Course Code / Title: SWE 321 – SOFTWARE DESIGN AND ARCHITECTURE

Professor: Dr. Zakarya Alzamil

Course Project Cover Sheet

QUESTIONS / STUDENT OUTCOMES: This project covers/targets the following student outcomes (SOs):

- **SO(1)**- an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics
- **SO(2)**- an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors
- **SO(3)**- an ability to communicate effectively with a range of audiences
- **SO(5)-** an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives
- **SO(7)**- an ability to acquire and apply new knowledge as needed, using appropriate learning strategies

	Problem Definition **(10 Marks)**		Architectural Document **(35 Marks)**		Design Document **(25 Marks)**		Presentation & Final Report **(30 Marks)**		**TOTAL (100 Marks)**
SO (1)	√		√		√		√		
SO (2)	√		√		√		√		
SO (3)	√		√		√		√		
SO (5)	√		√		√		√		
SO (7)	√		√		√		√		
TOTAL									

FEEDBACK SUMMARY:

………

………

………

……………………………………………………………………………………

King Saud University
College of Computer & Info Sciences
Department of Software Engineering

SWE 321 – SOFTWARE DESIGN AND ARCHITECTURE

Project

Automated Warehouse Control System (AWCS)

Final Report

Supervised By:

Dr. ZAKARYA ABDULLAH ALZAMIL
Dr. MUHAMMAD NASIR MUHAMMAD SARWAR

Prepared by:

Leader: Bader Alrashed 439102055
Ziyad Al-Nasser 439102219
Yousef Alfaifi 437100574
Faisal Almutairi 438103492
Mohammed Alqahtani 439101890

Contents

1-Introduction:

This document is concerned with developing an automated warehouse control system for Warehouse Automation Inc which is a control system that manage the goods using a new and fast technology which allow manager and operators to tracking easily and it will speed up the delivery and sorting, our team members are interested in developing this system because it uses modern technology like robots which is new at the market.

2-Problem Domain Analysis:

Traditional warehouses used to take a longer time to assemble and make the good ready to be handled also, the human mistakes will cause a lot of problem like packing a good to wrong customer, forget to put the barcode label on the package and, in the long term the salaries and labors injures risk will cost a lot. On the other hand, our new system automated warehouse control system (AWCS) is more efficient and faster, the parts needed for assembly are there and ready when they are needed, it will depend on the robots more than the human which will minimize mistakes, injuries and cost.

2.1 AWCS is consist of:

- Warehouse: is consist of conveyor belt and movable shelves and zones.
- Customer: is consist of people or factory that we are dealing with.
- Stored goods: is the goods that come from the factory which are labeled by barcode.
- Manager: who is responsible of monitoring whole process.
- Picking robot: will get the stuff from where it is racked.
- Packing robot: will pack the goods.
- Carrier: who is receiving the packed goods and deliver it to its destination.

3-The System Context View:

3.1 Storing goods process:

Factory or E-store want to store their goods, they fill the order request(input), after we receive the goods our picking robots will take the goods and stick barcode labels then store it on the shelves then adding the item to the system and notify the customer(output).

3.2 Packing an order:

E-store send an order to pack a customer goods(input), the barcode of order will be scanned to direct the picking robot to the right shelves to collect all the goods then send it to the packing robot which will assemble the items in one package then handle it to the carrier(output).

4-Functional & NFP of the System:

4.1 Functional requirements:

1. System shall allow operator and manager to track the stored goods.
2. System shall use robots to pick up the goods.
3. System shall use robots to packing up the goods.
4. System shall be able read scanned inbound goods.
5. System shall connect the warehouse to customer's web shopping cart.
6. System shall alarm Critical paths when fails occur.
7. System shall generate cost analysis report.
8. System shall generate maintenance records.
9. system shall allow admin to log in

10. system shall allow customer to request order

4.2 Non-Functional requirements:

1. System shall assemble the goods within 2 hours.
2. System shall be working 99% of the time, Total time of failure status should not exceed 10 minutes per month.
3. System shall let the operating staff to track the stored goods within 10 minutes.
4. System restart time shall not be exceeding 30 minutes.

5-Challenges:

The system is new to the market so it will be the first time to the team to develop this kind of system so there will be lack of time and experience and there is no direct communication with team member due COVID-19.

6-Projection:

1- Selecting the appropriate methodology for the project.

2- Understanding the process of organizing and managing warehouse orders.

3- Organizing our time despite the pressures.

4- Gaining the skill of assigning tasks.

5- Gaining the skill of team working.

6- Elicit the requirements that are not clear or need to be specified.

7-Documnting a real project.

8- Presenting our project in an understandable way.

7-System architecture Introduction:

This document is concerned with developing an automated warehouse control system for Warehouse Automation Inc which is a control system that manage the goods using a new and fast technology which allow manager and operators to tracking easily and it will speed up the delivery and sorting.

7.1 Changes recommended by instructor:

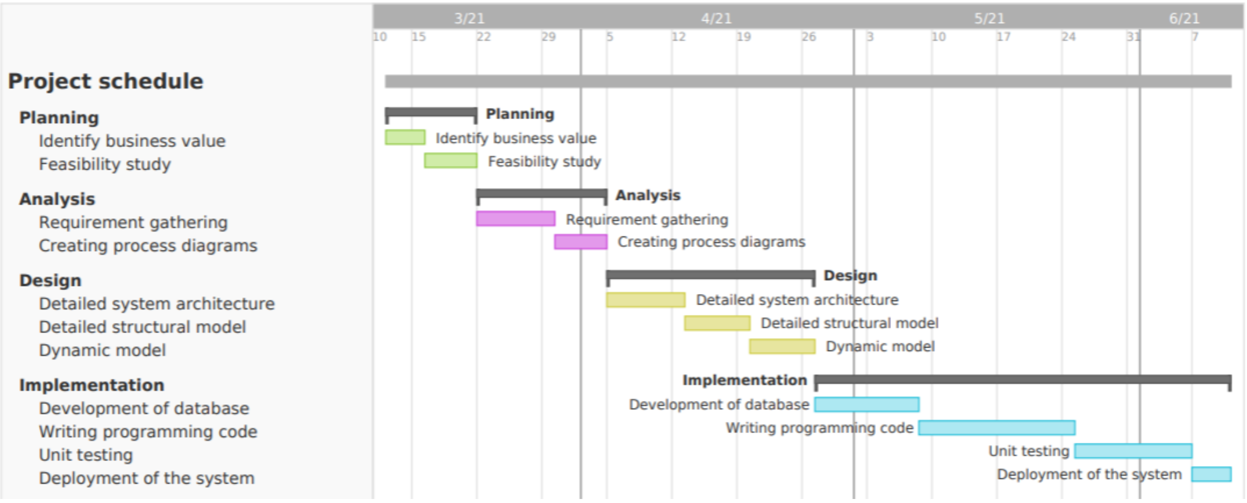
1- Numbering the headings of the document.

2- Recorrect the spelling error.

3- Numbering the requirement instead of listing them.

8-Methodology:

The system will be designed and implemented using Throwaway Prototyping, it will be the most suited methodology to use for the project it will use robot for packing and picking which is unfamiliar technology and linking the robot will be complex, and we have short time schedule to implement this system.



Planning:	**Assigned to:**
Identify business value	Ziyad
Feasibility study	Mohammed, Bader
Analysis:	
Requirement gathering	Yousef, Faisal
Creating process diagrams	Ziyad, Mohammed, Yousef
Design:	
Detailed system architecture	Bader, Mohammed
Detailed structural model	Faisal, Yousef
Dynamic model	Ziyad, Faisal, Bader
Implementation:	
Development of database	SQL programming team
Writing programming code	Java programming team
Unit testing	Testers
Deployment of the system	Deployment team

8.1 Risky part of the system:

Subsystem of Robots will be integrated as solution from existing commercial software component which is Commercial off the shelf (COTS)

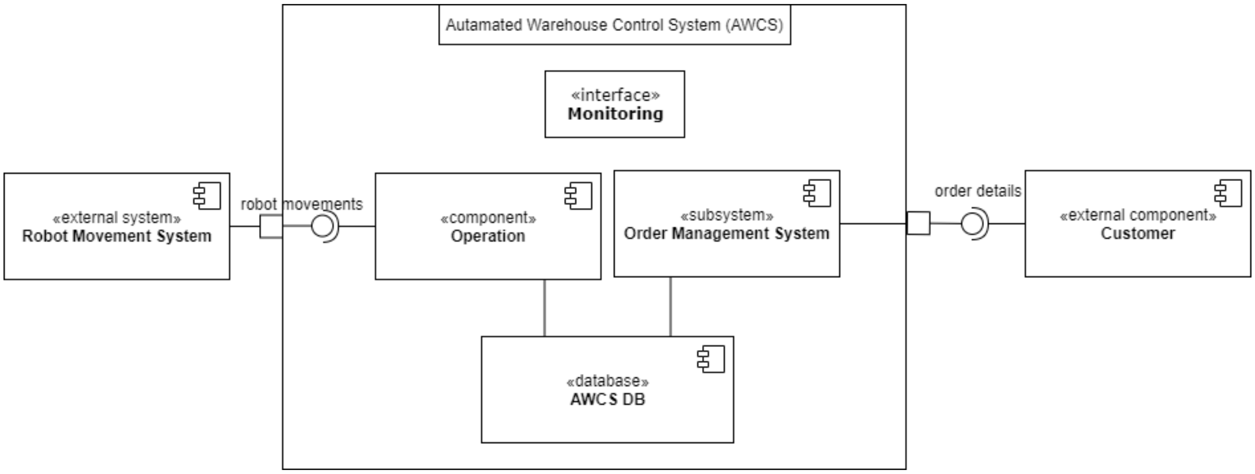
the database remains in a consistent (correct) state despite system failures (e.g., power failures and operating system crashes) and transaction failures.

