Project Title	**:**	MAERSK CONTAINER MANAGEMENT SYSTEM
Module Code	**:**	CT071-3-5-3-DDAC
Intake Code	**:**	UC3F1701SE
TP number	**:**	TP038611
Student Name	**:**	Marcus Lee Tzyy Min
Lecturer Name	**:**	DR. KALAI ANAND A/L RATNAM
Hand in Date	**:**	13 April 2018

Acknowledgement

Firstly, I would like to extend my appreciation towards my parents for the opportunity and support for my university education in terms of morale support and financial support. I would like to thank Asia Pacific University of Technology and Innovation for having this module, Designing and Developing Application on Cloud too to better understand the applications and deployments of cloud applications. Would like to express my gratitude towards Dr Kalai for his guidance and expertise in the field of cloud computing. His teaching and explanations had made it easier to understand the fundamentals of cloud computing with the various real-life experiences he provided.

Table of Contents

1.0 Introduction

1.1. Project Background

Maersk Line is the global container division founded in 1928 and is the largest operating unit of the A.P. Moller – Maersk Group, a Danish business conglomerate. It has customers through 374 offices in 116 countries, employs approximately 7,000 sea farers and approximately 25,000 land-based people and operates over 600 vessels with a capacity of 2.6 million TEU.

The company wish to support the overall business strategy from an IT perspective that will further business growth and increase organizational flexibility by consolidating all its data centres and server rooms operating worldwide onto a virtualized platform using Microsoft Azure. Maersk is currently changing over its IT setup based on Microsoft Azure, starting with the desktop environment up to container management.

1.2.Objective

To design and develop a Container Management System (CMS) to cater to manage the containers, a solution that reduces overall supply chain costs and an efficient way to manage logistics.

1.3.Scope

To develop the CMS and upload to Microsoft Azure to allow integration and the company to access the system

1.4.Requirement Specification

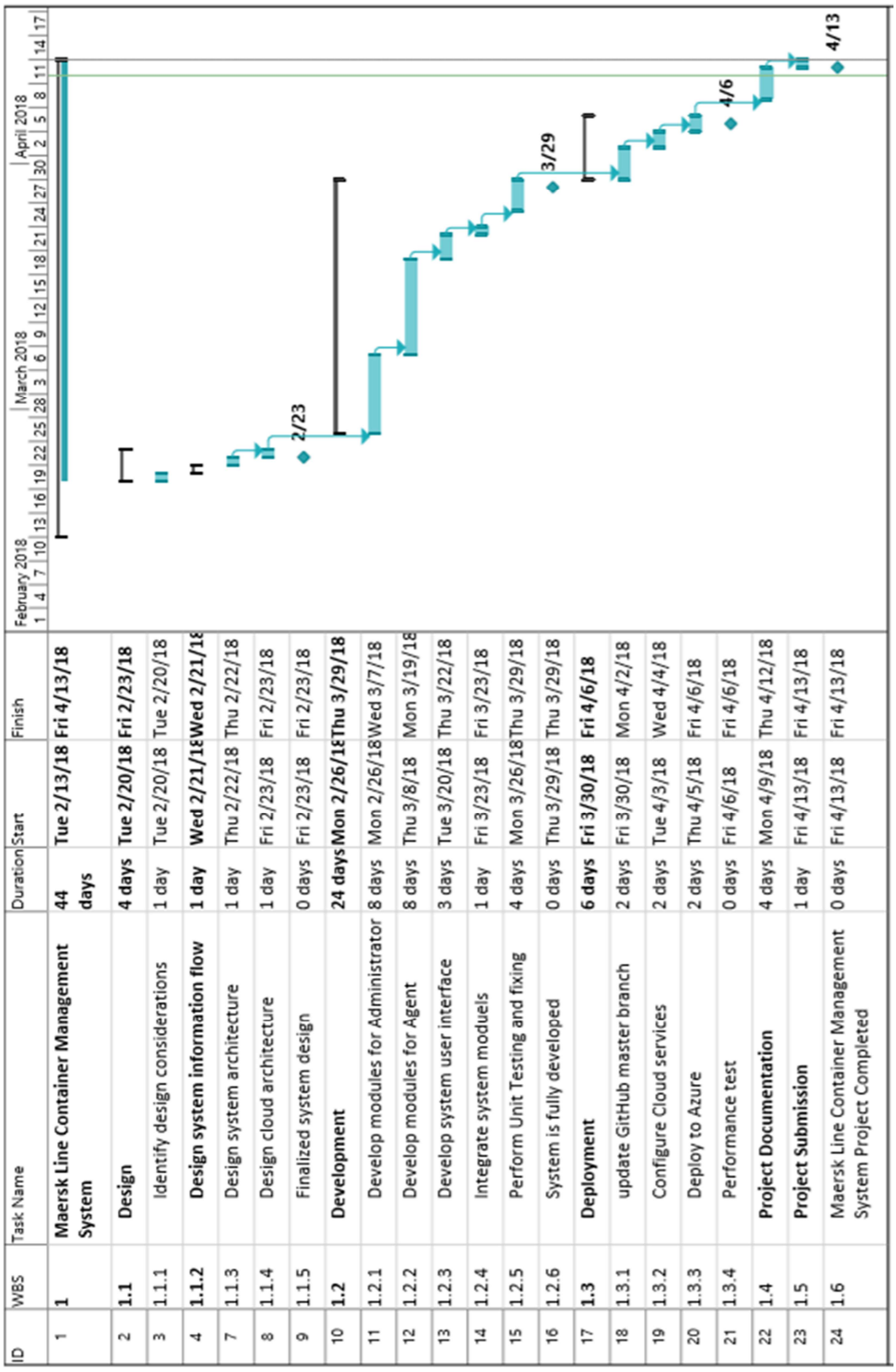
1. From import, export and transhipment processing to gate operations.
2. To be able to scale the solution to meet the needs of demands during peak seasons.
3. Improves profitability, reduce costs, increases productivity, eradicates errors and optimizes resources to future-proof your cargo handling business for high performance.
4. Assurance & reliability through Failover Management.

5. Accurately allocates inbound containers to yard locations and plan outbound containers to individual haulier vehicles, delivering an exceptional level of automation and removing human error.
6. Manage your entire booking process from schedule search to booking confirmation.

1.5.Summary of Functions Specifications

The main function of the CMS system will include 2 different logins, admin and agent. Admin of the shipping provider will be able to define schedule, register/create agents and view the different details. Agent can register customer, register the type of goods they are going to send, book the schedule and after booking receive confirmation.

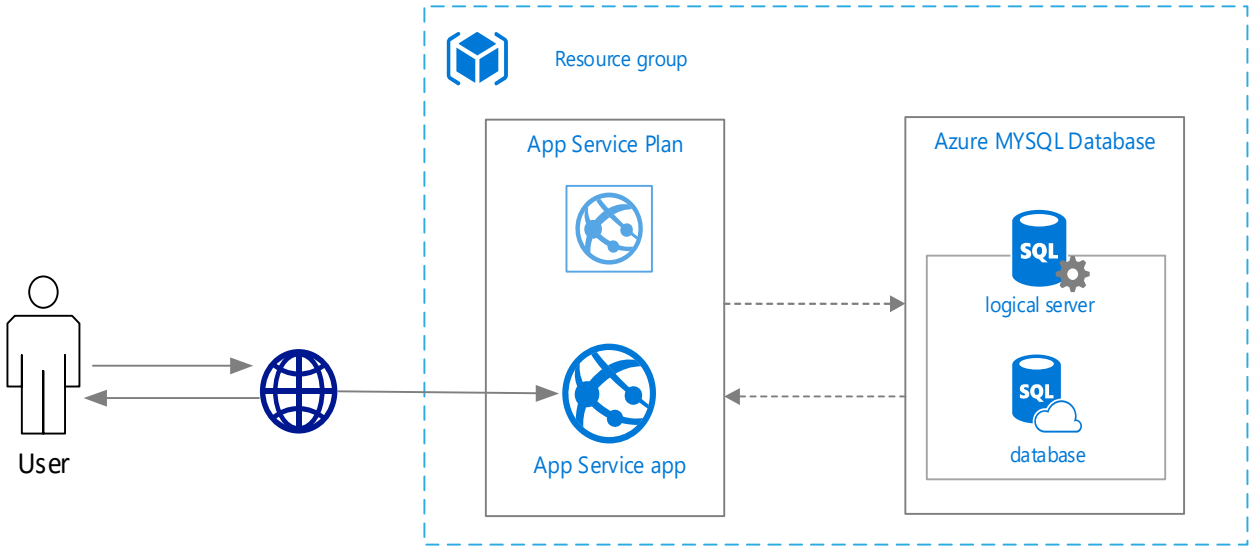
2.0 Project Plan



ID	WBS	Task Name	Duration	Start	Finish
1	1	**Maersk Line Container Management System**	**44 days**	**Tue 2/13/18**	**Fri 4/13/18**
2	1.1	**Design**	**4 days**	**Tue 2/20/18**	**Fri 2/23/18**
3	1.1.1	Identify design considerations	1 day	Tue 2/20/18	Tue 2/20/18
4	1.1.2	**Design system information flow**	**1 day**	**Wed 2/21/18**	**Wed 2/21/18**
7	1.1.3	Design system architecture	1 day	Thu 2/22/18	Thu 2/22/18
8	1.1.4	Design cloud architecture	1 day	Fri 2/23/18	Fri 2/23/18
9	1.1.5	Finalized system design	0 days	Fri 2/23/18	Fri 2/23/18
10	1.2	**Development**	**24 days**	**Mon 2/26/18**	**Thu 3/29/18**
11	1.2.1	Develop modules for Administrator	8 days	Mon 2/26/18	Wed 3/7/18
12	1.2.2	Develop modules for Agent	8 days	Thu 3/8/18	Mon 3/19/18
13	1.2.3	Develop system user interface	3 days	Tue 3/20/18	Thu 3/22/18
14	1.2.4	Integrate system moduels	1 day	Fri 3/23/18	Fri 3/23/18
15	1.2.5	Perform Unit Testing and fixing	4 days	Mon 3/26/18	Thu 3/29/18
16	1.2.6	System is fully developed	0 days	Thu 3/29/18	Thu 3/29/18
17	1.3	**Deployment**	**6 days**	**Fri 3/30/18**	**Fri 4/6/18**
18	1.3.1	update GitHub master branch	2 days	Fri 3/30/18	Mon 4/2/18
19	1.3.2	Configure Cloud services	2 days	Tue 4/3/18	Wed 4/4/18
20	1.3.3	Deploy to Azure	2 days	Thu 4/5/18	Fri 4/6/18
21	1.3.4	Performance test	0 days	Fri 4/6/18	Fri 4/6/18
22	1.4	**Project Documentation**	4 days	Mon 4/9/18	Thu 4/12/18
23	1.5	**Project Submission**	1 day	Fri 4/13/18	Fri 4/13/18
24	1.6	Maersk Line Container Management System Project Completed	0 days	Fri 4/13/18	Fri 4/13/18

