Design Proposal for the Pick and Place Solution

1. Cobot Selection

After reviewing the detailed specifications of the UR10e, Doosan M1013, Doosan H2515, UR3e, and CRX 10iA, here are the key points of consideration for the selection:

Payload Capacity:

UR10e: 12.5 kg (sufficient for handling both the cardboard box (7 kg) and the wooden cube (0.5 kg)). ✅

Doosan M1013: 10 kg (sufficient for handling both the cardboard box and the wooden cube). ✅

Doosan H2515: 25 kg (exceeds the required payload but more than necessary). ✖

UR3e: 3 kg (insufficient for the application due to low payload). ✖

CRX 10iA: 10 kg (sufficient for handling both the cardboard box and the wooden cube). ✅

Reach:

UR10e: 1300 mm (allowing it to cover the necessary workspace between Table A and B). ✅

Doosan M1013: 1300 mm (ideal for the workspace between Table A and Table B). ✅

Doosan H2515: 1500 mm (ideal for the workspace between Table A and Table B). ✅

UR3e: 500 mm (not enough to cover the necessary workspace between Table A and Table B). ✖

CRX 10iA: 1249 mm (ideal for the workspace between Table A and Table B). ✅

While all three robots (UR10e, CRX10iA, and Doosan M1013) meet the client's basic requirements for payload and reach, still the UR10e has a higher payload capacity compared to the CRX10iA and Doosan M1013, which provides more flexibility and future-proofing for handling heavier or additional tools. As well, both the UR10e and Doosan M1013 have a reach of 1300 mm, while the CRX10iA has a slightly shorter reach at 1249 mm. The additional reach of the UR10e can provide better coverage and flexibility in the workspace.

Ease of Use and Programming Interface:

The PolyScope interface of UR10e and other UR models is known for its user-friendly design, making it easy to program even for operators with minimal robotics experience. Dasoon also offers a user-friendly programming interface with Doosan's DART platform. CRX 10iA also has a reputation for being easy to program with an intuitive teach pendant.

Safety Features:
All models mentioned are collaborative robots and integrated force/torque sensors for collision detection, and power and force limiting. all designed to operate safely alongside human workers without additional safety fencing.

Cost:
According to my investigation, pricing can vary based on region and configuration, but it still looks like UR10e competitively has a better price than both CRX10iA, and Doosan M1013.

Conclusion:
While the CRX10iA and Doosan M1013 are capable robots that meet the client's requirements for payload and reach, I recommend selecting the UR10e due to its slightly higher payload capacity and reach.
These factors make the UR10e the best choice for the client's pick and place application, ensuring ease of use, flexibility, and lower total cost.

2. Gripper Selection
Considering the project requirements for picking and placing a cardboard box (300mm x 400mm x 300mm, 7kg) and a wooden cube (40mm x 40mm x 40mm, 0.5kg), the optimal gripper should handle both objects. After reviewing the gripper options provided in the documents, the recommended choice is the OnRobot VG10 Electric Vacuum Gripper. It has flexible arms and adjustable vacuum enabling the VG10 to handle a variety of objects in many different sizes and making it suitable for both the cardboard box and the wooden cube. Also it has two separate air channels which allow for dual gripping and can enhance productivity. Additionally the VG10 does not require an external compressor that simplifies the setup and reduces the need for additional equipment. So according to all these features VG10 would be a great choice.

3. Calibration Procedure
The calibration procedure ensures that the robot accurately understands the positions of the tables (Table A and Table B) and can consistently pick and place objects correctly. For performing the calibration we have to follow the below steps:

- Ensure the robot is in the home position.
- Position Table A and Table B within the designated workspace.
- Use a known calibration object with precise dimensions.
- Place the calibration object at a predefined reference point on Table A.

- Use the robot's teach pendant to manually guide the end-effector to the exact location of the calibration object.
- When the end-effector is precisely at the reference point, record this position in the robot's control system.
- Save the coordinates and orientation of the end-effector at the reference point.
- Repeat the process for multiple reference points if necessary to ensure the entire surface of the table is accurately mapped.
- Repeating the same process for Table B:
- Move the calibration object to Table B and place it at the corresponding reference point.
- Repeat the teaching and recording process for Table B.

Verification:

- After calibrating both tables, verify the accuracy by guiding the robot to a few random points on both tables using the saved coordinates.
- Ensure that the end-effector reaches the expected positions accurately.

Optional question:

What are the restrictions and internal safeguards of the cobot that allows it to be used without
any other external safeguard?

Cobots are designed with various internal safeguards and restrictions that allow them to operate safely alongside humans without requiring additional external safeguards. These features ensure that cobots can work in environments where traditional industrial robots would need protective barriers or safety cages. Some key restrictions and internal safeguards are listed below:

1. Force and Torque Limiting
   Cobots are equipped with sensors that measure force and torque. These sensors detect when the robot comes into contact with an object or person and limit the force applied to avoid injury. If the force exceeds a predefined threshold, the cobot will stop its movement.

2. Speed and Separation Monitoring
   Cobots can adjust their speed based on their proximity to humans. When a person approaches, the robot slows down or stops completely to prevent collisions.

3. Power and Force Limiting (PFL)

Cobots are designed to inherently limit the power and force they can exert. This ensures that even in the event of a collision, the energy transferred is low enough to prevent serious injury.

4. Protective Stop and Safety Monitored Stop
   Cobots can be programmed to enter a protective stop mode if an unsafe condition is detected. This mode halts all movements until the issue is resolved.

5. Collision Detection and Avoidance
   Cobots are equipped with collision detection algorithms that allow them to sense unexpected obstacles in their path and take immediate action to avoid collisions.

6. Lightweight and Compliant Design
   Many cobots are designed to be lightweight and compliant, meaning they can absorb impacts and reduce the risk of injury. The compliance allows for some give and take during interactions with humans.

7. Intuitive Programming and Safety Feature
   Cobots come with user-friendly programming interfaces that allow operators to set safety parameters easily. These parameters include speed limits, force thresholds, and safe zones.

Program (Proof of Concept)

For the proof of concept, I set up the ROS environment for the UR10e robot and prepared two scripts to control the robot using both joint positions and Cartesian positions. These scripts guide the robot from the home position to the pickup position, then to the drop-off zone, and finally back to the home position. Additionally, these scripts output all requested positions in the terminal.

Unfortunately, due to my tight schedule, I couldn't allocate more time to attach the gripper to the robot and add the cube to the Gazebo simulation world. Therefore, the scripts currently move the robot to the pick and place positions without actually grasping the cube. I apologize for this shortcoming and hope these efforts meet your expectations.