9-System Architecture:

9.1 Design decisions:

1- The system shall use Oracle database.
2- The system shall be implemented using Java programming language.

9.2 Domain model:



9.3 Architectural Style:

Component-Based

We decide to choose this architecture because it has many benefits that supports our system, Component-Based maintenance and evolution are easy, you can change and update the implementation without affecting the rest of the system, also productivity for the software development and future software development is better.

Repository Architecture:

One of the biggest benefits of this architecture is data integrity which will make it a lot easier to backup and restore data, and on repository architecture it is easy to add new software components, but we decide not to choose this architecture because the data store reliability and availability is a very important issue because the centralized repository is vulnerable to failure.

9.4 Structural model:

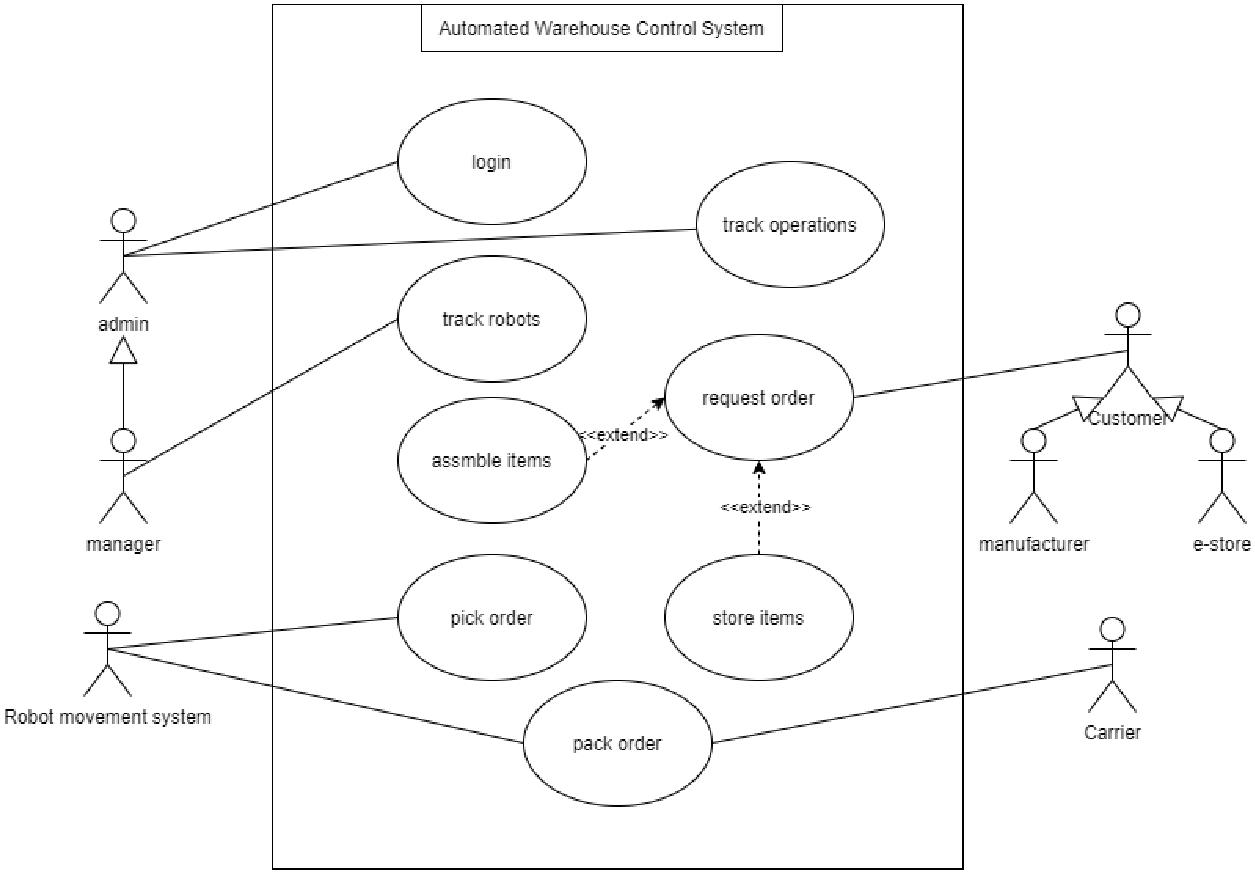
i. Main component in the system:

1- Admin
2- Database
3- Order
4- Operation

ii. 3rd party component:

Robot movement system

iii. Use-case diagram:



Flows:

1-Log in use case

Admin	System
1. in the log in interface admin enter his ID and PASSWORD.	**2. The system will compare the ID and PASSWORD.** **3. The system will display admin page**
Alternative and exceptional flows: **3.1 when the id or password does not match with corresponding database the system will display an error message (The ID or PASSWORD incorrect, please try again)**	

2-Track robots use case

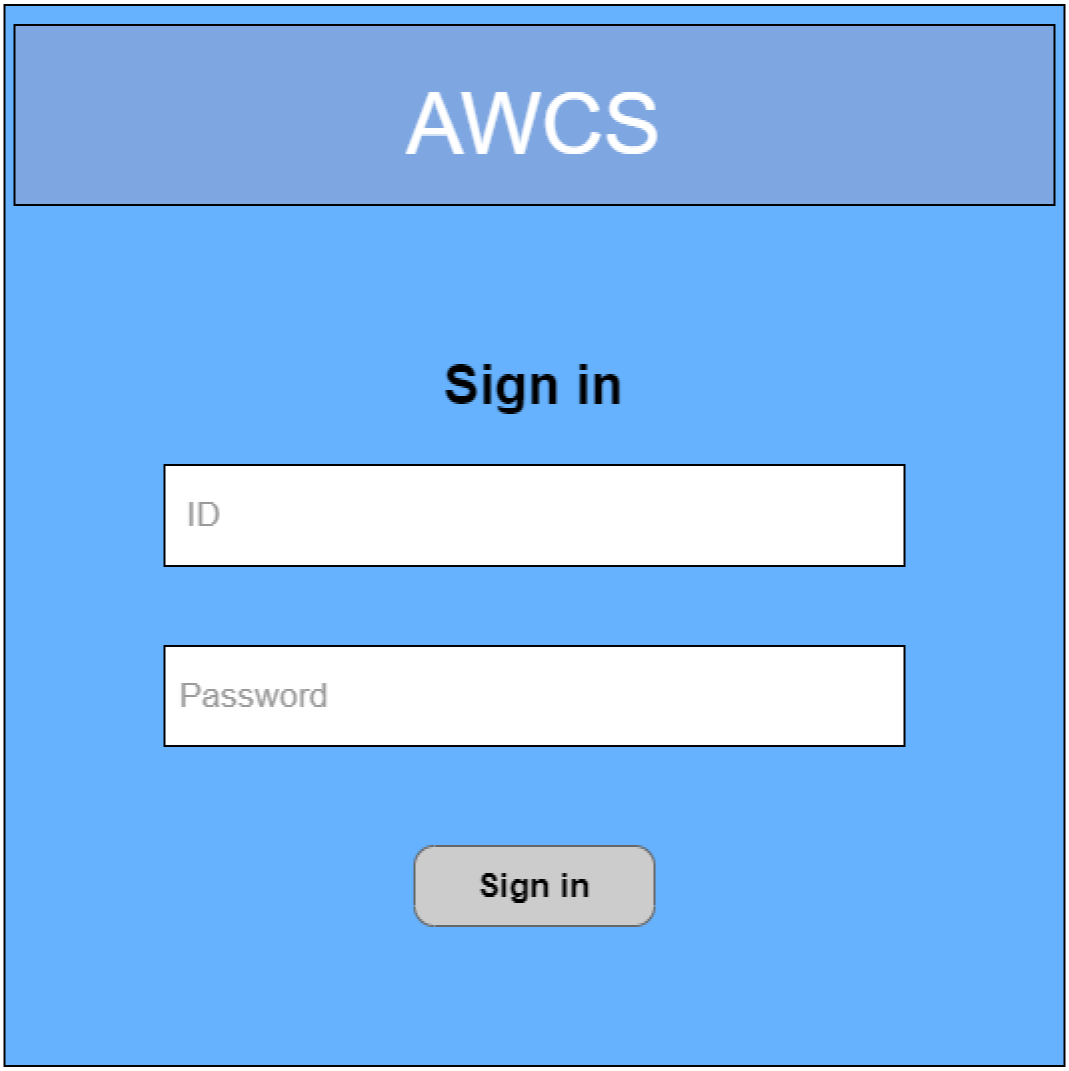
Admin	System
1. in the admin page admin access track robot by clicking on Track robot button	2. The system will display track robot page
Alternative and exceptional flows:	

3-Request order use case

Customer	System
1. Customer will fill an order request form	2. The system will check customer inputs 3. The system will save customer form in the database 4. The system will notify customer that his order is accepted
Alternative and exceptional flows: 2.1 When entering an invalid input the system will display an error message(invalid input)	