3.0 Design

3.1 Cloud Architectural Diagram

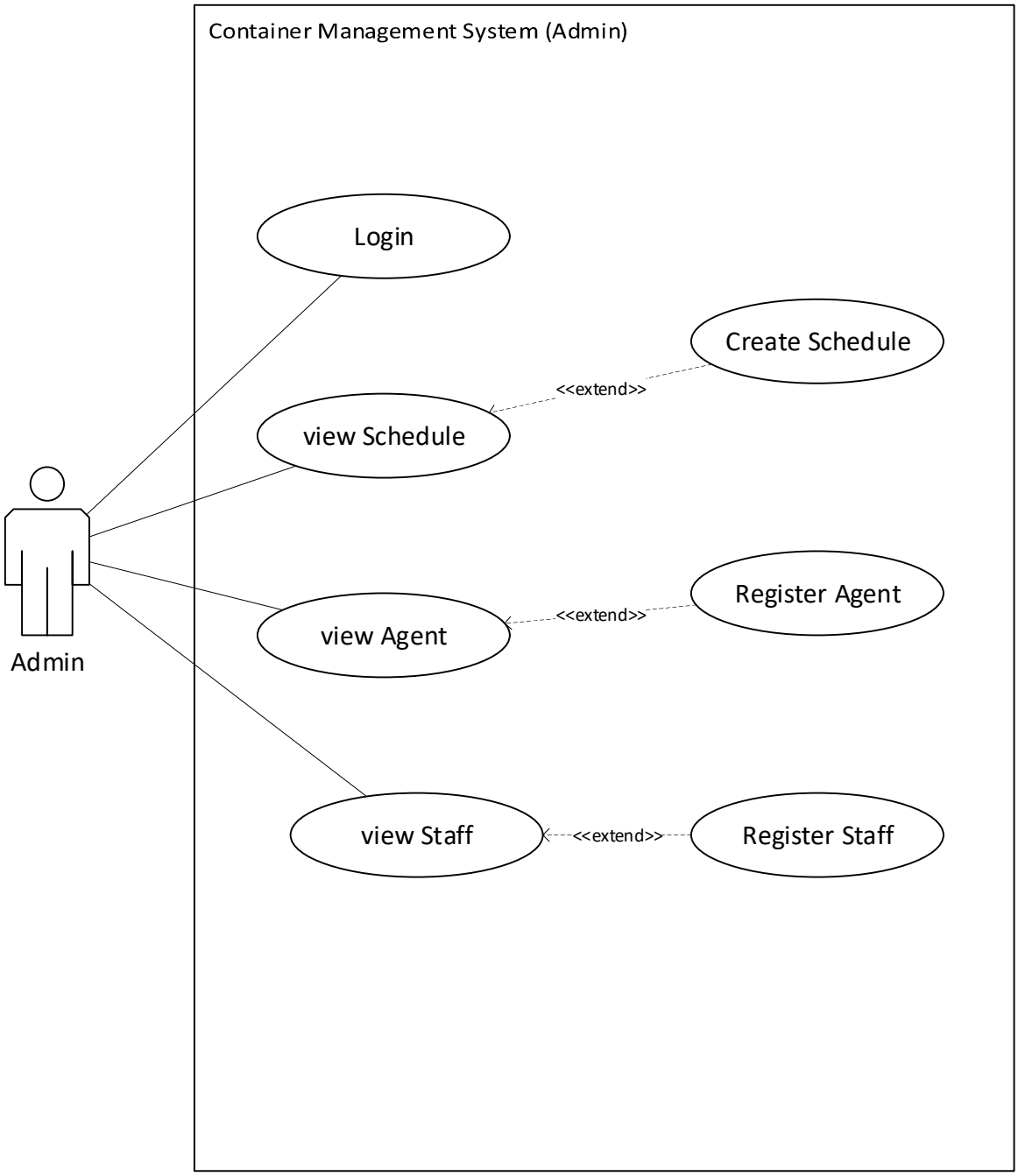


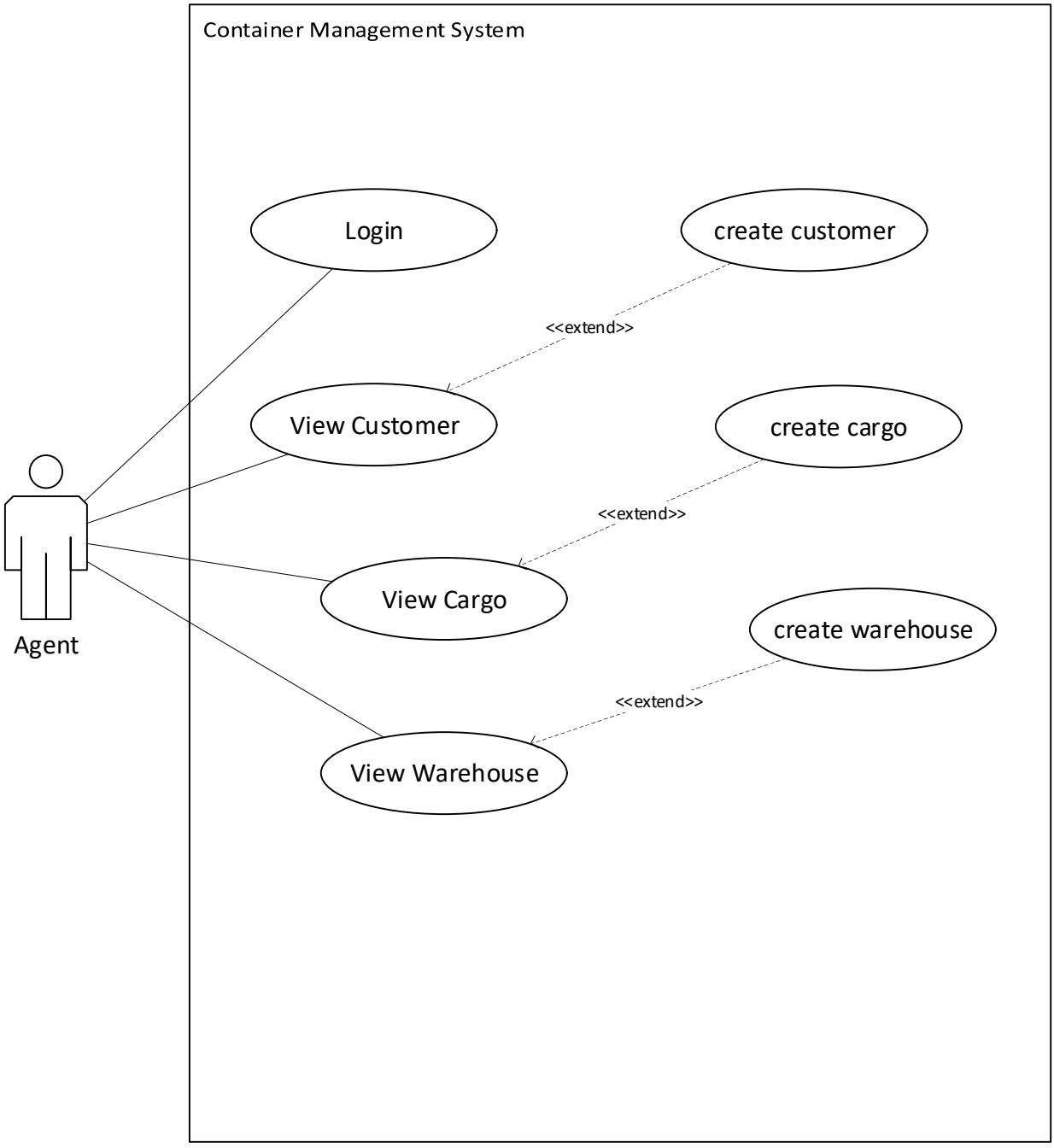
The diagram above is the cloud architecture used for deploying the application to the Azure cloud platform. Maersk Line is looking to create a container management system to manage the containers, a solution that reduces overall supply chain costs and an efficient way to manage logistics. Because the application was written using PHP and MySQL, a MySQL server was provisioned instead of the standard SQL server. The total estimated cost incurred by implementing the architecture above is shown in the table below:

Microsoft Azure Estimate				
Your Estimate				
Service type	Custom name	Region	Description	Estimated Cost
App Service		Southeast Asia	1 instance(s) x 1 Months, Size: B1, Basic tier, 0 SNI connection(s), 0 IP connection(s)	$54.75
Azure Database for MySQL		Southeast Asia	Basic Tier, 1 Gen 4 (2 vCore) x 1 Months, 5 GB Storage, 10 GB Additional Backup storage - LRS redundancy	$77.67
Support			Support	$0.00
			Monthly Total	$132.42
			Annual Total	$1,589.04
Disclaimer				
All prices shown are in US Dollar ($). This is a summary estimate, not a quote. For up to date pricing information please visit https://azure.microsoft.com/pricing/calculator/				
This estimate was created at 4/12/2018 3:55:06 PM UTC.				

