1)The architecture component diagram with accurate description detailing where different design patterns could play a role in the final architecture implementation.

2)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).

3)Define the KPI of the main components :

Crtieria evaluated
Correctness and Functionality

- Performances and Efficiency
- Scalability/Extensibility
- Reliability and Fault Tolerance
- Interoperability/Communication
- Security/Robustness

Does the software behave as expected in normal and edge cases?
Do all components interact correctly ?
Can the system handle unexpected/malicious inputs?

▪ Orchestration Manager:

The criteria that will be evaluated for this component will be the correctness and functionality, the security and robustness.

Main key performance indicators (KPIs):

▪ Orders handling:

- Percentage of order sent by the points of sale successfully received by the Orchestration Manager: computed by taking the ratio between the number of orders received by the Orchestration Manager and the number of orders sent by the points of sale and by multiplying the total by 100. By computing this parameters, one can verify if the Orchestration Manager successfully receives every order messages published by the points of sales.

- Percentage of incorrect orders received and successfully analyzed: computed by taking the ratio between the number of incorrect orders correctly processed and the number of incorrect orders received and by multiplying the total by 100. By sending deliberately wrong orders and computing this parameter, one can effectively assess the robustness of the Orchestration Manager's order handling is case of unexpected errors in the order sent by the point of sales.

- Percentage of correct orders correctly analyzed: computed by taking the ratio between the number of correct orders correctly processed and the number of correct orders received and by multiplying the total by 100. By computing this parameter, one can assess how well the Orchestration Manager analyzes correct orders and proceed in nominal conditions.

▪ Handling of the availability updates of the TIAGo robots:

- Percentage of availability messages sent by the TIAGo robots successfully received by the Orchestration Manager: computed by taking the ratio between the number of availability messages received by the Orchestration Manager and the number of availability message sent by the TIAGo robots and by multiplying the total by 100. By computing this parameters, one can verify if the Orchestration Manager successfully receives every availability messages published by the TIAGo robots.

- Percentage of correct availability messages successfully analyzed: computed by taking the ratio between the number of correct availability messages successfully analyzed by the Orchestration Manager and the number of correct availability messages received by the Orchestration Manager and by multiplying the total by 100. By computing this number, one can verify if the statuses of the TIAGo robots are correctly processed in the nominal conditions.

- Percentage of incorrect availability messages generating the publication of an error message: computed by taking the ratio between the number of incorrect availability messages generating the publication of an error message and the number of incorrect availability messages received by the Orchestration Manager and by multiplying the total by 100. By computing this number, one can verify if the Orchestration efficiently raises error messages in the case of a TIAGo robot updating him with a wrong status.

▪ Handling of the position updates of the TIAGo robots:

- Percentage of position messages sent by the TIAGo robots successfully received by the Orchestration Manager: computed by taking the ratio between the number of position messages received by the Orchestration Manager and the number of position messages sent by the TIAGo robots and by multiplying the total by 100. By computing this parameters, one can verify if the Orchestration Manager successfully receives every position messages published by the TIAGo robots.

- Percentage of correct position messages successfully analyzed: computed by taking the ratio between the number of correct position messages successfully analyzed by the Orchestration Manager and the number of correct position messages received by the Orchestration Manager and by multiplying the total by 100. By correct position messages, we mean that the position tuple received does not give the robot a position where some parts of it intersects with a wall or a table. By computing this number, one can verify if the positions of the TIAGo robots are correctly processed and updated in the nominal conditions.

- Percentage of incorrect position messages generating the publication of an error message: computed by taking the ratio between the number of incorrect position messages generating the publication of an error message and the number of incorrect position messages received by the Orchestration Manager and by multiplying the total by 100. By computing this number, one can verify if the Orchestration efficiently raises error messages in the case of a TIAGo robot updating him with a position which make it intersect with a wall or a table, both these situation being physically impossible.

▪ Distance computation:

- Percentage of distances computed correctly: computed by taking the ratio of the number of correctly computed distances over the number of distances computed, the total multiplied by 100. By computing this parameter, one can verify if the Orchestration Manager computes successfully the distance between two objects. This ability is essential for giving the orders to optimal TIAGo robot.