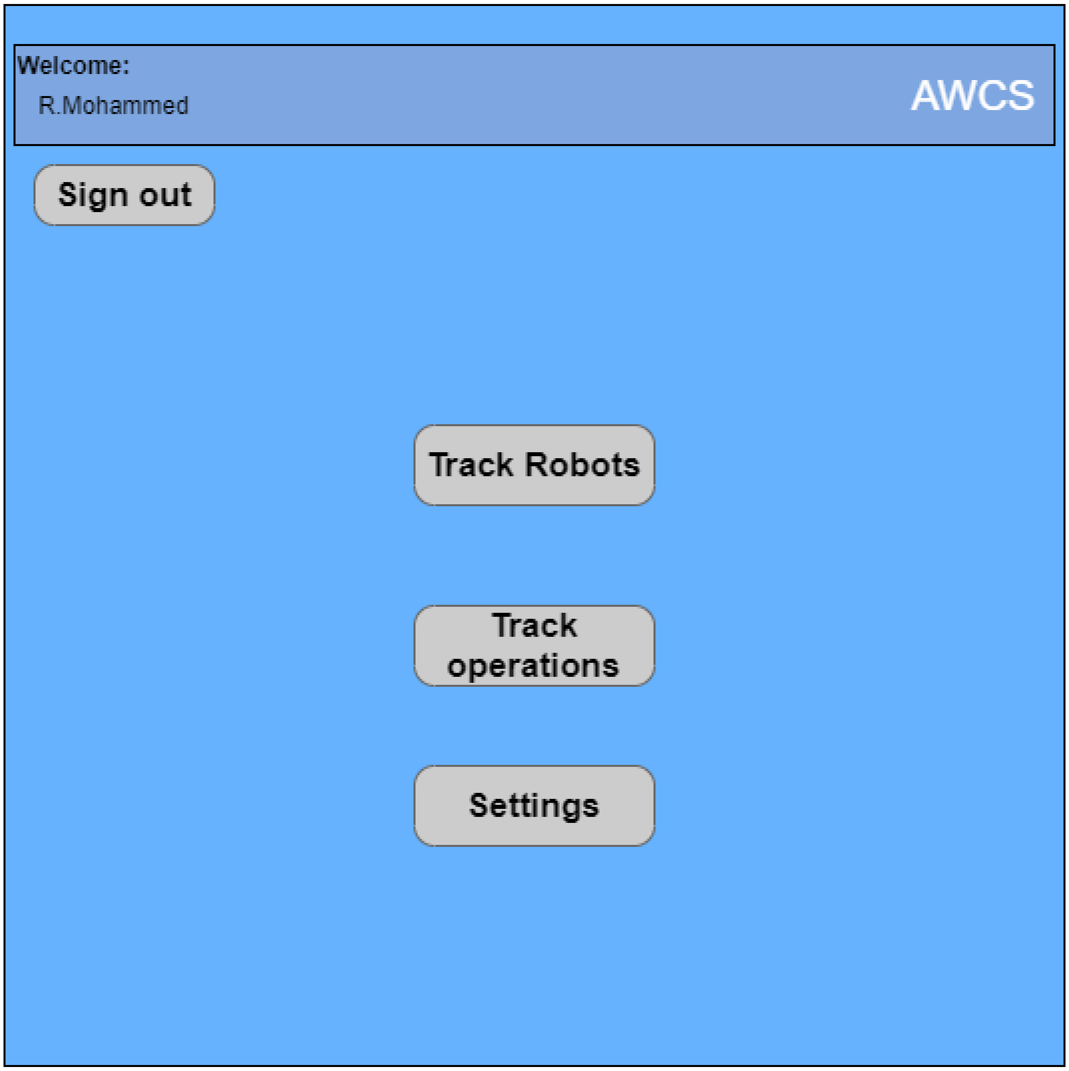
9.5 User interface:

1-Admin log in.



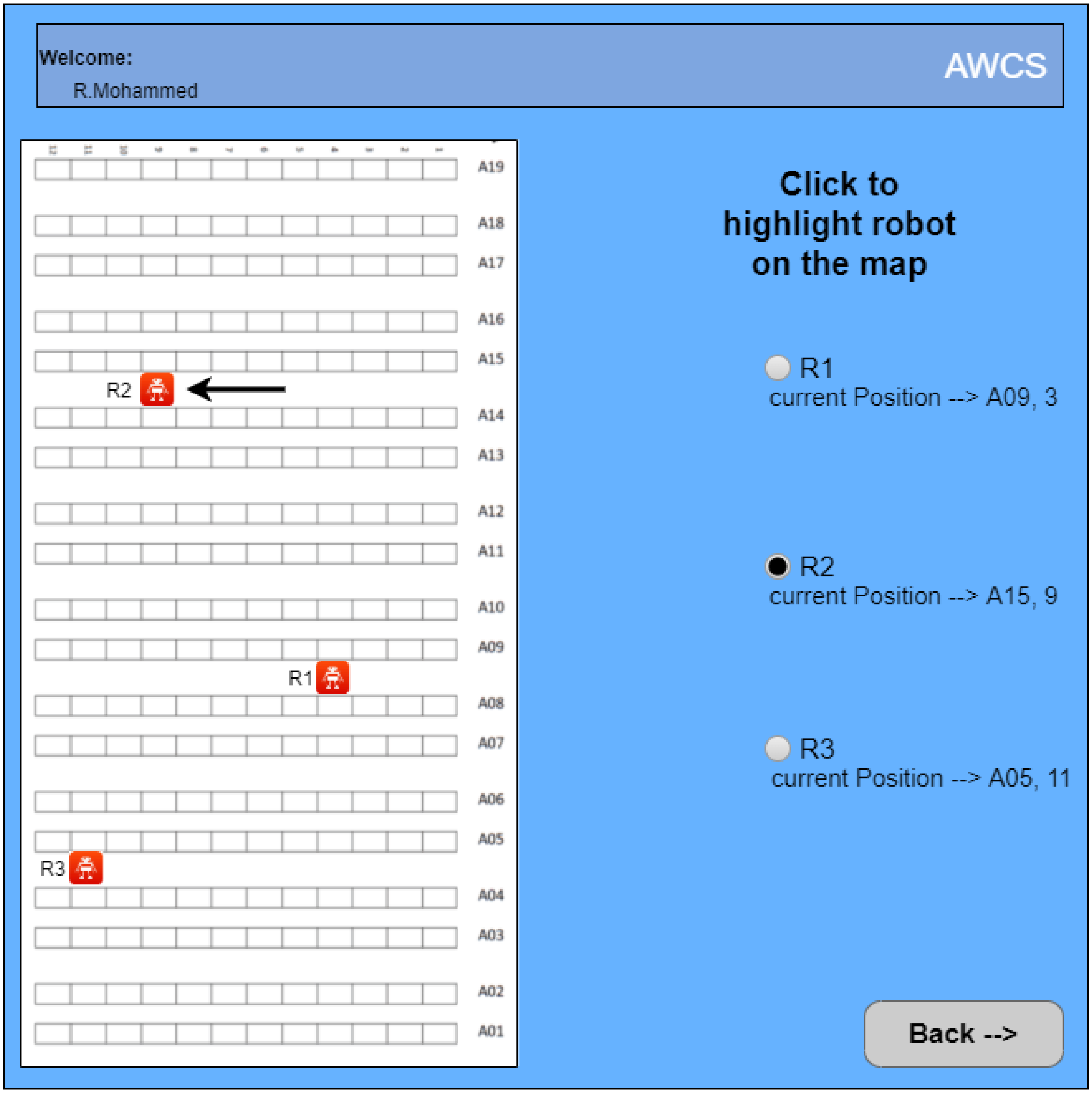
The admin enters his ID and Password to access the system.

2-Admin page



In the admin page, admin has 3 options in the page (1)Track Robots (2)Track operation (3)Settings or sign out