3.2 Modelling

3.2.1 Use Case Diagram




Container Management System
Login
create customer
<<extend>>
View Customer
create cargo
<<extend>>
View Cargo
create warehouse
<<extend>>
View Warehouse
Agent

3.2.2 Use case description

Use Case	Login
Actor	Admin, Agent
Summary	Admin and agent log in using their username and password
Precondition	None
Description of main sequence	1. Open the main page 2. Enter credentials 3. Login 4. Enter to the system
Post condition	Logged in to the system as the respective privilege

Use Case	View Schedule
Actor	Admin
Summary	To view all previously created schedule in the system
Precondition	Login as admin, a schedule is created
Description of main sequence	1. Login to the system 2. Click on the schedule tab at the navigation bar 3. All schedule will be displayed in the page
Post condition	The schedule will be displayed on the screen

Use Case	Create Schedule
Actor	Admin
Summary	Admin to create a schedule for agents to book schedule
Precondition	Login to the system as an admin
Description of main sequence	1. Login into the system as an admin 2. Click the shipping button at the navigation bar 3. Enter the necessary Details 4. Click Create button at the bottom of page 5. Schedule created
Post condition	Schedule created in the system database

Use Case	View Agent
Actor	Admin, agent
Summary	To view all the agent that are in the system
Precondition	Login to the system
Description of main sequence	1. Login to the system as an admin 2. Click the “agent” button at the navigation bar 3. The agents will be displayed in the screen
Post condition	Agent displayed on the screen

Use Case	Register Agent
Actor	Admin
Summary	Admin to be able to create an agent for the container management system
Precondition	Login to the system as an Admin
Description of main sequence	1. Login to the system as an admin 2. Click the “agent” button at the navigation bar 3. Click on the Create button at the bottom of the page 4. Enter the details required 5. Click on the “Submit” button 6. Agent is created for the system
Post condition	Agent is created in the system database

Use Case	View Staff
Actor	Admin
Summary	To view all the staff that are in the system
Precondition	Login as admin
Description of main sequence	1. Login to the system as an admin 2. Click the “staff” button at the navigation bar 3. The staff will be displayed in the screen
Post condition	All staff displayed on the screen in table form

Use Case	Register Staff
Actor	Admin
Summary	Admin to be able to create an staff for the container management system
Precondition	Login to the system as an Admin
Description of main sequence	1. Login to the system as an admin 2. Click the “staff” button at the navigation bar 3. Click on the Create button at the bottom of the page 4. Enter the details required 5. Click on the “Submit” button 6. Staff is created for the system
Post condition	The staff is created in the system database

Use Case	View customer
Actor	Agent
Summary	To view all the customer that are in the system
Precondition	Login as agent
Description of main sequence	1. Login to the system as an agent 2. Click the “customer” button at the navigation bar 3. The customer will be displayed in the screen
Post condition	All customers will be displayed on the screen in table form

Use Case	Create customer
Actor	Agent
Summary	Agent to be able to create a new customer for the container management system
Precondition	Login to the system as an Agent
Description of main sequence	1. Login to the system as an agent 2. Click the “customer” button at the navigation bar 3. Click on the Create button at the bottom of the page 4. Enter the details required 5. Click on the “Submit” button 6. Customer is created for the system
Post condition	The customer is created in the system database

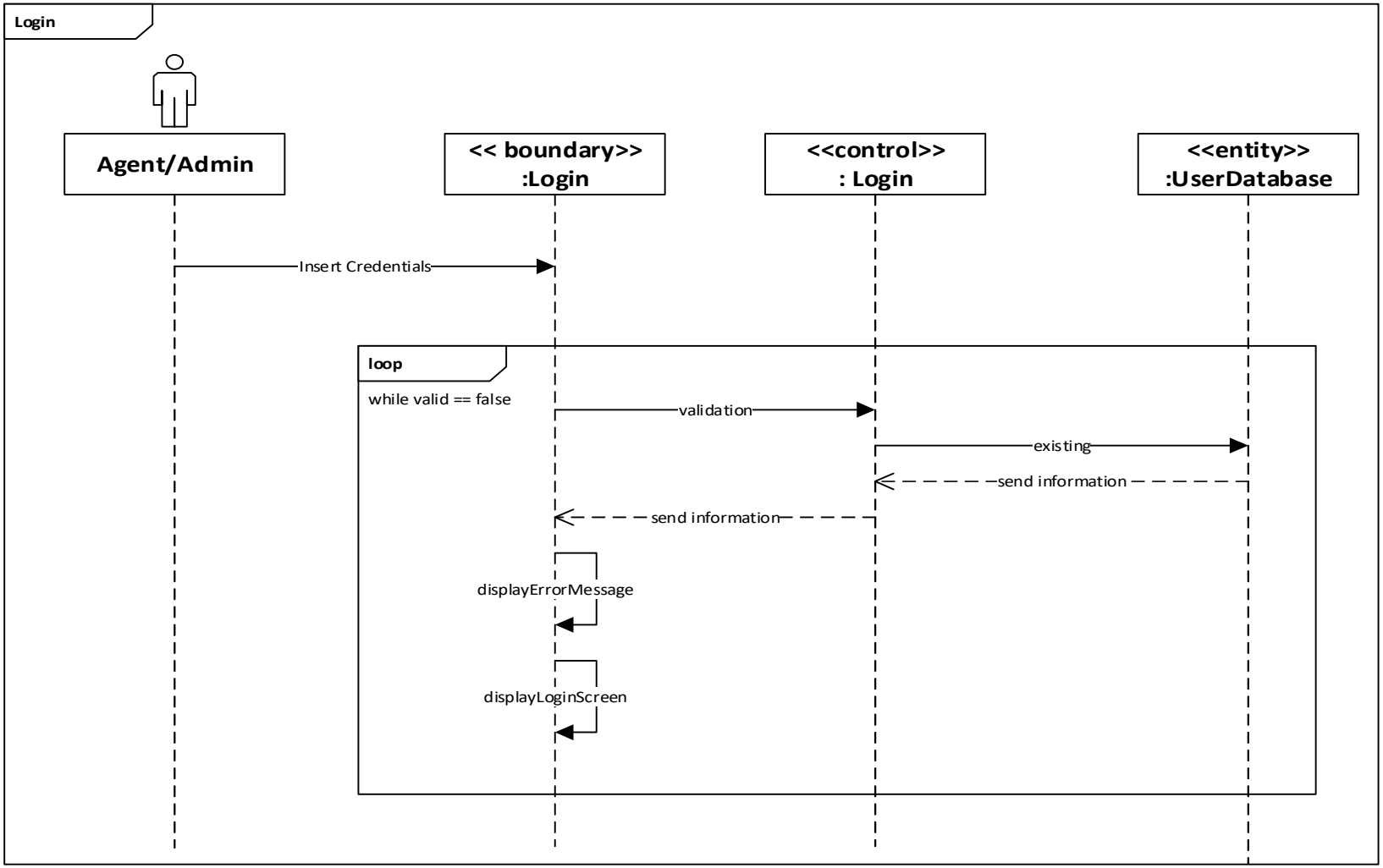
Use Case	View cargo
Actor	Agent
Summary	To view all the customer that are in the system
Precondition	Login as agent
Description of main sequence	1. Login to the system as an agent 2. Click the “cargo” button at the navigation bar 3. The cargo will be displayed in the screen
Post condition	All cargo will be displayed on the screen in table form

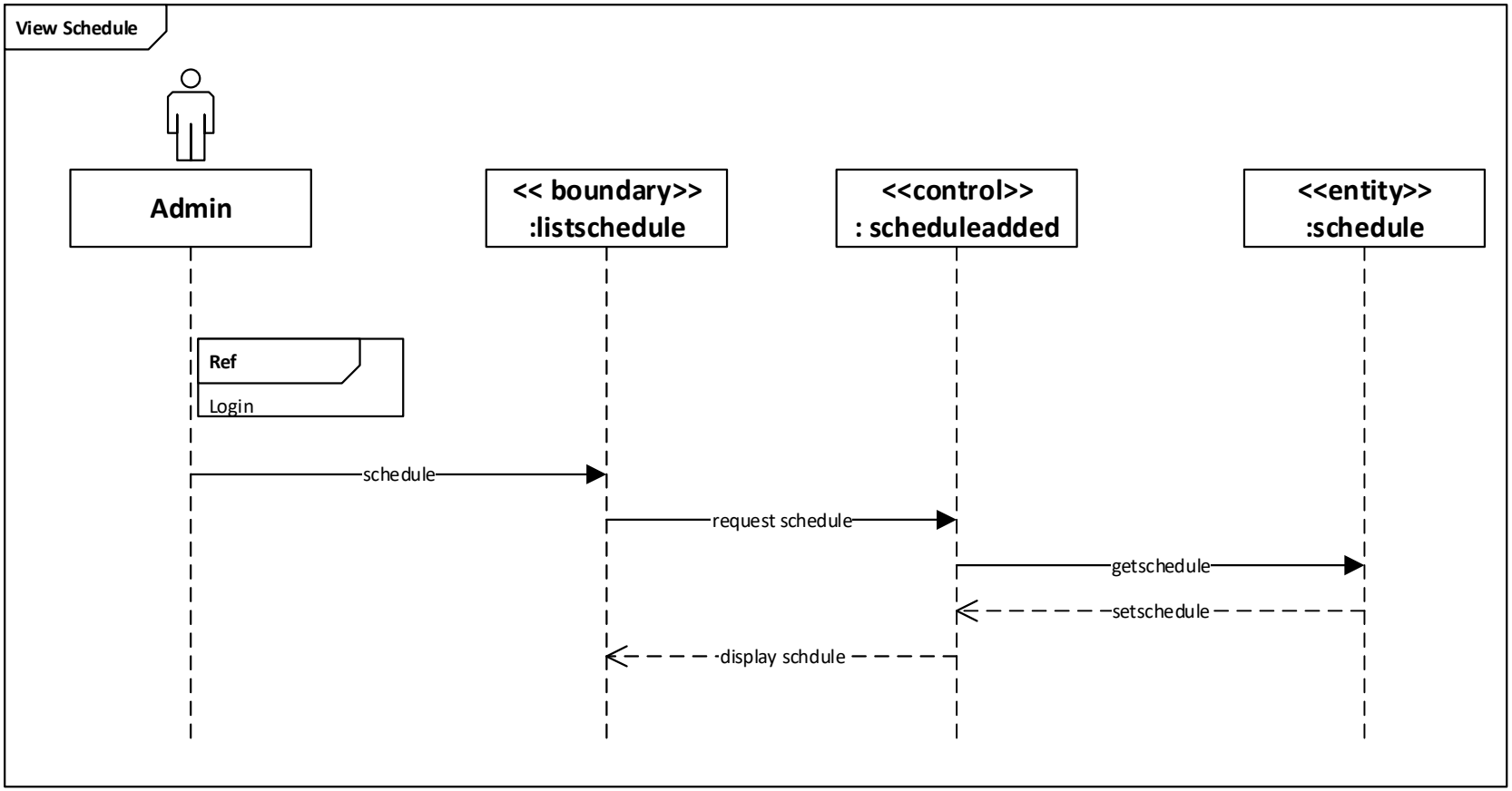
Use Case	Create cargo
Actor	Agent
Summary	Agent to be able to create a new cargo for the container management system
Precondition	Login to the system as an Agent
Description of main sequence	1. Login to the system as an agent 2. Click the “cargo” button at the navigation bar 3. Click on the Create button at the bottom of the page 4. Enter the details required 5. Click on the “Submit” button 6. Cargo is created for the system
Post condition	The cargo is created in the system database

