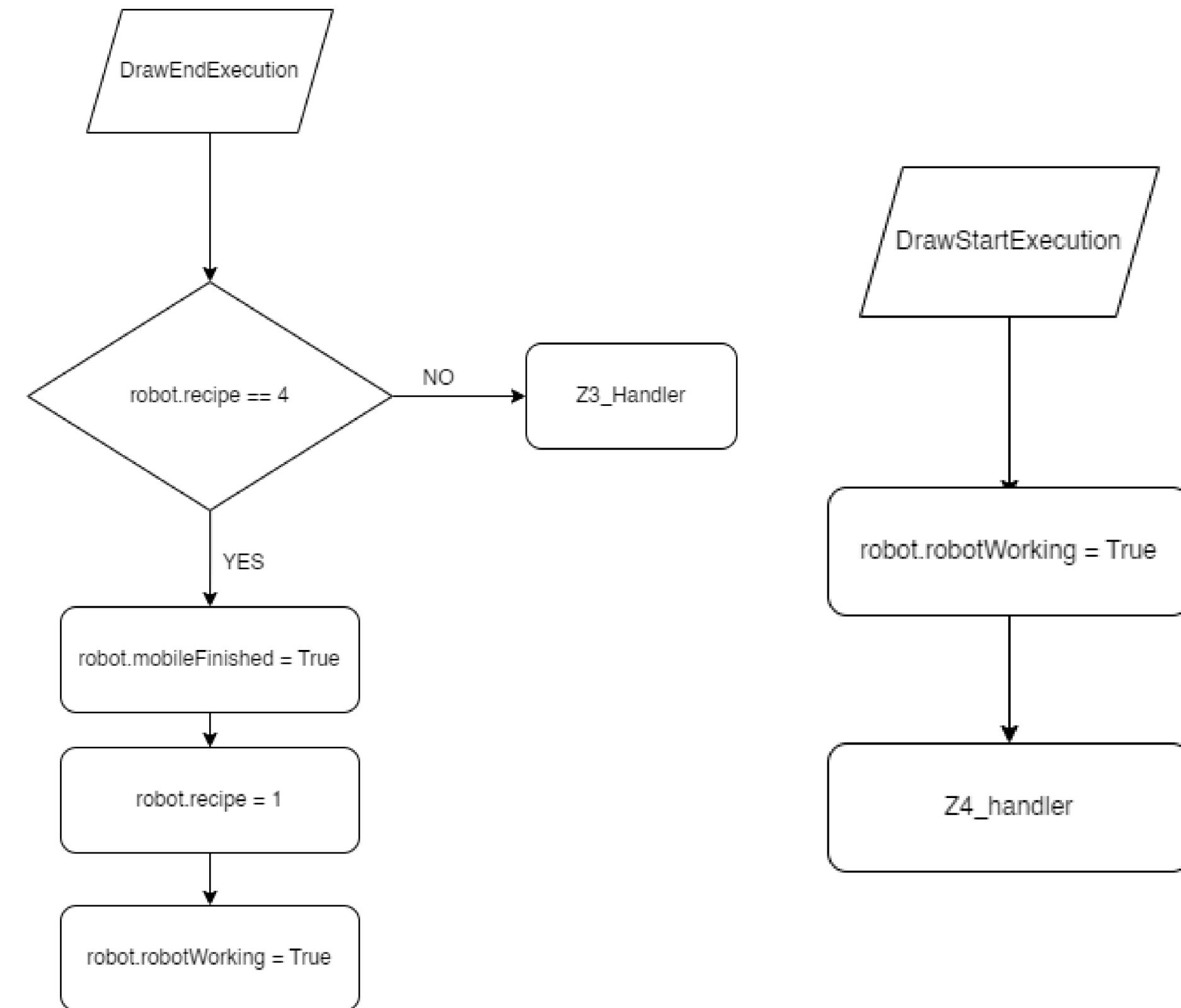
EVENT-BASED SYSTEM ZONE 4



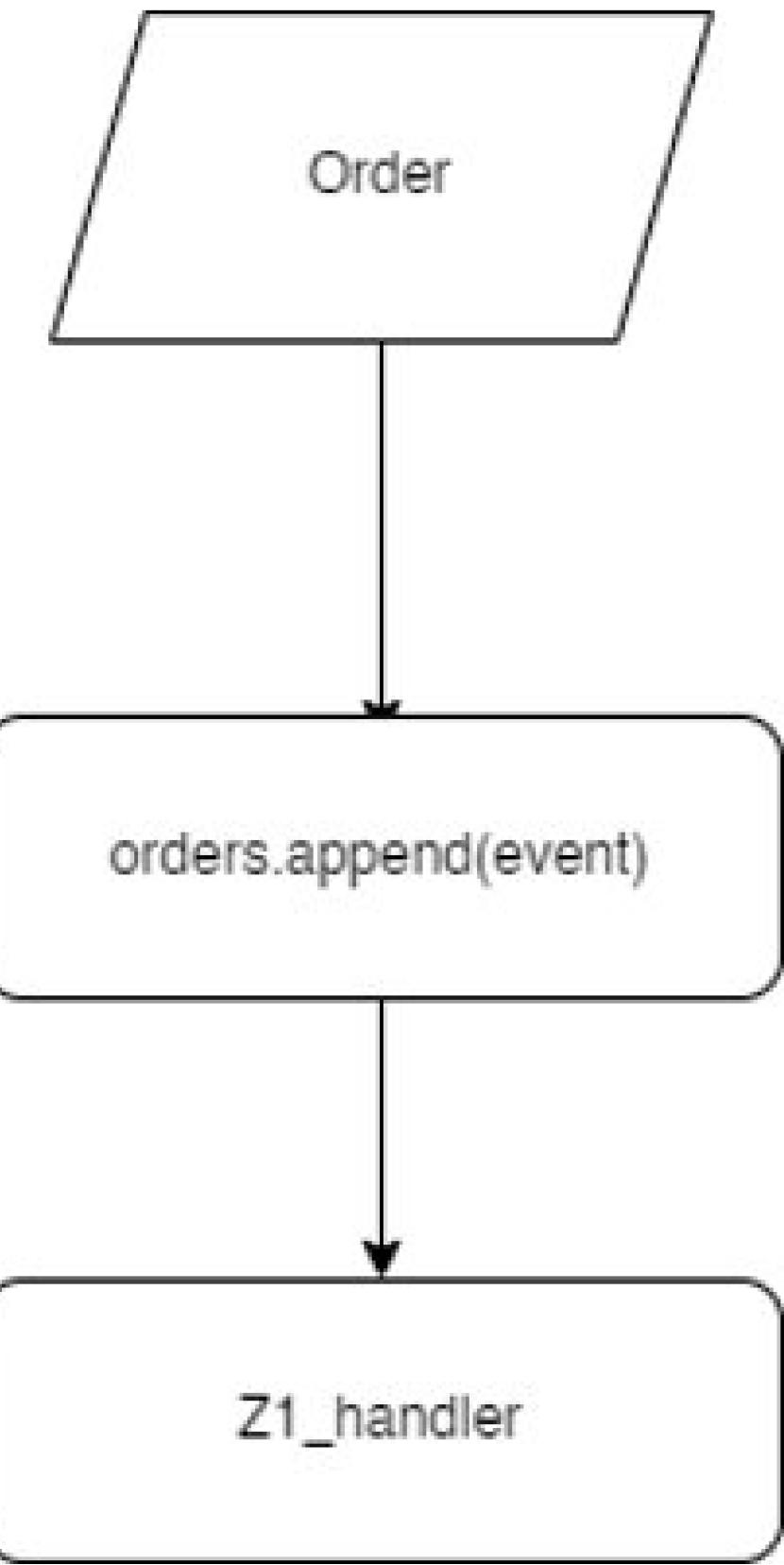
EVENT-BASED SYSTEM ZONE 5



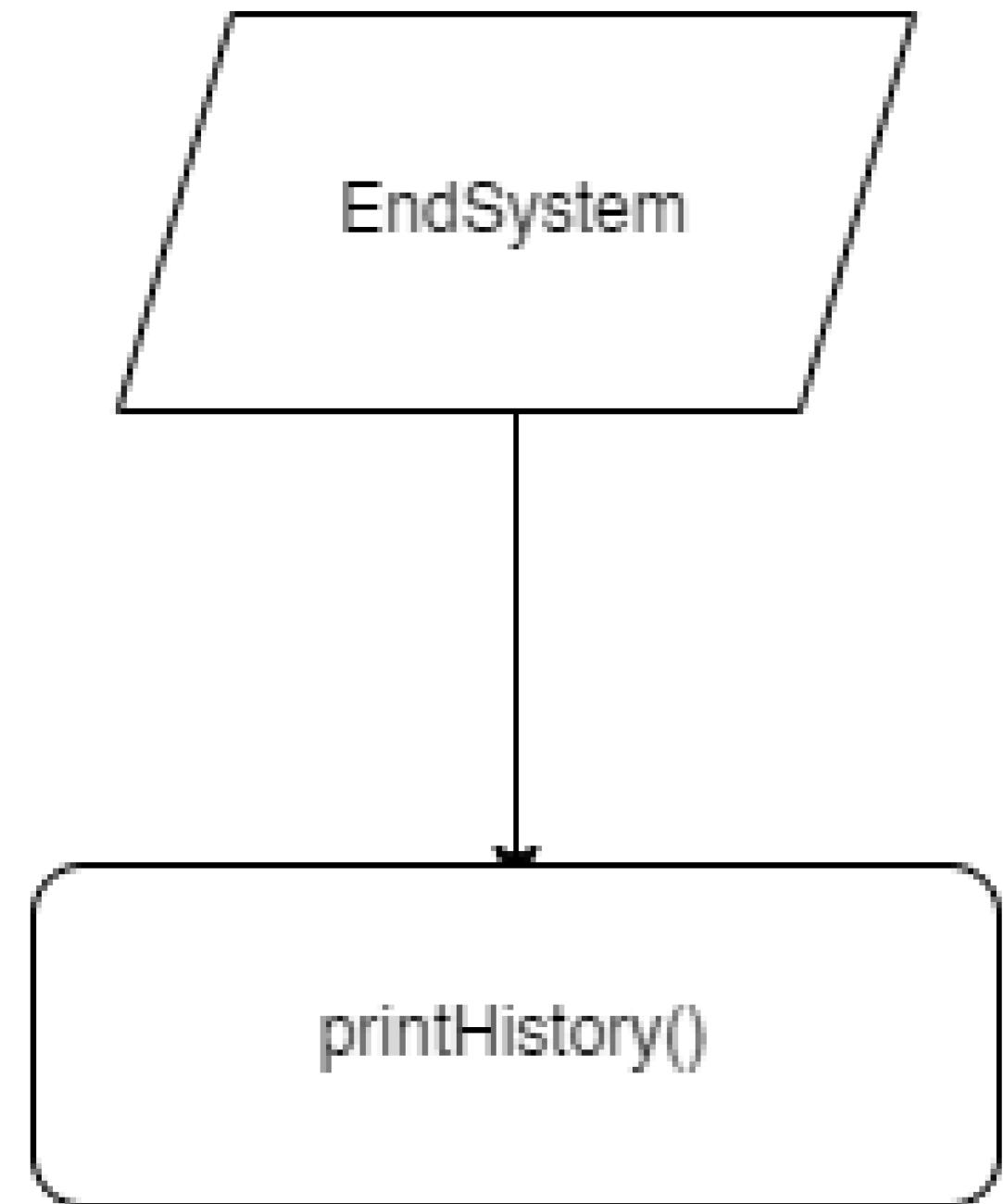