3-Track Robots



Welcome:
R.Mohammed

AWCS

Click to highlight robot on the map

○ R1
current Position --> A09, 3

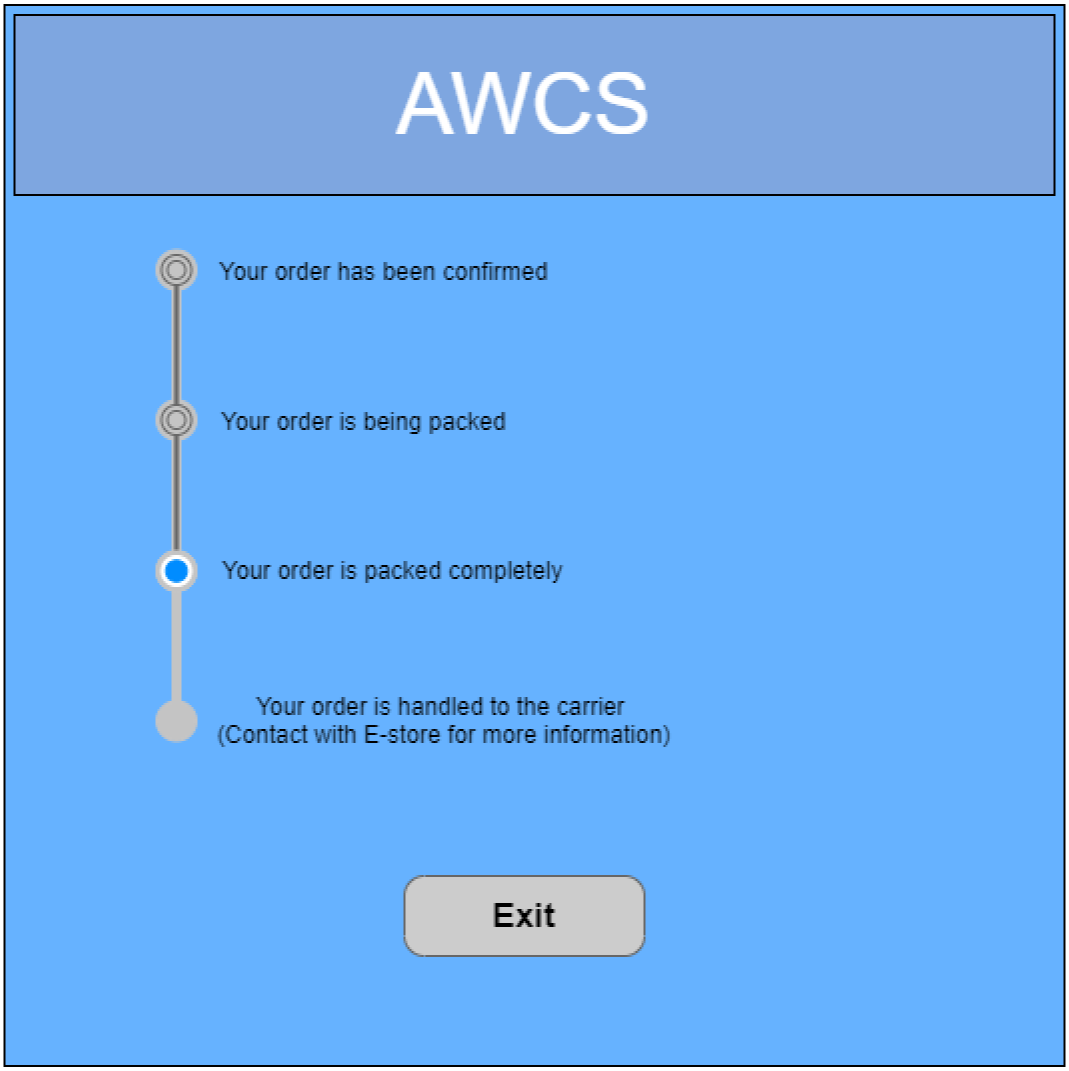
● R2
current Position --> A15, 9

○ R3
current Position --> A05, 11

In Track Robots page, on the left side of the page it will display map of the storing zone and it will display every robot with his name.

On the right side of the page, admin can choose any robot and it will be highlighted on the map also the is current tracking position under every robot name.

4-Order status



Your order has been confirmed

Your order is being packed

Your order is packed completely

Your order is handled to the carrier
(Contact with E-store for more information)

When customer orders, customer will be provided with a link which will open the Order status page.

10-Non-Functional properties:

10.1 Related quality attributes:

10.1.1 Reliability: our system needs to be reliable, because we store a lot of goods so if there is anything wrong with storing the goods (e.g., sorting them with a wrong order on wrong barcode) it will cost a lot of time to fix it.

10.1.2 Efficiency: our robots need to perform and process a lot of task in short time due to the amount of goods that need to be stored.

10.1.3 Correctness: The system should send warning message to the manager when robots heat goes up.

10.1.4 Integrity: Our system needs to prevent unauthorized access to critical paths in the system, so we need the access control to managed.

10.1.5 Usability: Due to the unfamiliar technology of robots, tracking and operating robots should be easy to learn and use.

10.1.6 Flexibility: The automated warehouse in a new idea of storing goods an in effective way, and new ideas have possibility on adding new technology, so it should be easy to add new changes to the system.

10.2 Constraint:

The system should be available on both Arabic and English language.

11-Quality assurance:

11.1 Reviews:

Formal review: We do the formal review after each phase to assure that we done everything right and we can go to the next phase.

11.2 Verification:

We can assure the developers built what was specified in the design by reading customer requirement many times to assure we specify everything that customer asked for, also by doing multiple testing through the whole project. We will assure that the developers followed standards and regulation according to ISO 9000 standards.

11.3 Validation:

11.3.1 Requirements based Testing to ensure the requirement of customer are met.

11.3.2 Penetration test to test the security of the system.

11.3.3 Scalability testing to measure the capability of the system to scale up or scale out.

11.3.4 Beta version to ensure the system perform as expected

11.4 Acceptance Criteria:

11.4.1 Procedures: install the system hardware then integrate it with the system software then the admin log in to start the system operations.

11.4.2Testing: the system should pass all test cases.

11.4.3Training: train the labors how to use the system and provide them with user manual

11.4.4 Documentation: documenting the whole system process from picking goods to handle it to carrier.

12-Future consideration:

The system will have the ability to add new upgrades to all warehouses by installing them to centralized server.

12.1 Possible features that can be added:

12.1.1 Add live chat support to communicate with the customer.

12.1.2 Subsystem of robots will be locally made (not from 3rd party).

12.1.3Sending picture of the product that will be shipped to the customer.

12.2 Supportability of the architecture:

Component-Based architecture is the most suitable for this project because, it has many benefits that support our systems, Component-Based maintenance and evolution are easy, you can change and update the implementation without affecting the rest of the system.

12.3 Changes on the architecture:

No changes will be made on this architecture.

12.4 Key parts of the system:

1-Robots

2-Database

12.5 Most critical part:

The database is the most critical part of the system because it stores everything on the system (e.g., the place of stored product, the quantity of every product).

12.6 Feasibility check:

We are going to check its feasibility by doing beta version for the robots, database and keep continuously testing them, and ensure integration process for the external subsystem (robot movement system).

13- System design Introduction:

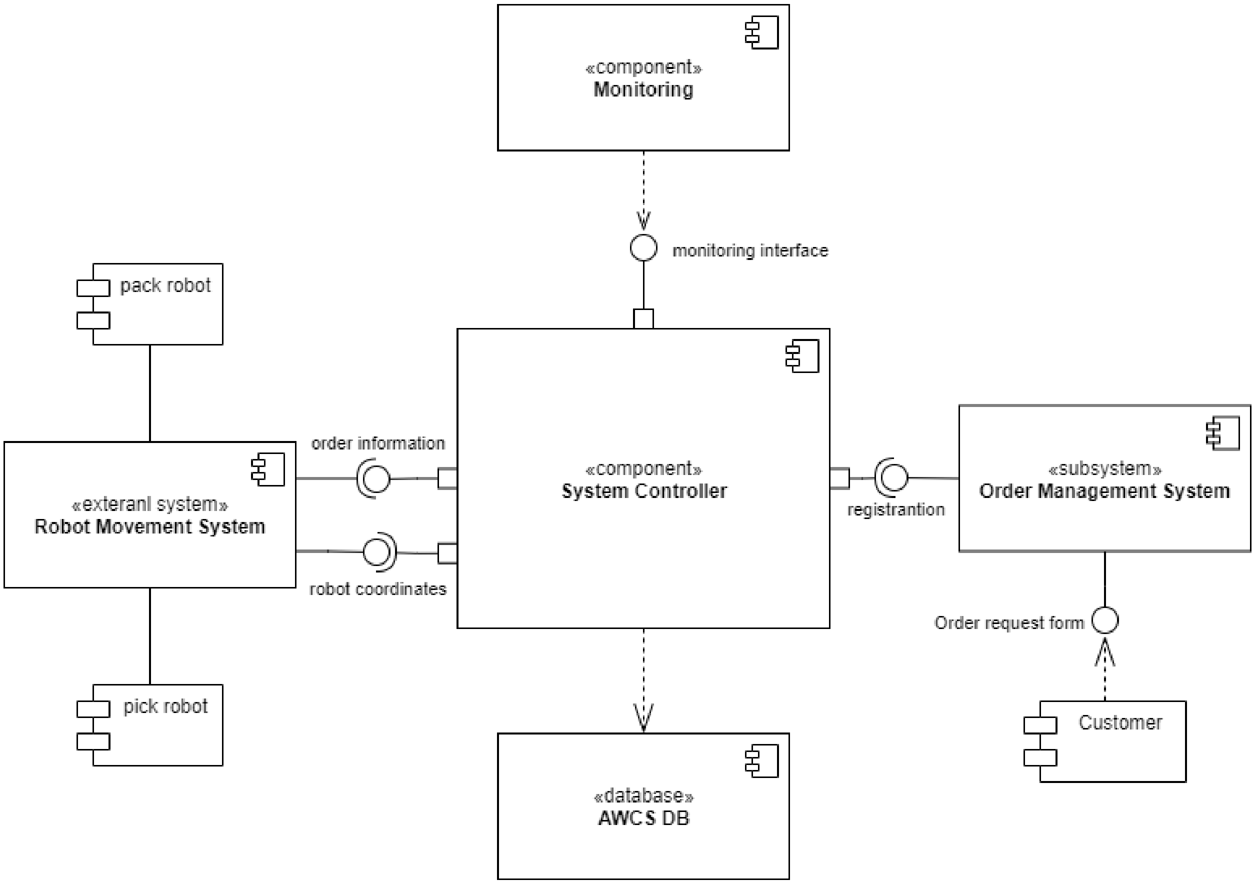
This section describes should provide a high-level system architecture diagram showing a subsystem breakout, it will be consisting of two parts: the architectural design and the detailed design, in the architectural design will refine the system architecture that have been described at previous phase The detailed design is a refinement of the architectural design in which more elaboration is performed on the components and connectors.