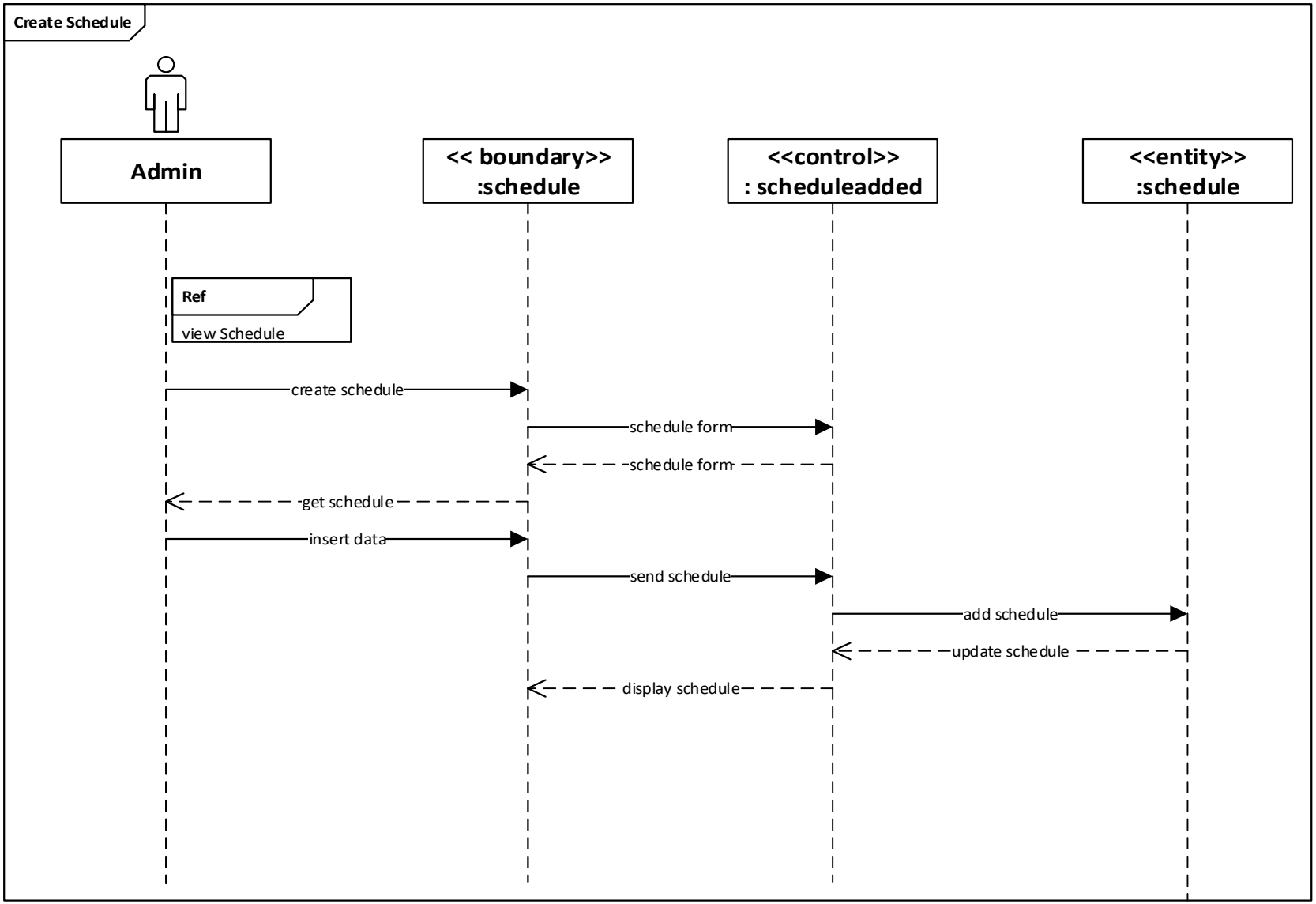
Use Case	View warehouse
Actor	Agent
Summary	To view all the customer that are in the system
Precondition	Login as agent
Description of main sequence	1. Login to the system as an agent 2. Click the “warehouse” button at the navigation bar 3. The warehouse list will be displayed in the screen
Post condition	All the warehouses will be displayed on the screen in table form

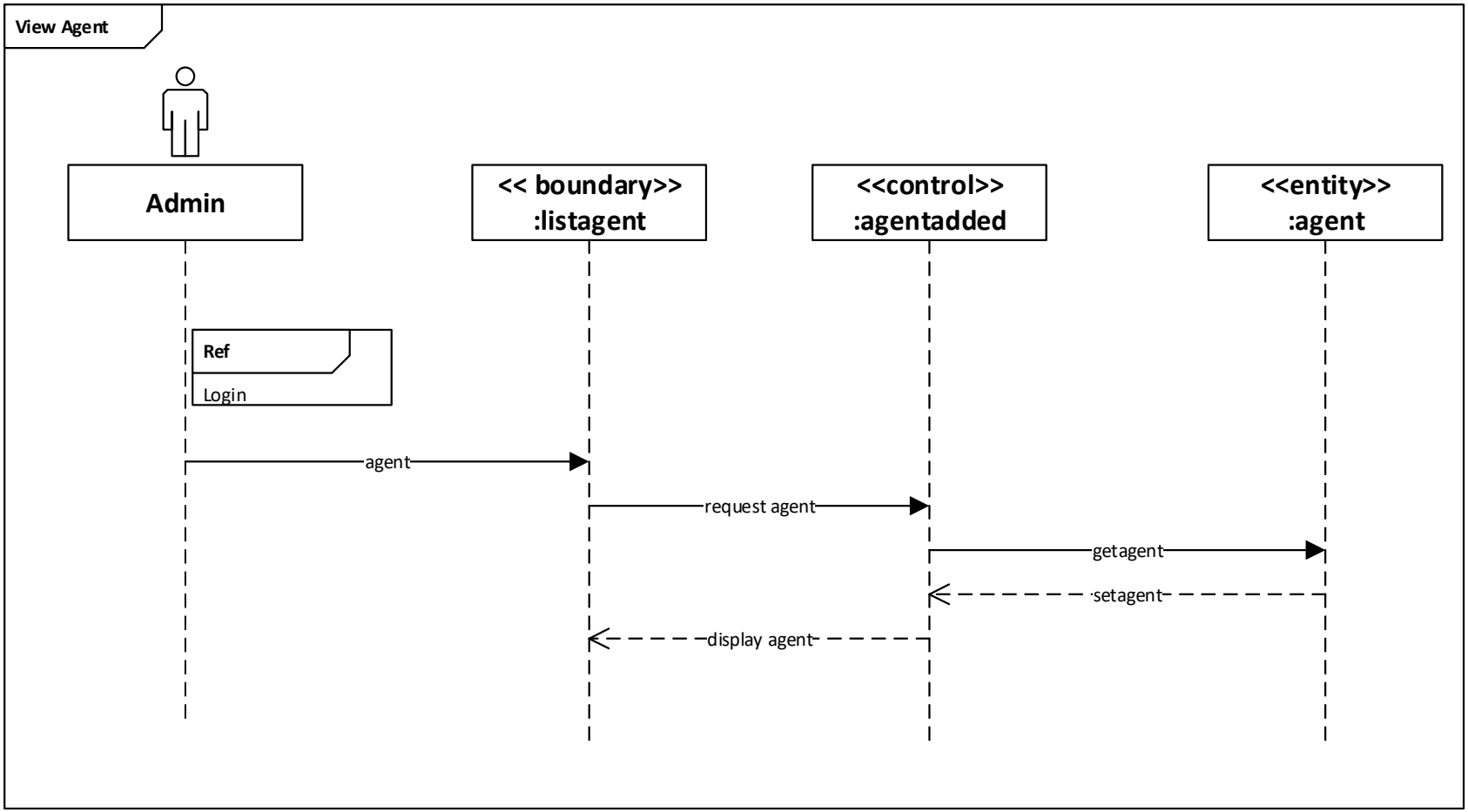
Use Case	Create warehouse
Actor	Agent
Summary	Agent to be able to register a new warehouse r the container management system
Precondition	Login to the system as an Agent
Description of main sequence	1. Login to the system as an agent 2. Click the “warehouse” button at the navigation bar 3. Click on the Create button at the bottom of the page 4. Enter the details required 5. Click on the “Submit” button 6. Warehouse is created for the system
Post condition	The warehouse is created in the system database