EVENT-BASED SYSTEM DRAW EXECUTION



EVENT-BASED SYSTEM ORDER HANDLER



EVENT-BASED SYSTEM SYSTEM CLOSE

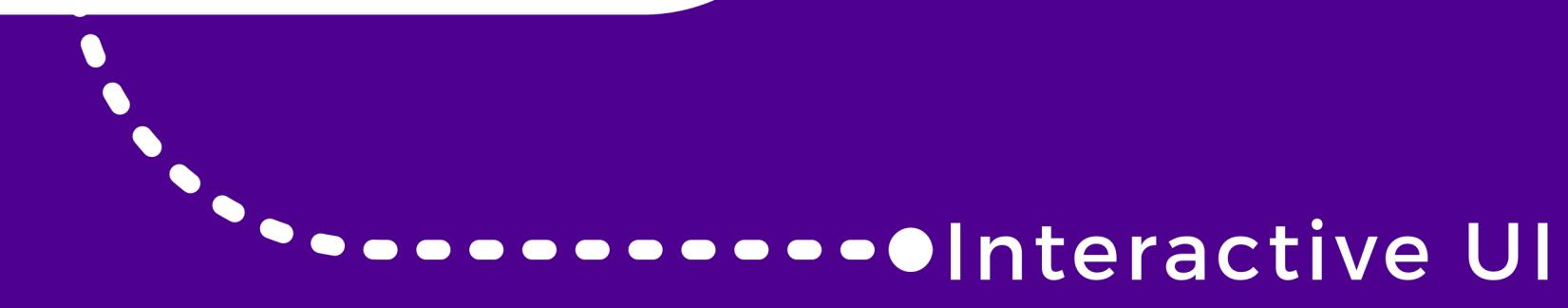


Full report of orders



penChange functions
(draw each part in different colors)

Serial Production with multiple workstations
(One Part from each Workstation)



Interactive UI

DEMONSTRATION