The static and dynamic diagrams will be covered in this document and gave us a wide point of view in how the components of the system will communicate with each other.

14-Detailed design:

a-Detailed system architectural:

a.1-Component diagram:



Component Name:	System controller
Description:	This component is responsible of taking actions in case of orders registration, robots' locations and monitoring information
Properties/data:	This component will store and receive data from AWCS
Behavior/functionality:	Save order details, provide monitoring details, update order details
Connectors and Interfaces:	It connects with order management system, monitoring, robot movement system
Dependencies:	AWCS DB
Resources:	AWCS DB

Component Name:	Monitoring
Description:	This component is concerned with provide the admin about the whole system information
Properties/data:	This component will receive data form controller
Behavior/functionality:	Provides robots coordinates and display any wronging
Connectors and Interfaces:	System controller
Dependencies:	Monitoring interface
Resources:	controller

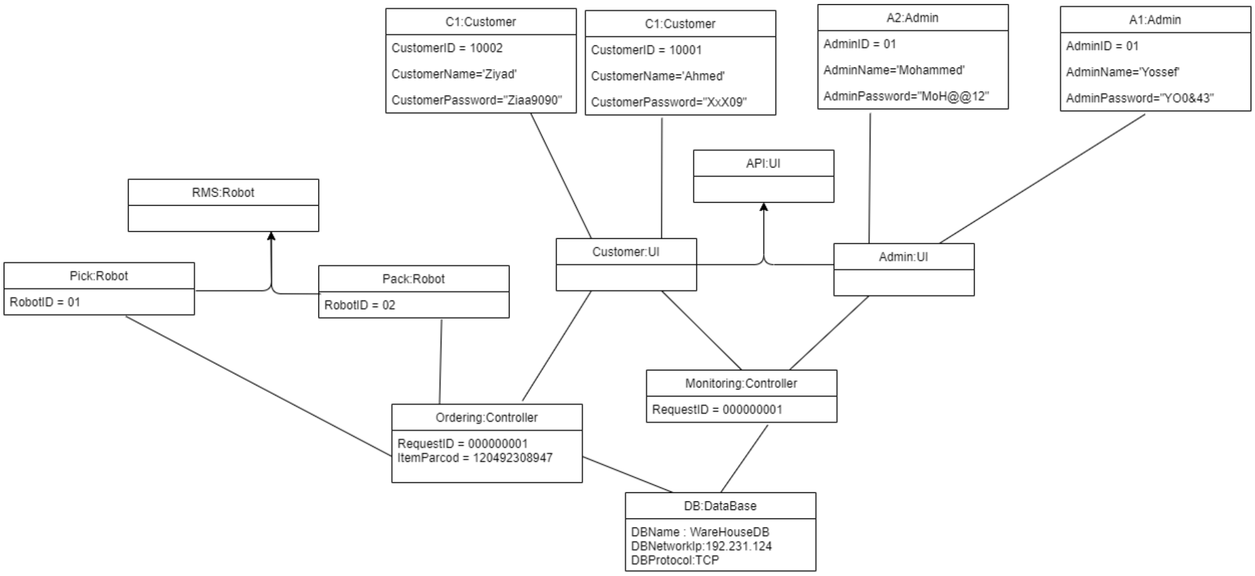
Component Name:	Order management system
Description:	This component is responsible of taking orders from customer and configure the order type
Properties/data:	Order request form
Behavior/functionality:	Provide an interface for customer for order registration and provide a complete registration form to the controller
Connectors and Interfaces:	It connects with system controller by registration interface
Dependencies:	none
Resources:	Customer's order details

Component Name:	database
Description:	This component is the database of the system and it’s connected with all warehouses
Properties/data:	Stores order details and all component status
Behavior/functionality:	Inserting, updating, deleting and retrieving data
Connectors and Interfaces:	Connected with system controller through dedicated API for accessing purpose
Dependencies:	System controller
Resources:	Database

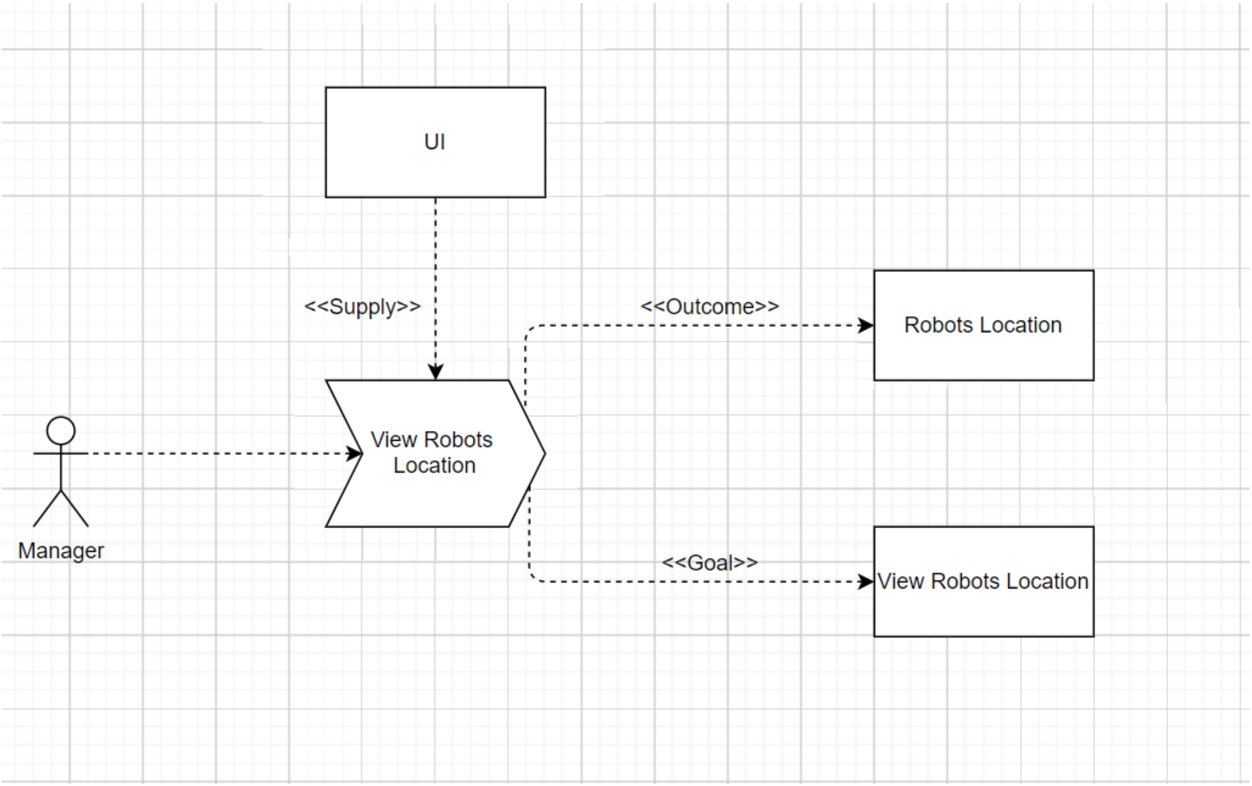
Component Name:	Robot movement system
Description:	This component is an API to connected with external system to provide robots management
Properties/data:	Robot coordinates and order information
Behavior/functionality:	Provide an API for robot movement, provide robot logs and
Connectors and Interfaces:	This component is connected with system controller by robot coordinates and order information
Dependencies:	none
Resources:	Packing robots and picking robots