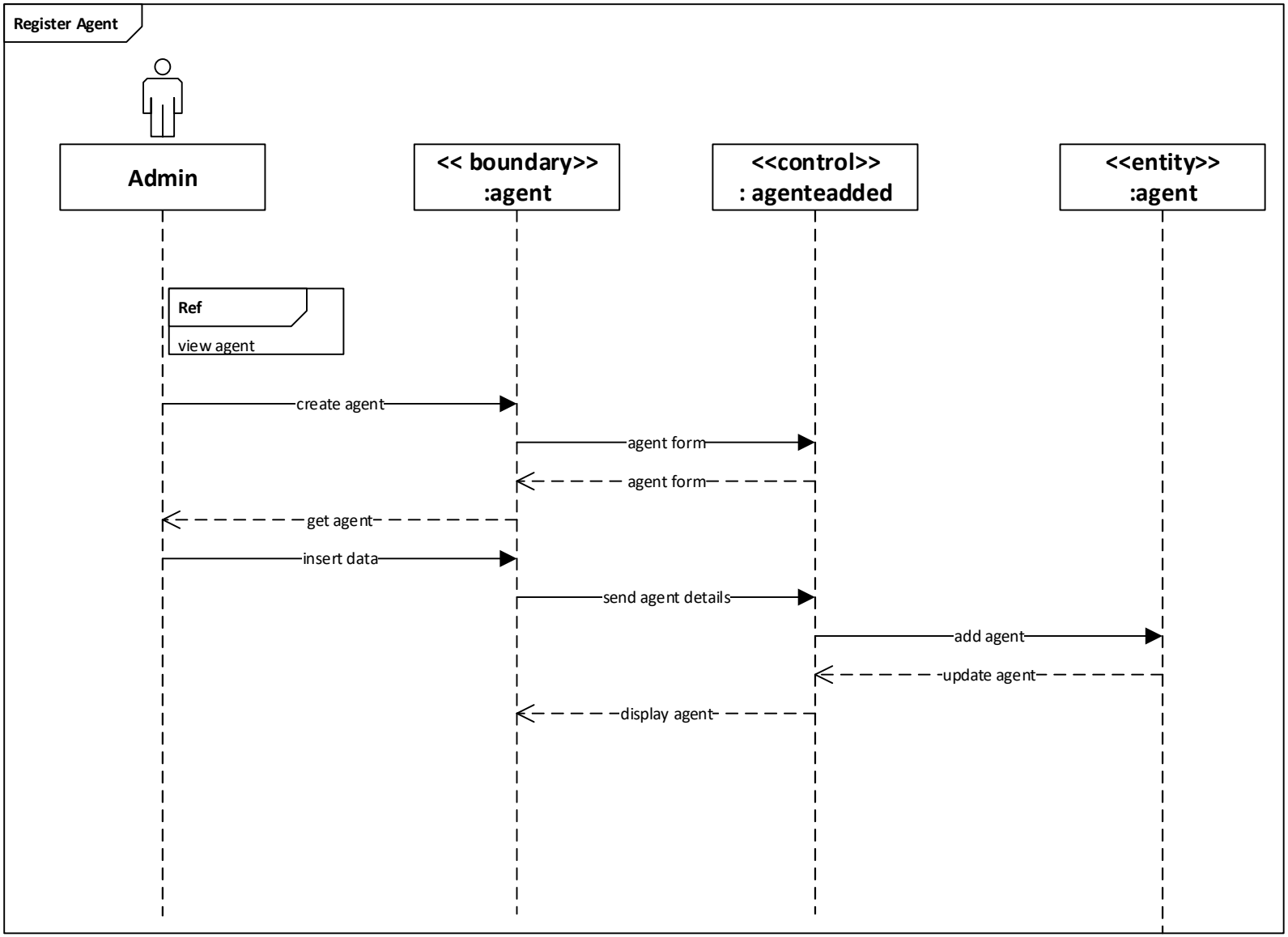
3.2.3 Sequence Diagram

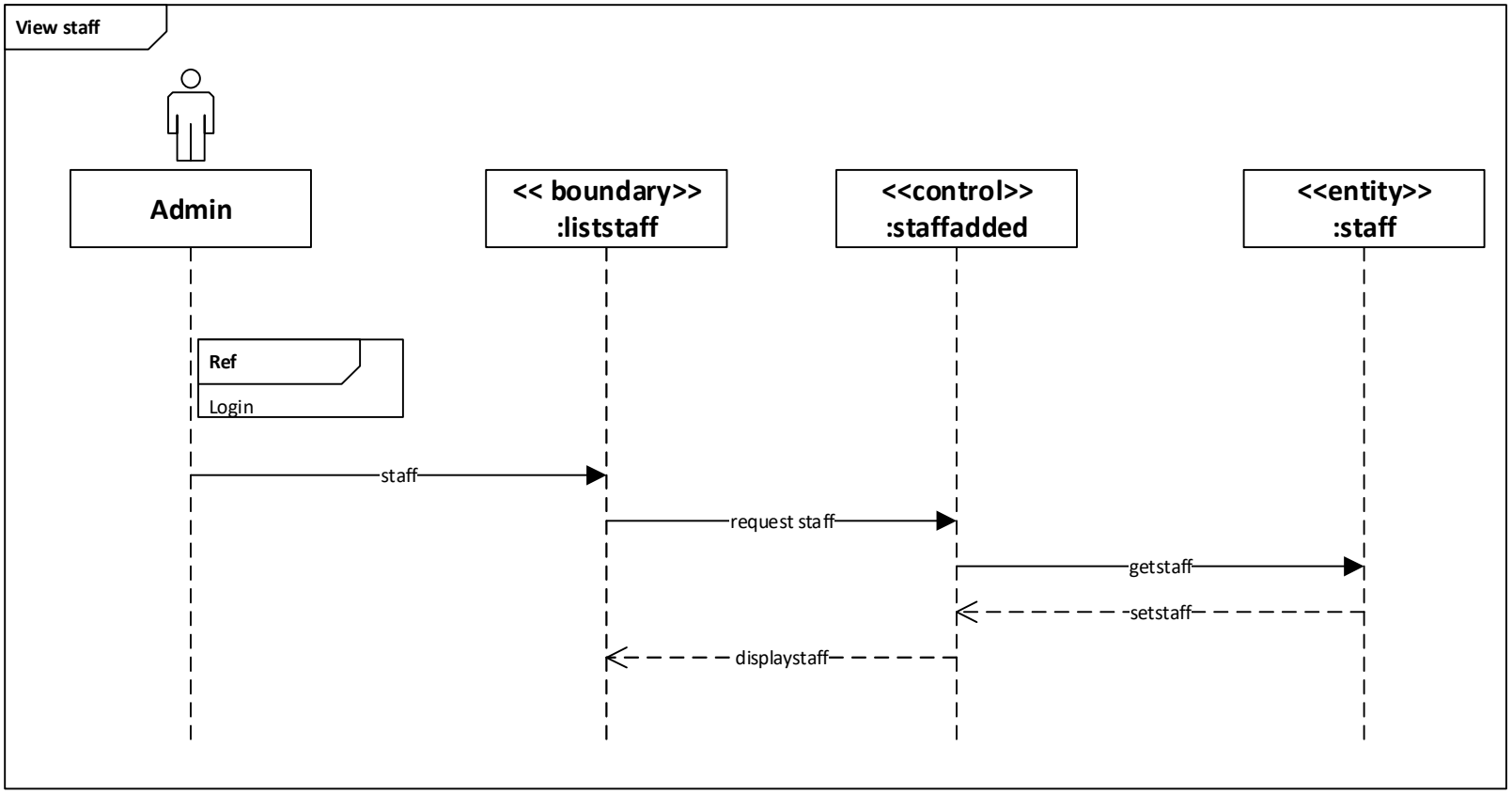


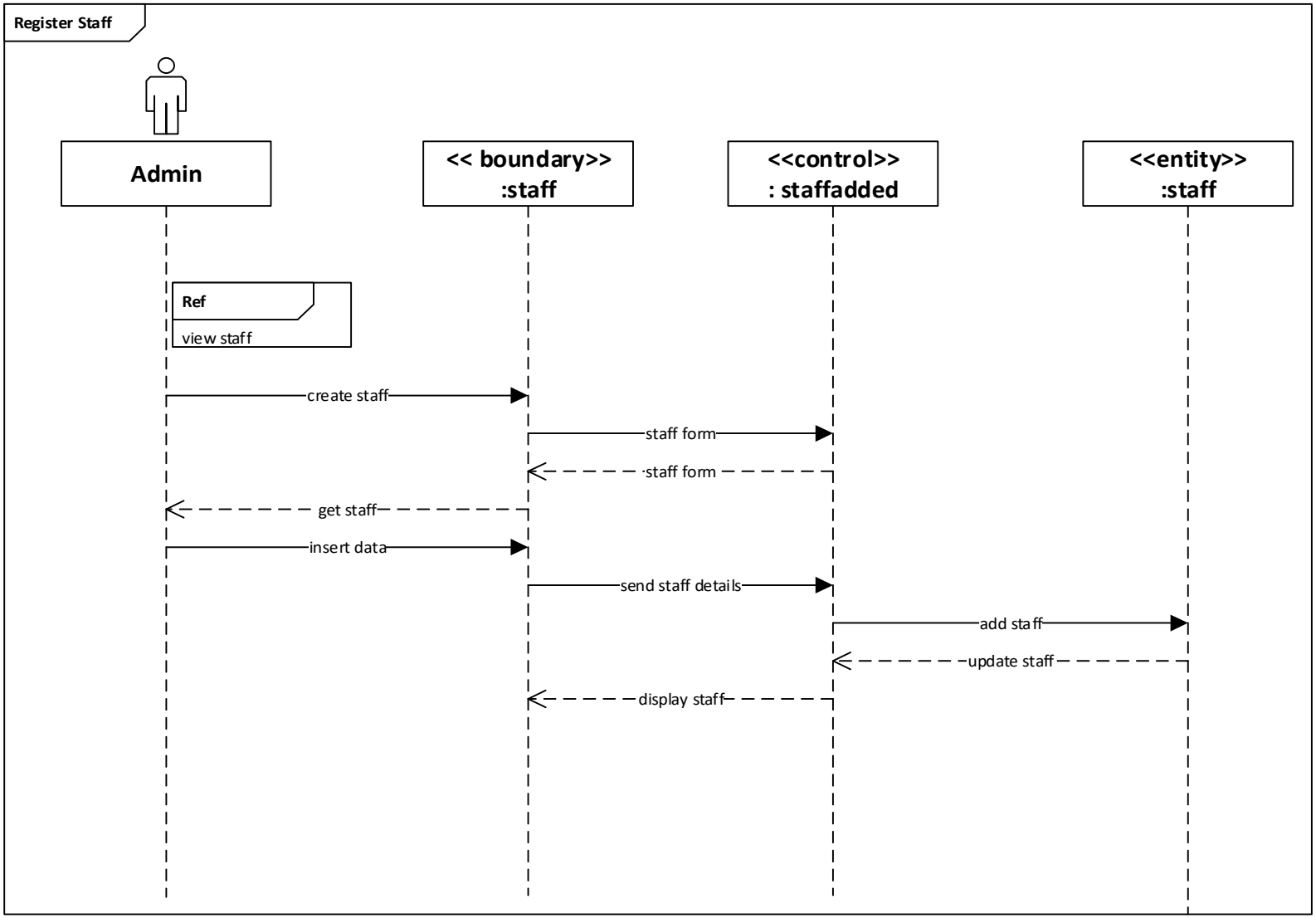

View Schedule
Admin
<< boundary>>
:listschedule
<<control>>
: scheduleadded
<<entity>>
:schedule
Ref
Login
schedule
request schedule
getschedule
setschedule
display schdule

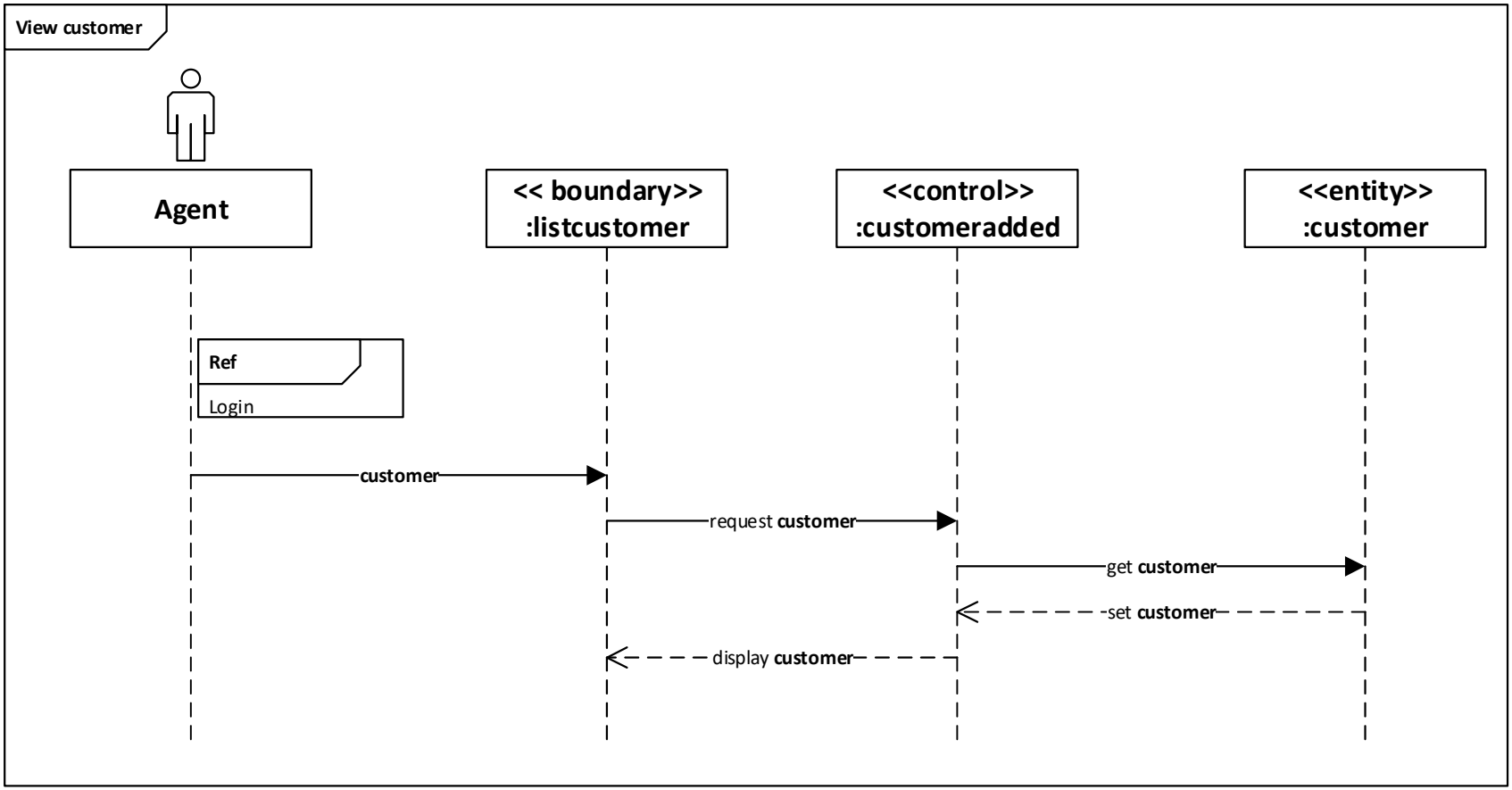

Create Schedule
Admin
<< boundary>>
