• For each component list their interfaces and describe them according to the component-based software architecture paradigm (i.e., stateless/statefull, data/service, strongly-typed/loosely-typed, etc).

**Components details**

**1. PointOfSale (POS)**

* **Provided Interface:**
  + /orders (ROS Topic): Publishes new customer orders.
    - **Type:** Data Interface (publishes order data).
    - **State:** Stateless (each published order is independent).
    - **Typing:** Loosely-typed (uses std\_msgs/String but contains structured data within the string: "Table : X, dish : Y, ..."). Requires parsing by the subscriber.

**2. OrchestrationManager**

* **Required Interfaces:**
  + /orders (ROS Topic): Subscribes to receive new orders from POS. (See POS description above).
  + /availability (ROS Topic): Subscribes to receive robot availability status.
    - **Type:** Data Interface (receives robot state data).
    - **State:** Stateless (each message represents current status).
    - **Typing:** Loosely-typed (uses std\_msgs/String like "TIAGo X : available").
  + /position (ROS Topic): Subscribes to receive robot positions.
    - **Type:** Data Interface (receives robot state data).
    - **State:** Stateless (each message is current position).
    - **Typing:** Strongly-typed (uses geometry\_msgs/Point).
* **Provided Interfaces:**
  + /order\_TIAGo (ROS Topic): Publishes specific task assignments to robots.
    - **Type:** Data Interface (publishes command data). Could also be seen as triggering a service, but uses a topic.
    - **State:** Stateless (each message is a new assignment).
    - **Typing:** Loosely-typed (uses std\_msgs/String like "TIAGo n°X, table : Y, dish : Z").
  + /error\_messages (ROS Topic): Publishes error messages for staff/logging.
    - **Type:** Data Interface.
    - **State:** Stateless (each message is an independent error report).
    - **Typing:** Strongly-typed (uses std\_msgs/String, but typically carries unstructured text).

**3. TIAGo (Main Robot)**

* **Required Interfaces:**
  + /order\_TIAGo (ROS Topic): Subscribes to receive task assignments from OrchestrationManager. (See OrchestrationManager description).
* **Provided Interfaces:**
  + /availability (ROS Topic): Publishes its current status ("available" or "occupied").
  + /position (ROS Topic): Publishes its current position (x, y) and ID (in z). (See OrchestrationManager description).
  + /orders (ROS Topic): Publishes requests for table clearing (uses the same topic as POS, potentially causing confusion, sends geometry\_msgs/Point in the code, which differs from POS's String format - this seems like a potential bug/design issue).
    - **Type:** Data Interface.
    - **State:** Stateless.
    - **Typing:** Strongly-typed (uses geometry\_msgs/Point, mismatch with POS string).
* **Internal Interfaces :**
  + Calls methods on PerceptionSystem, OrderVerificationSystem, NavigationSystem, ManipulationSystem, ReasoningSystem. **Service Interfaces**, **Stateful**, and **Strongly-Typed**.

**4. PerceptionSystem**

* **Provided Interfaces (Conceptual - often via method calls from TIAGo):**
  + perception\_for\_navigation(): Returns obstacle detection boolean.
    - **Type:** Service Interface (provides a computation result on demand).
    - **State:** Stateless (result depends only on current sensor simulation).
    - **Typing:** Strongly-typed (returns Python boolean).
  + verification\_of\_grasping\_and\_placement(operation): Returns success boolean.
    - **Type:** Service Interface.
    - **State:** Stateless (result based on probability).
    - **Typing:** Strongly-typed (takes string, returns boolean).