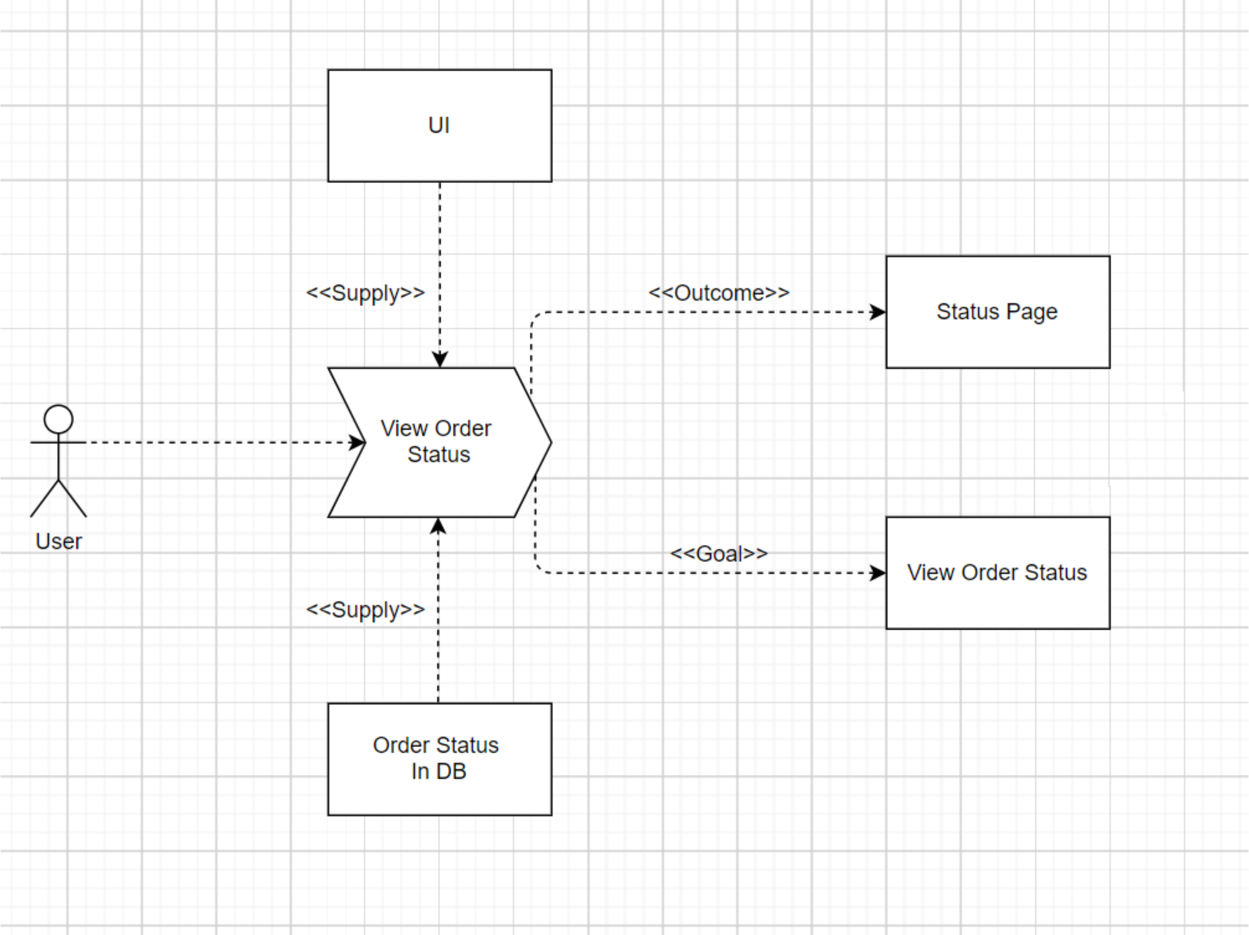
b-Detailed structural model:

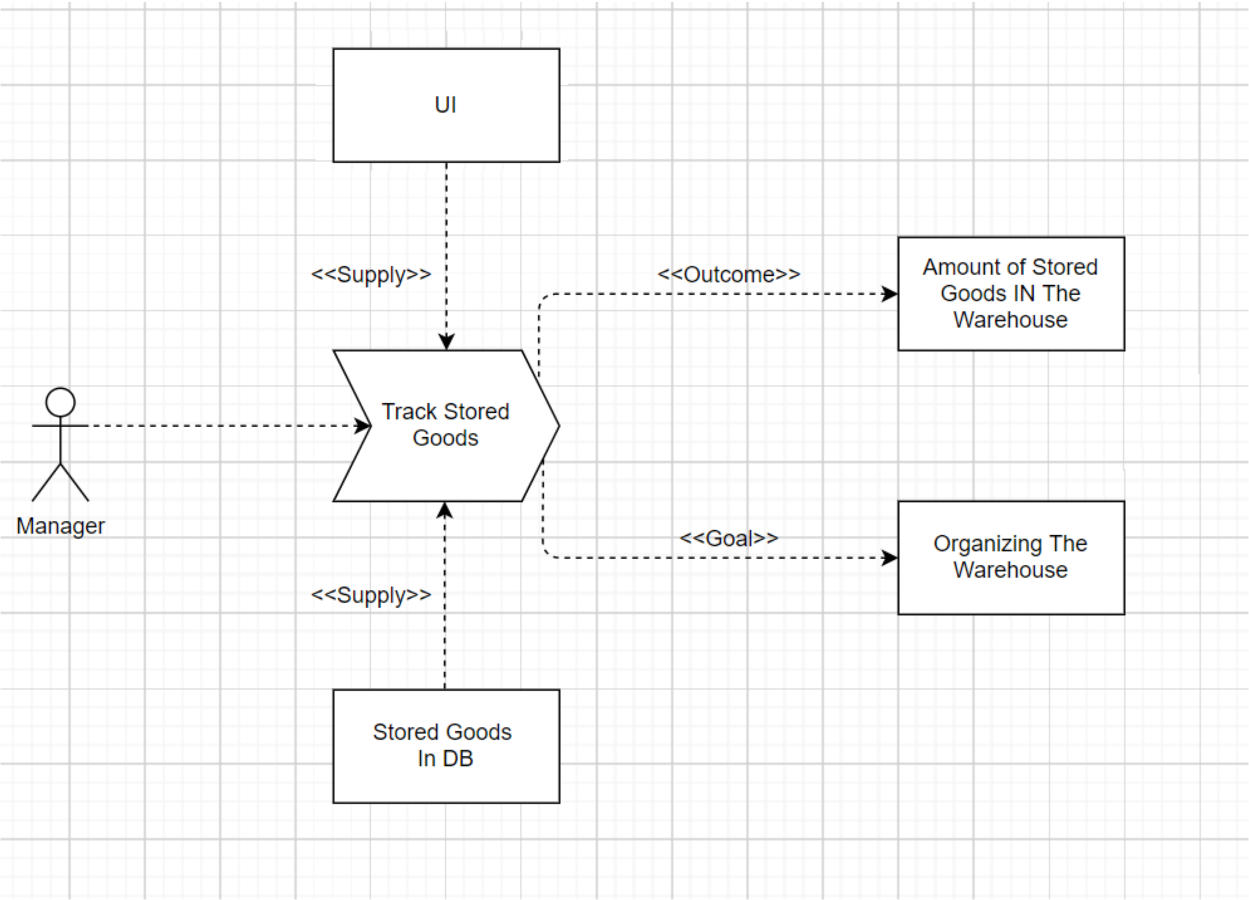
b.1-detailed object diagram:

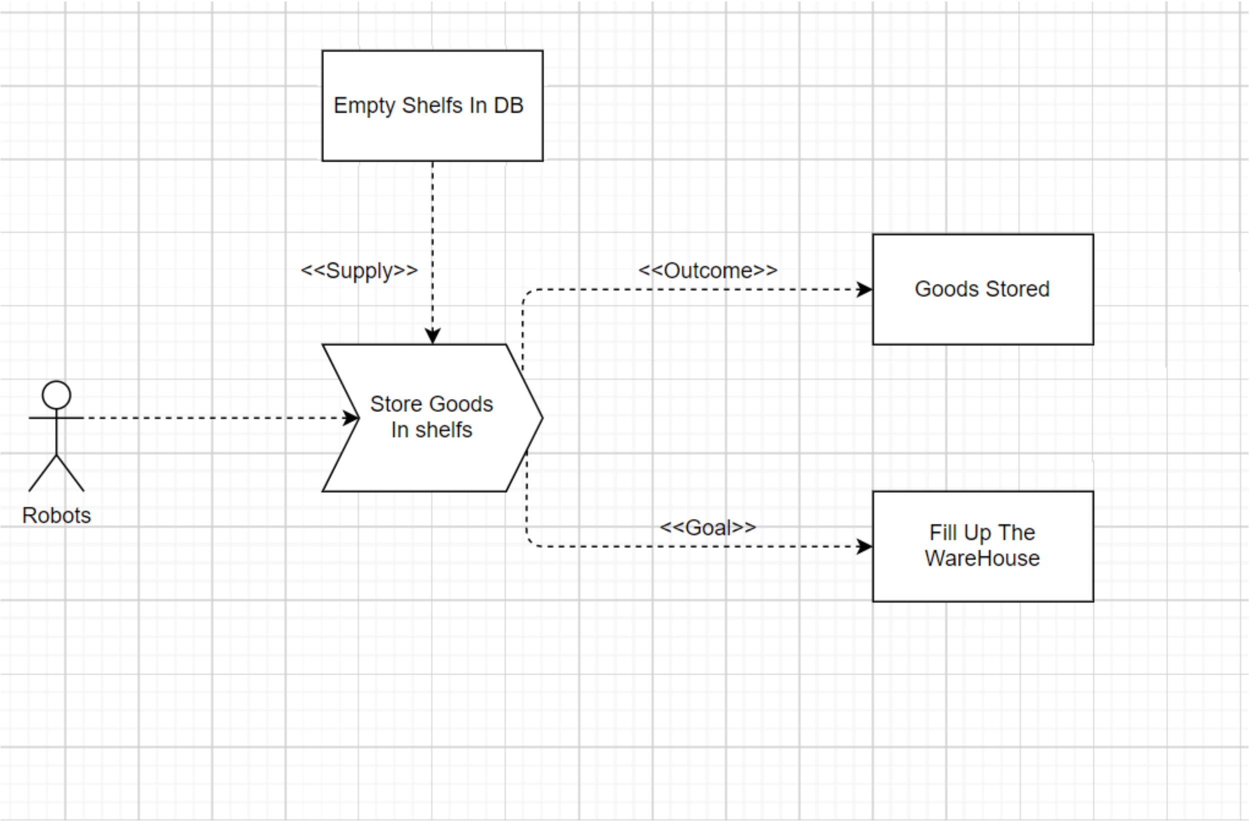


b.2-analysis diagram:

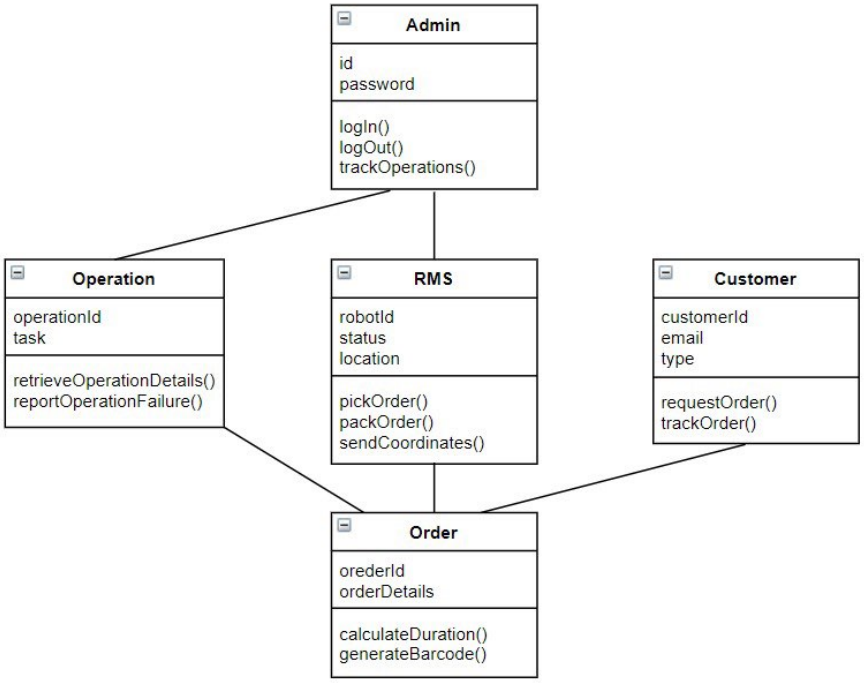



UI
<<Supply>>
<<Outcome>>
Status Page
View Order Status
User
<<Goal>>
View Order Status
<<Supply>>
Order Status In DB


UI
<<Supply>>
<<Outcome>>
Amount of Stored Goods IN The Warehouse
Track Stored Goods
Manager
<<Goal>>
Organizing The Warehouse
<<Supply>>
Stored Goods In DB

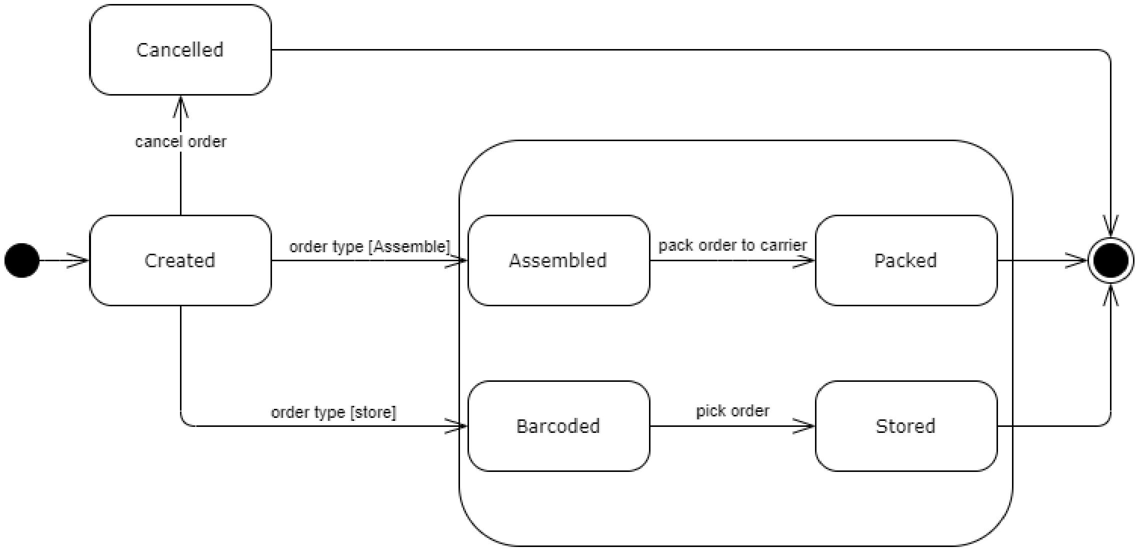


b.3-view of participating classes (VOPC):

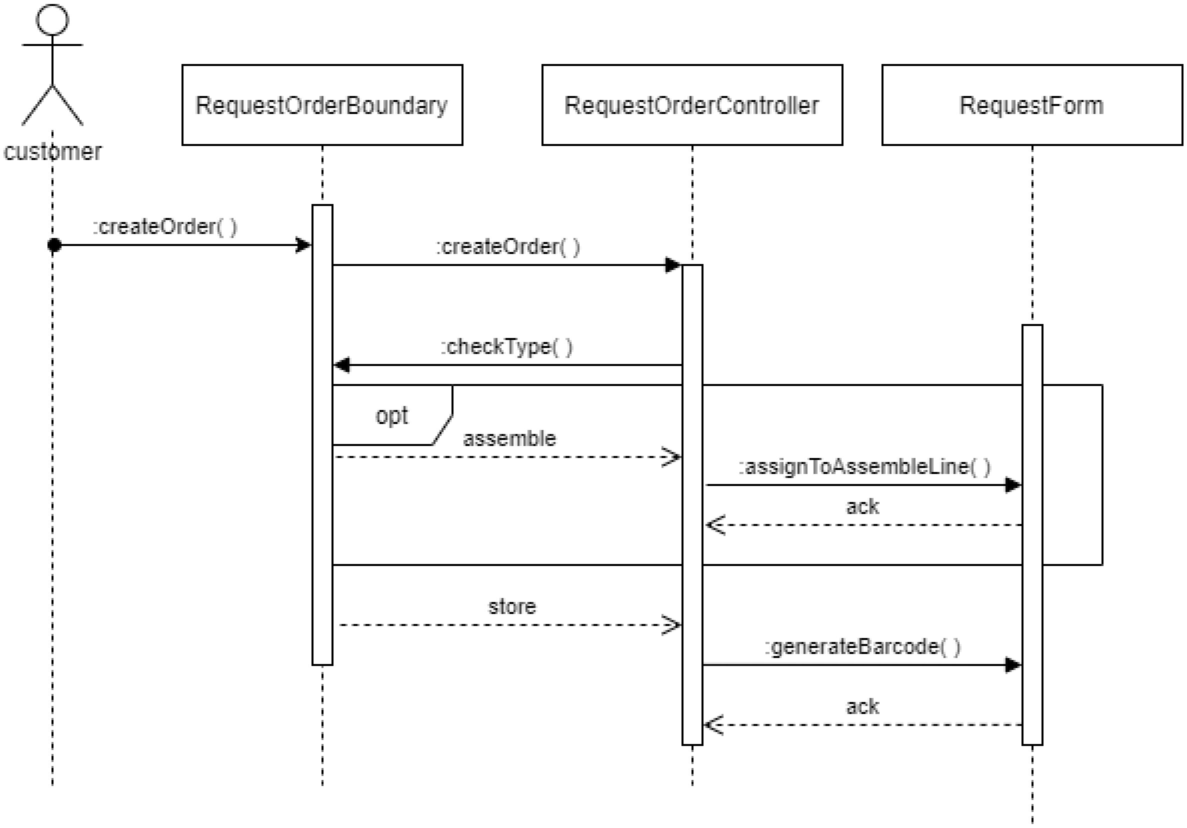


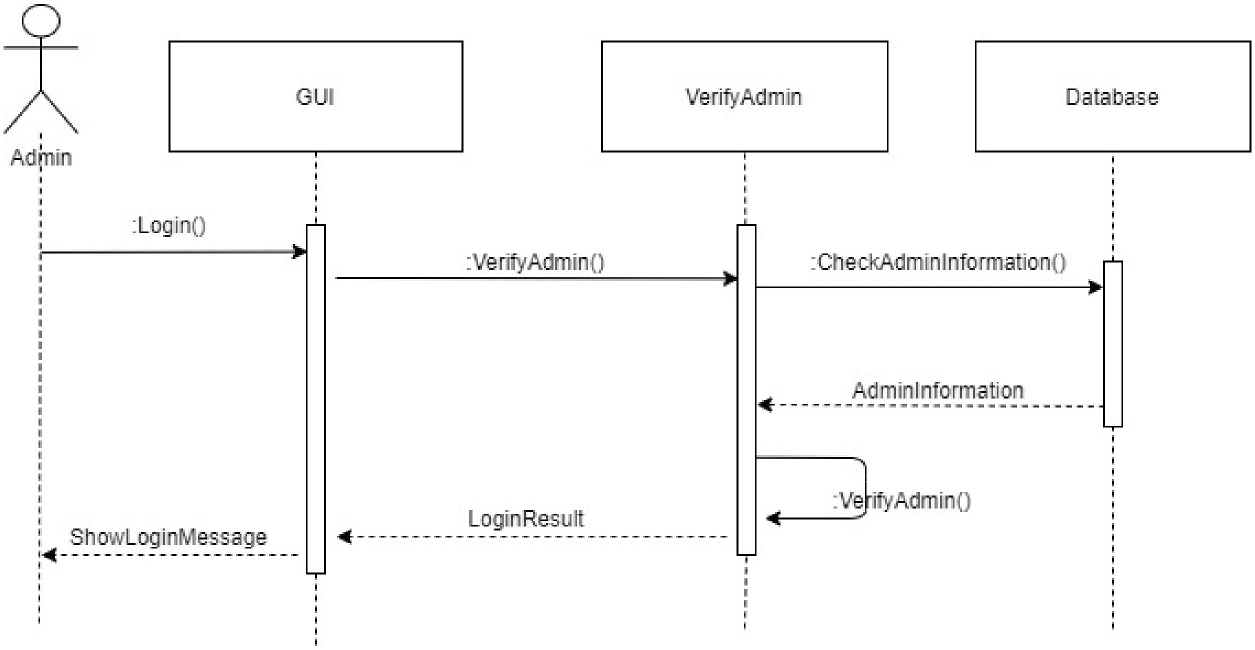
c-Dynamic model:

c.1-state diagram:



c.2-sequence diagrams:



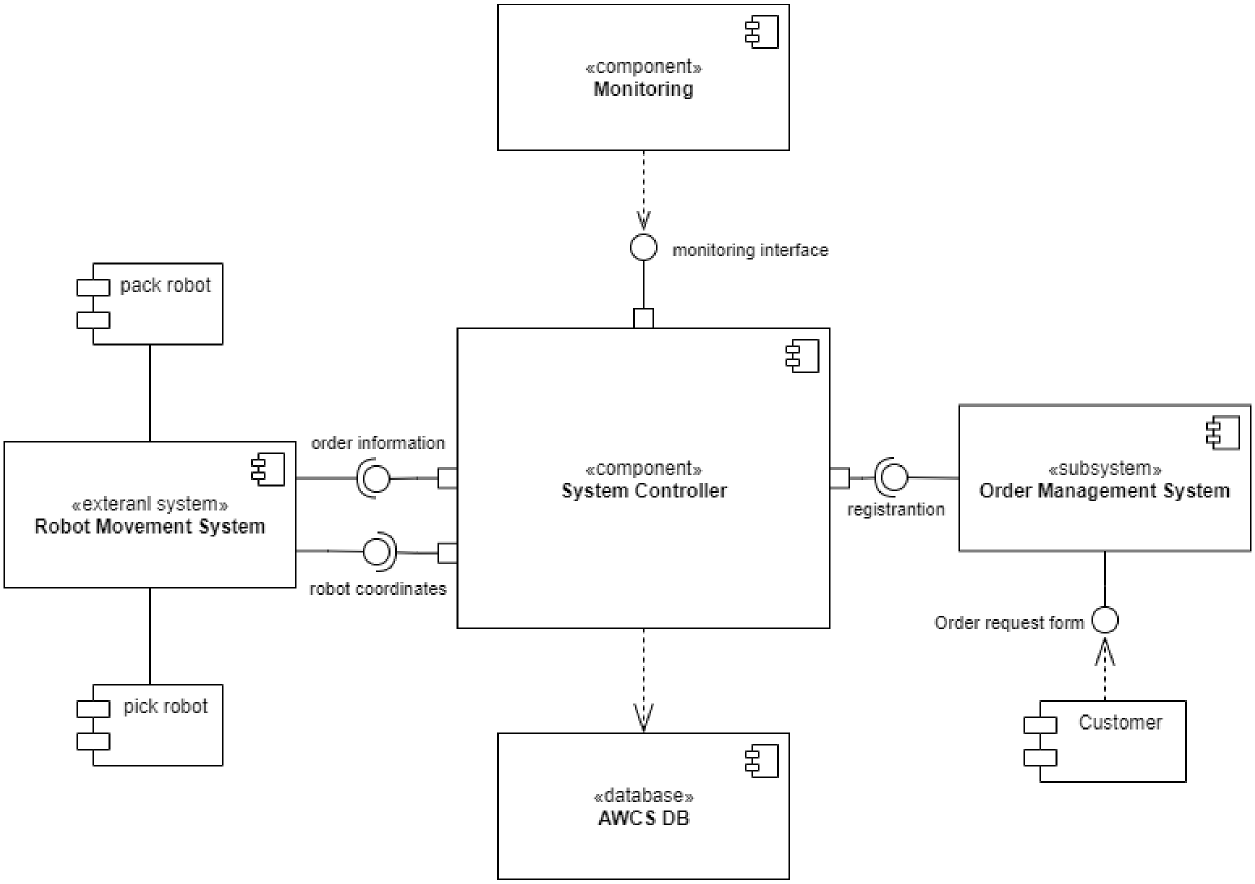


15-The architecture and design documents:

We only make two changes according to our lecturer which are numbering functional and non-functional requirement and correcting spelling mistakes in phase 1.

16-Implementation Consideration:

Mapping:



This component diagram describes the mapping of the component, connectors, interfaces, configuration of the system.

Data structures:

The system focuses mainly on double linked list, linked list, Array.

Component interaction:

System controller interacts with order management system through dedicated API to register into the database, and the customer interacts with OMS dependent on the specialized API that OMS provides, and the system controller provides API for monitoring all operations for people that have privileges, finally Robot movement system associated with picking and packing robots and it provides robot coordinates through an API for system controller and system controller provide order information through an API to RMS.

Detailed transmitted data:

when the customer order for example form Amazon GUI AWCS will receive the order and it will search for the goods in the database if its available or not then confirm the order after that customer will receive message to track the order.

Also, this is the non-functional properties related to our system:

Reliability: our system need to be reliable, because we store a lot of goods so if there is anything wrong with storing the goods(e.g., sorting them with a wrong order on wrong barcode) it will cost a lot of time to fix it.

Efficiency: our robots need to perform and process a lot of task in short time due to the amount of goods that need to be stored.

Correctness: The system should send warning message to the manager when robots heat goes up.

Integrity: Our system needs to prevent unauthorized access to critical paths in the system, so we need the access control to managed.

Usability: Due to the unfamiliar technology of robots, tracking and operating robots should be easy to learn and use.

Flexibility: The automated warehouse in a new idea of storing goods an in effective way, and new ideas have possibility on adding new technology, so it should be easy to add new changes to the system.

Also, we have one constraint which is:

The system should be available on both Arabic and English language.

Frameworks and APIs:

AWCS needs (JavaRMI) framework to facilitate communication between local and remote objects. ORB frameworks eliminate many tedious, error-prone, and non-portable aspects of creating and managing distributed applications and reusable service components. In addition, AWCS needs Robotics API to facilitate communication between system controller and Robot Management System.

17-Architecture Analysis/Testing:

For the Formal reviews we will do it after each phase to assure that we done everything right and we can go to the next phase.

For the Verification we want to assure we check requirement that customer provide by reading requirement many times to assure we specify everything that customer asked for, also by doing multiple testing through the whole project. And we will follow standards and regulation according to ISO 9000 standards.

We will implement some Validation test:

1 Requirements based Testing to ensure the requirement of customer are met.

2 Penetration test to test the security of the system.

3 Scalability testing to measure the capability of the system to scale up or scale out.

4 Beta version to ensure the system perform as expected.

Acceptance Criteria:

For the Procedures we will install the system hardware then integrate it with the system software then the admin login to start the system operations.

For the Testing, the system should pass all test cases.

For Training we will train the labors how to use the system and provide them with user manual

In the Documentation we will document the whole system process from picking goods to handle it to carrier.

Test Scenario ID	Login-1	**Test Priority**	High
Pre-Requisite	N/A	**Post-Requisite**	N/A

S.NO	**Action**	**inputs**	**Expected output**	**Actual output**	**Test result**	**Test comment**
1	**Enter valid ID and password**	**ID: 198476234 Password: ********	**Admin page**	**Admin page**	**Pass**	**[11/04, 1:11 am] : login successfully**
2	**Enter valid ID and invalid password**	**ID: 198476234 Password: ****	**The password is incorrect please try again, or hit reset password**	**The password is incorrect please try again, or hit reset password**	**Pass**	**[11/04, 1:15 am] : invalid login attempt stopped**
3	**Enter invalid ID and valid password**	**ID: 88888 Password: ********	**The ID is incorrect please try again**	**The ID is incorrect please try again**	**Pass**	**[11/04, 1:19 am] : invalid login attempt stopped**

Test Scenario ID	Scan_Barcode-1	**Test Priority**	High
Pre-Requisite	N/A	**Post-Requisite**	N/A

S.NO	Action	inputs	Expected output	Actual output	Test result	Test comment
1	Scan correct barcode	1234567	Scan complete, and goods is ready for packing	Scan complete, and goods is ready for packing	Pass	[11/04, 1:11 am] : Scan complete
2	Scan barcode which already been scanned	1234567	Scan is already have been done	Scan is already have been done	Pass	[11/04, 1:11 am] : invalid scan attempt stopped
3	Scan unclear barcode	1234#67	Scan is not complete, please try again	Scan is not complete, please try again	Pass	[11/04, 1:11 am] : invalid scan attempt stopped

18-Deployment and Mobility:

We will install the system hardware then integrate it with the system software then the admin login to start the system operations.

AWCS will be developed using Java, the network bandwidth should handle 50 order / second, the system will be deployed on windows 10, intel i5 cores, 8GB RAM, 250Gb HDD, the system needs cameras across the warehouse to control the process, also printers to print the reports, and a barcode scanner device, the system will take the data from the database which is installed in the warehouse.

19-Personal Reflection:

Now after we finished the project, we feel more comfortable architecting systems now than we did before, this project give us an experience that simulate real world project which is the part of the course that improve the quality of our architecture and design. In addition, this experience gave us a practical training which improve teamwork skills, also the course would be better if it has more practical parts and decrease the theoretical parts.