:schedule
<<control>>
: scheduleadded
<<entity>>
:schedule
Ref
view Schedule
create schedule
schedule form
schedule form
get schedule
insert data
send schedule
add schedule
update schedule
display schedule

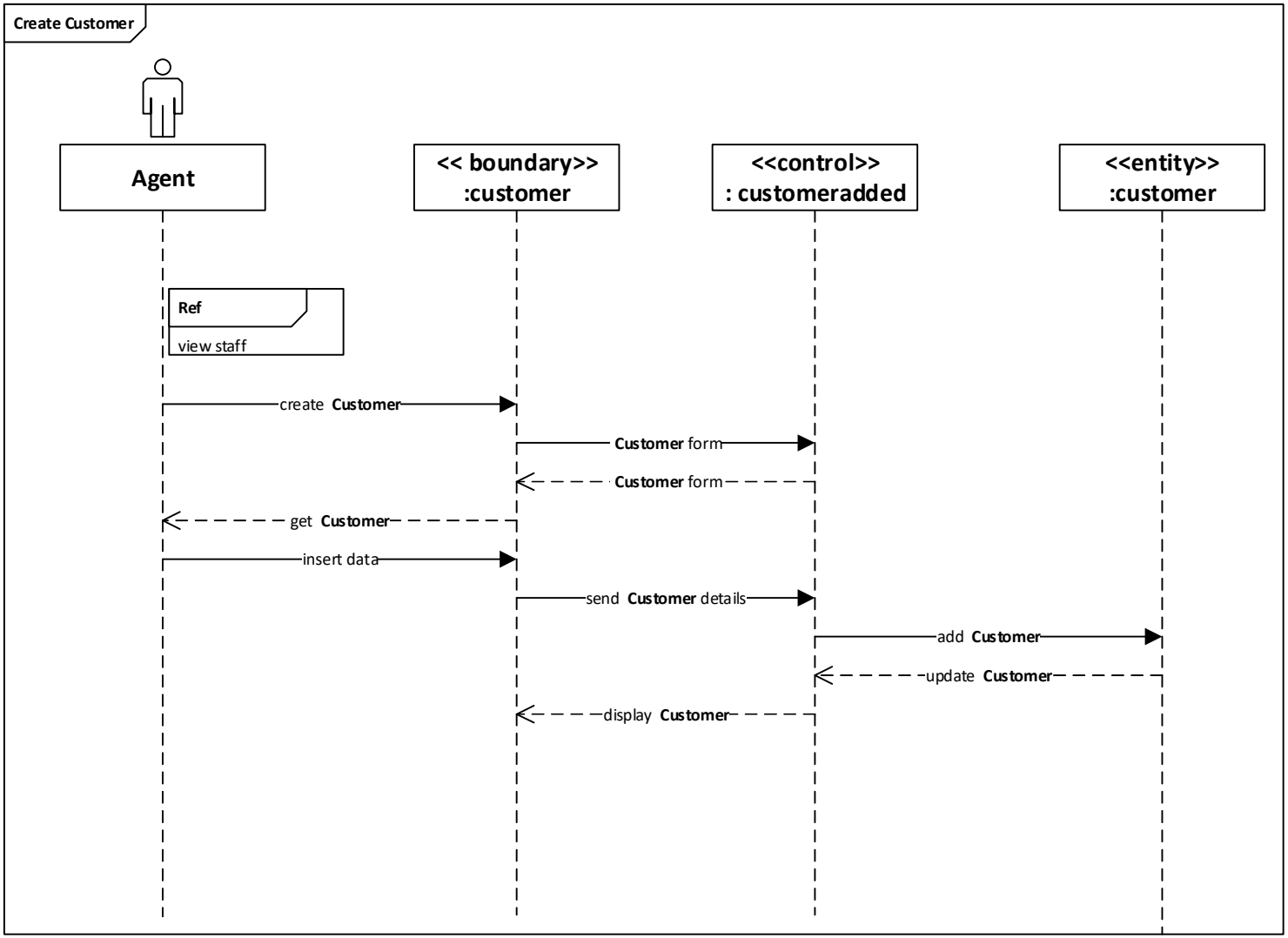

View Agent
Admin
<< boundary>>
:listagent
<<control>>
:agentadded
<<entity>>
:agent
Ref
Login
agent
request agent
getagent
setagent
display agent