**5. NavigationSystem**

* **Provided Interfaces:**
  + /map (ROS Topic): Publishes the nav\_msgs/OccupancyGrid.
    - **Type:** Data Interface.
    - **State:** Stateful (map represents accumulated knowledge).
    - **Typing:** Strongly-typed (nav\_msgs/OccupancyGrid).
  + /robot\_pose (ROS Topic): Publishes the estimated geometry\_msgs/PoseStamped.
    - **Type:** Data Interface.
    - **State:** Stateful (pose depends on history).
    - **Typing:** Strongly-typed (geometry\_msgs/PoseStamped).
  + /trajectory (ROS Topic): Publishes the planned nav\_msgs/Path.
    - **Type:** Data Interface.
    - **State:** Stateless (represents the current plan for the current goal).
    - **Typing:** Strongly-typed (nav\_msgs/Path).
* **Required Interfaces:**
  + /goal (ROS Topic): Subscribes to receive navigation goals (geometry\_msgs/PoseStamped).
    - **Type:** Data Interface (receives command data).
    - **State:** Stateless (each goal is independent).
    - **Typing:** Strongly-typed (geometry\_msgs/PoseStamped).
  + Sensor Topics (e.g., /imu\_data, /left\_encoder, /right\_encoder, /object\_detection, /manipulation\_status - based on SLAM subscribers in code): These are Data Interfaces, Stateless, and likely Strongly-typed (Float32, String etc.) depending on sensor nodes.
* **Provided Interfaces (to TIAGo via methods):**
  + Maps\_to(): Service Interface, triggers stateful navigation behavior, Strongly-typed.
  + update(): Service Interface, Stateless trigger, Strongly-typed.
  + stop(): Service Interface, Stateless trigger, Strongly-typed.

**6. ManipulationSystem**

* **Provided Interfaces:**
  + /force\_data (ROS Topic): Publishes std\_msgs/Float32. Data, Stateless, Strongly-typed.
  + /motors\_feedback (ROS Topic): Publishes std\_msgs/String (contains structured data). Data, Stateful (reflects motor state), Loosely-typed.
  + /safety\_status (ROS Topic): Publishes std\_msgs/String. Data, Stateful, Strongly-typed (enum-like string).
  + /manipulation\_status (ROS Topic): Publishes std\_msgs/String. Data, Stateful, Strongly-typed (enum-like string).
* **Required Interfaces:**
  + /gripper\_commands (ROS Topic): Subscribes std\_msgs/Float32. Data/Command, Stateless, Strongly-typed.
  + /grasp\_commands (ROS Topic): Subscribes std\_msgs/String. Data/Command, Stateless, Strongly-typed.
  + /joint\_commands (ROS Topic): Subscribes std\_msgs/String. Data/Command, Stateless, Loosely-typed (string encodes positions).
  + /trajectory\_commands (ROS Topic): Subscribes std\_msgs/String. Data/Command, Stateless, Loosely-typed (string encodes waypoints).
  + /target\_dish\_position (ROS Topic): Subscribes geometry\_msgs/Point. Data, Stateless, Strongly-typed.
  + /force\_data (ROS Topic): Internal subscription. (See above).
  + /safety\_status (ROS Topic): Internal subscription. (See above).
  + /motors\_feedback (ROS Topic): Internal subscription. (See above).
  + /perception\_data (ROS Topic): Subscribes std\_msgs/String (generic placeholder). Data, Stateless, Loosely-typed.
* **Provided Interfaces (to TIAGo via methods):**
  + execute\_manipulation():
  + Service Interface,
  + triggers stateful manipulation,
  + Strongly-typed.

**7. ReasoningSystem**

* **Provided Interfaces:**
  + /target\_dish\_position (ROS Topic): Publishes geometry\_msgs/Point. Data, Stateless (result of a specific request), Strongly-typed.
* **Required Interfaces:** None explicitly shown via ROS (depends on how perception\_data is passed to reason\_about\_placement).
* **Provided Interfaces (to TIAGo via methods):**
  + reason\_about\_placement():
  + Service Interface,
  + Stateless (performs calculation based on input),
  + Strongly-typed.

**8. OrderVerificationSystem**

* **Required Interfaces:**
  + Access to TIAGo instance attributes (perception\_system, id, target\_table). (Internal interface/dependency).
  + Interaction with SpeechInterface (via method call verify\_delivery\_client). Service, Stateful, Strongly-typed.
  + Interaction with PerceptionSystem (via method call like verify\_item\_served). Service, Stateless, Strongly-typed.
* **Provided Interfaces (to TIAGo via methods):**
  + verify\_served\_order(): Service, Stateless, Strongly-typed.
  + verify\_delivery\_client(): Service, Stateful, Strongly-typed.
* **(If Error Notification Added Back): Provided Interface:**
  + /error\_messages (ROS Topic): Publishes std\_msgs/String. Data, Stateless, Strongly-typed .

**9. SpeechInterface**

* **Provided Interfaces (to OrderVerificationSystem):**
  + verify\_delivery\_client(): Service Interface, Stateful, Strongly-typed.