▪ Publication of orders to the TIAGo robots:

- Idleness coefficient: computed by taking the minimum between the number of TIAGo robots available at the end of a loop and the number of orders that remains to be given to a TIAGo robot. The lower this parameter, the better the order publication ability of the Orchestration Manager. If this parameter is greater than zero, it means that it remains at least one TIAGo robot available and at least one order that has to be given at the end of the control loop, meaning that the Orchestration Manager is not performing correctly.

▪ Analysis of the message transmission part of the component:

-Percentage of situations where the messages containing the orders for the TIAGo robots have been correctly sent: computed by dividing the number of times the order messages have been correctly sent by the numbers of attempts to send the order messages and multiplying the total by 100. By computing this parameter, one may assess the ability of the Orchestration Manager to send the orders to the TIAGo robots.

-Percentage of situations where the error messages have been correctly sent: computed by dividing the number of times the error messages have been correctly sent by the numbers of attempts to send the error messages and multiplying the total by 100. By computing this parameter, one may assess the ability of the Orchestration Manager to send error messages to the staff.

▪ Reasoning About Food Placement:

▪ Analysis of the table layouts:

NB: for these tests, we divide the table in squares of 1 cm of side.

- Percentage of occupancy grid correctly analyzed (with respect to the fact that the squares are free or not): computed by increasing the counter of squares correctly guessed if the reasoning component guess correctly the state of one square (A square is correctly guessed if, when the reasoning component tells that the square is empty, it really is on the real table, and, if it tells that it is occupied, it is at least partially occupied on the real plate). We then divide the number of squares correctly guessed by the number of square of the table and multiply the total by 100. The computation of this parameter allows one to quantify the accuracy of the table analyzing part of the Reasoning About Food Placement component. The closer this parameter is from 100%, the more accurate the analyzing part of the Reasoning About Food Placement component is.

- Percentage of occupancy grid correctly analyzed (with respect to the fact that the squares are part of a free spots or not): computed by increasing the counter of squares correctly guessed if the reasoning component guess correctly the state of one square (A square is correctly guessed if, when the reasoning component tells that the square is part of a free spot, it really is on the real table, and, if it tells that it is not part if a real free spot, it is at least partially occupied on the real plate). We then divide the number of squares correctly guessed by the number of square of the table and multiply the total by 100. The computation of this parameter allows one to quantify the accuracy of the table analyzing part of the Reasoning About Food Placement component in determining free spots. This parameter allows one to verify that small free areas of the table are not seen as free spots by the Reasoning About Food Placement component. The closer this parameter is from 100%, the more accurate the analyzing part of the Reasoning About Food Placement component is.

- Percentage of optimal placement correctly determined: computed assuming that the free spots to place the plate are correctly determined by the Reasoning About Food Placement component (this parameter has no meaning if that is not the case). First we determine the common area between the optimal placement area determined by the Reasoning About Food Placement component and the one computed in the real world. Then we divide it by the area of the plate and multiply it by 100. The closer to 100% this parameters is, the more efficient is the Reasoning About Food Placement component ability to locate the optimal placement for the plate.

▪ Analysis of the determination of the risks:

- Percentage of situations where the risks have been guessed correctly: provided by the optimal placement of the plate and the different object on the table for different situation, we compute the number of times the risk has been correctly determined by the Reasoning About Food Placement component. We then divide this number by the numbers of attempt to determine the risk and we multiply it by 100. By computing this parameter, one may assess how skilled is the Reasoning About Food Placement component to accurately determine the risks in a real situation (the closer this parameter is from 100% the better the ability of the Reasoning About Food Placement component to determine accurately the risks).

▪ Analysis of the message transmission part of the component:

-Percentage of situations where the messages containing the optimal placement for the plate have been correctly sent: computed by dividing the number of times the messages have been correctly sent by the numbers of attempts to send the messages and multiplying the total by 100. By computing this parameter, one may assess the ability of the Reasoning About Food Placement component to send the messages containing the optimal position for the plate.

4)The description of the integration testing KPIs and the obtained result

4)