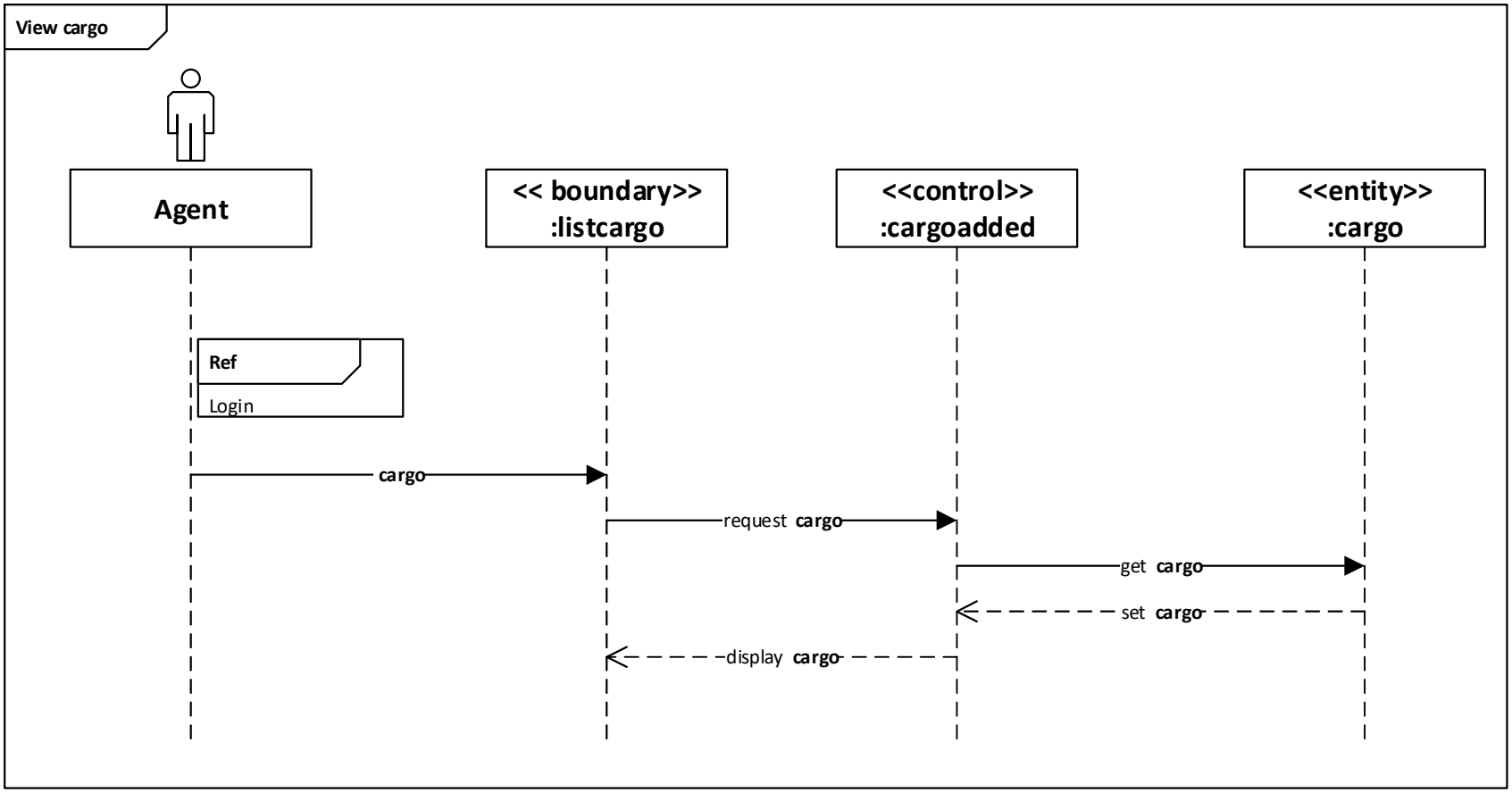

Register Agent
Admin
<< boundary>>
:agent
<<control>>
: agenteadded
<<entity>>
:agent
Ref
view agent
create agent
agent form
agent form
get agent
insert data
send agent details
add agent
update agent
display agent

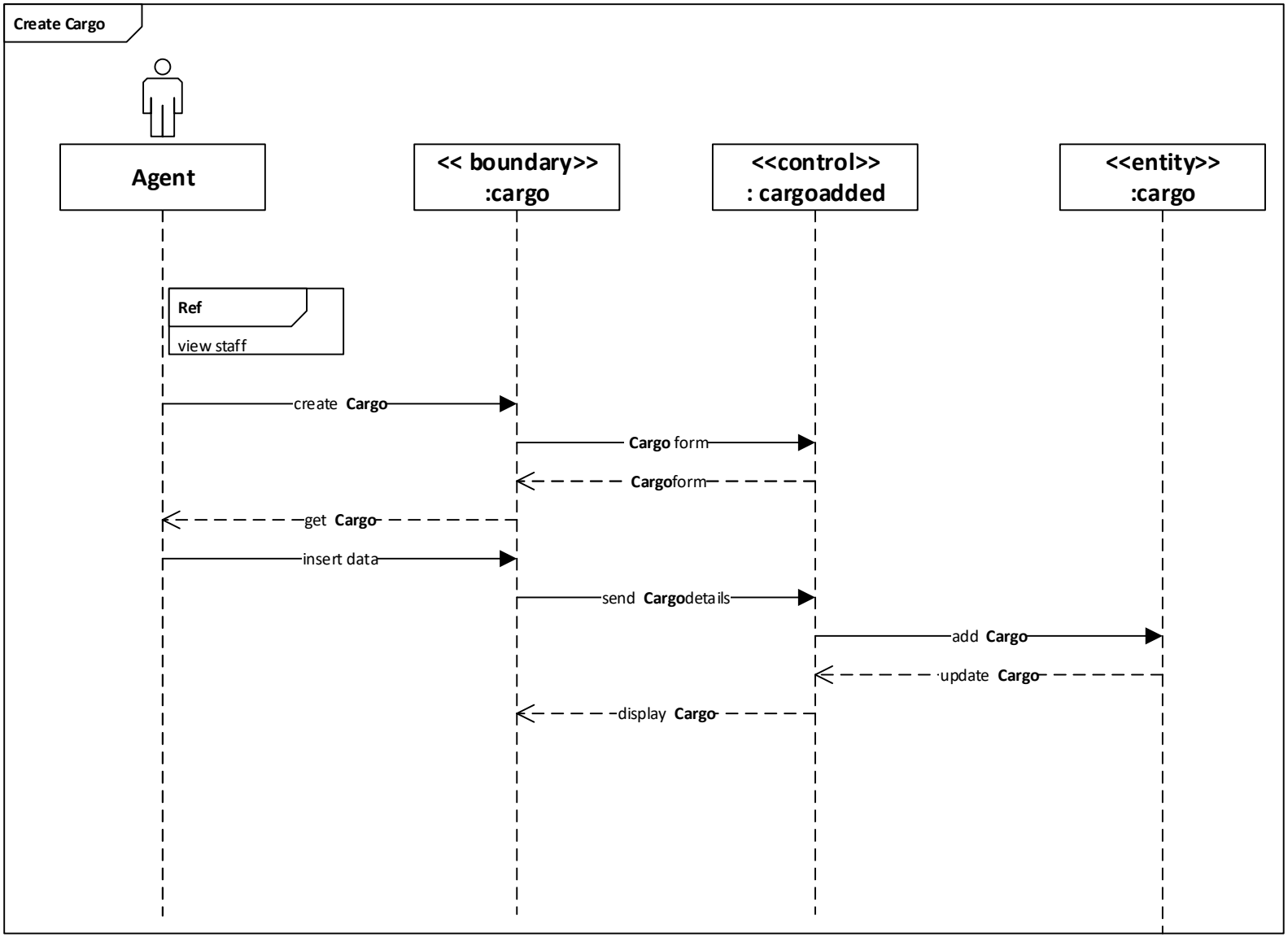

View staff
Admin
<< boundary>>
:liststaff
<<control>>
:staffadded
<<entity>>
:staff
Ref
Login
staff
request staff
getstaff
setstaff
displaystaff

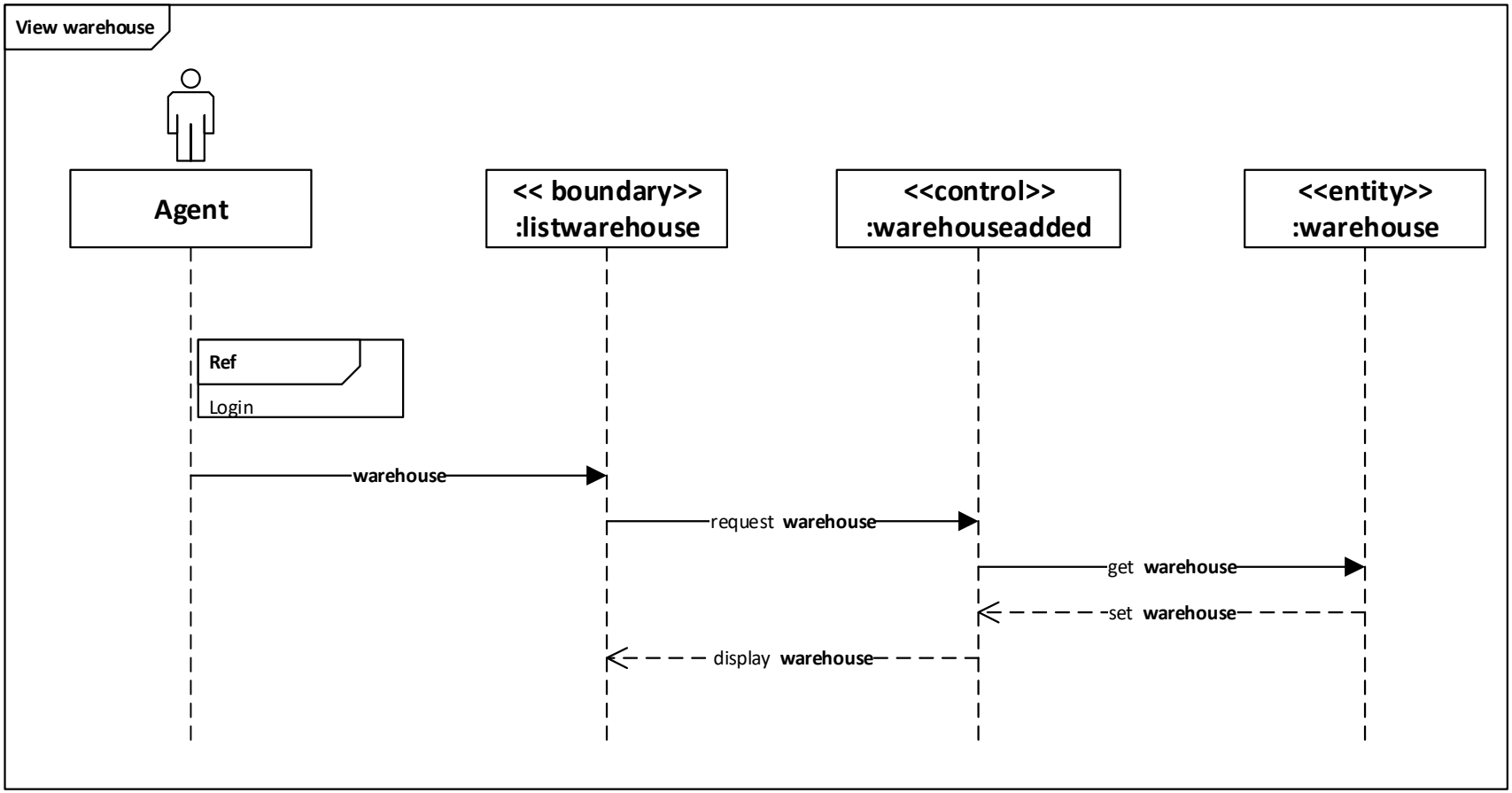

Register Staff
Admin
<< boundary>>
:staff
<<control>>
: staffadded
<<entity>>
:staff
Ref
view staff
create staff
staff form
staff form
get staff
insert data
send staff details
add staff
update staff
display staff

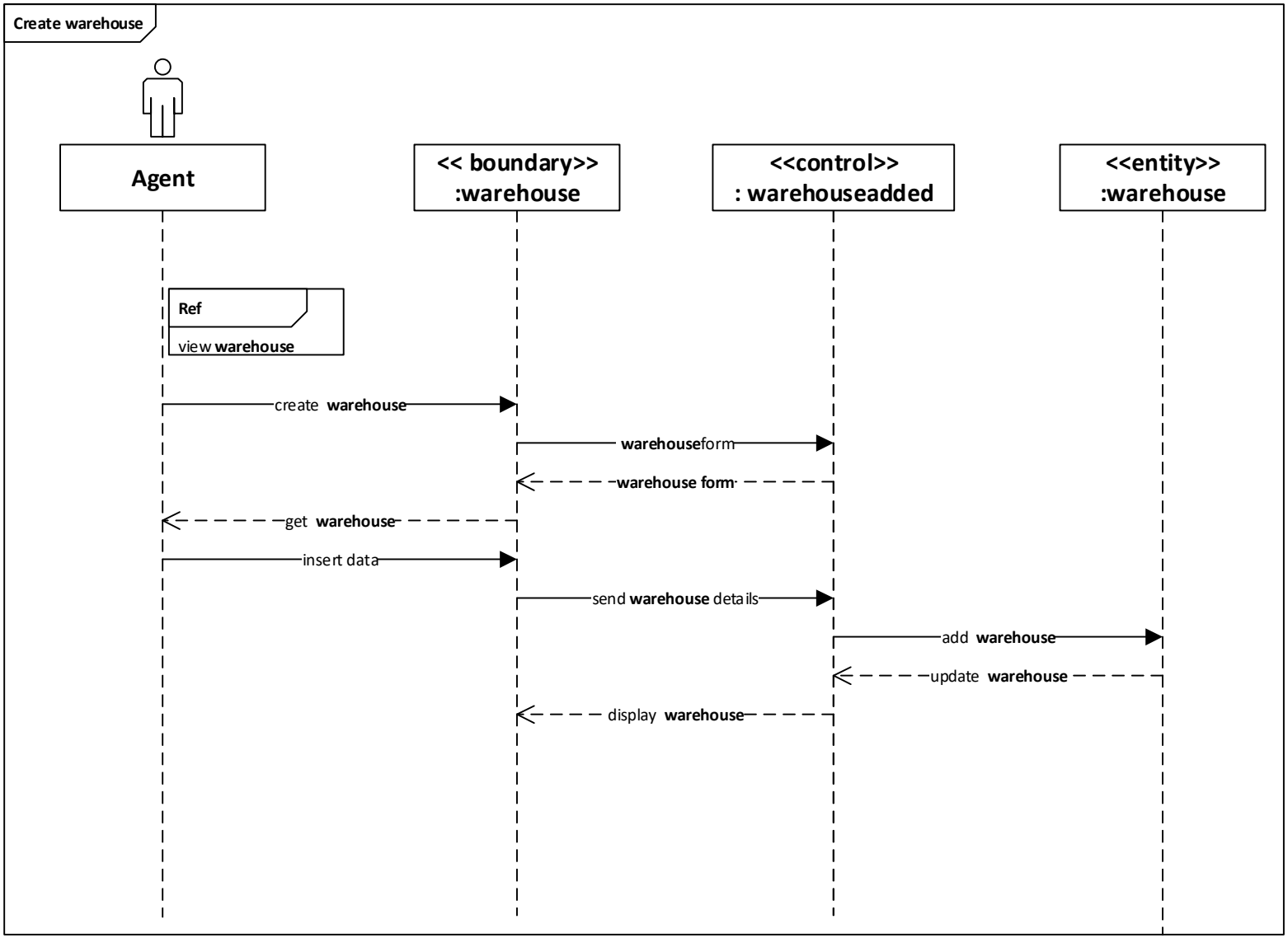

View customer
Agent
<< boundary>>
:listcustomer
<<control>>
:customeradded
<<entity>>
:customer
Ref
Login
customer
request customer
get customer
set customer
display customer


Create Customer
Agent
<< boundary>>
:customer
<<control>>
: customeradded
<<entity>>
:customer
Ref
view staff
create Customer
Customer form
Customer form
get Customer
insert data
send Customer details
add Customer
update Customer
display Customer


View cargo
Agent
<< boundary>>
:listcargo
<<control>>
:cargoadded
<<entity>>
:cargo
Ref
Login
cargo
request cargo
get cargo
set cargo
display cargo


Create Cargo
Agent
<< boundary>>
:cargo
<<control>>
: cargoadded
<<entity>>
:cargo
Ref
view staff
create Cargo
Cargo form
Cargoform
get Cargo
insert data
send Cargodetails
add Cargo
update Cargo
display Cargo

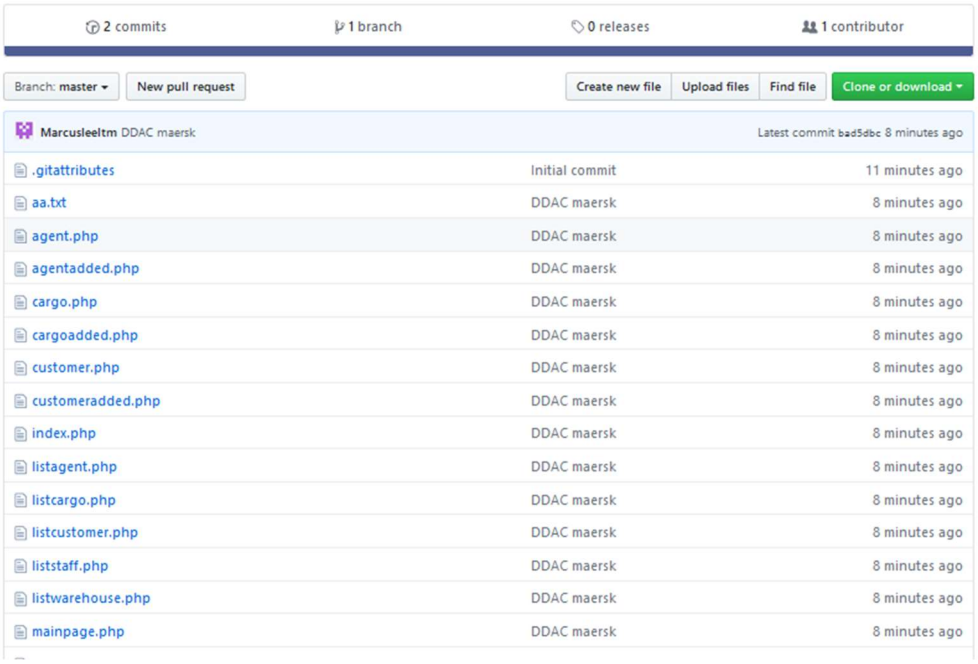

View warehouse
Agent
<< boundary>>
:listwarehouse
<<control>>
:warehouseadded
<<entity>>
:warehouse
Ref
Login
warehouse
request warehouse
get warehouse
set warehouse
display warehouse


Create warehouse
Agent
<< boundary>>
:warehouse
<<control>>
: warehouseadded
<<entity>>
:warehouse
Ref
view warehouse
create warehouse
warehouseform
warehouse form
get warehouse
insert data
send warehouse details
add warehouse
update warehouse
display warehouse

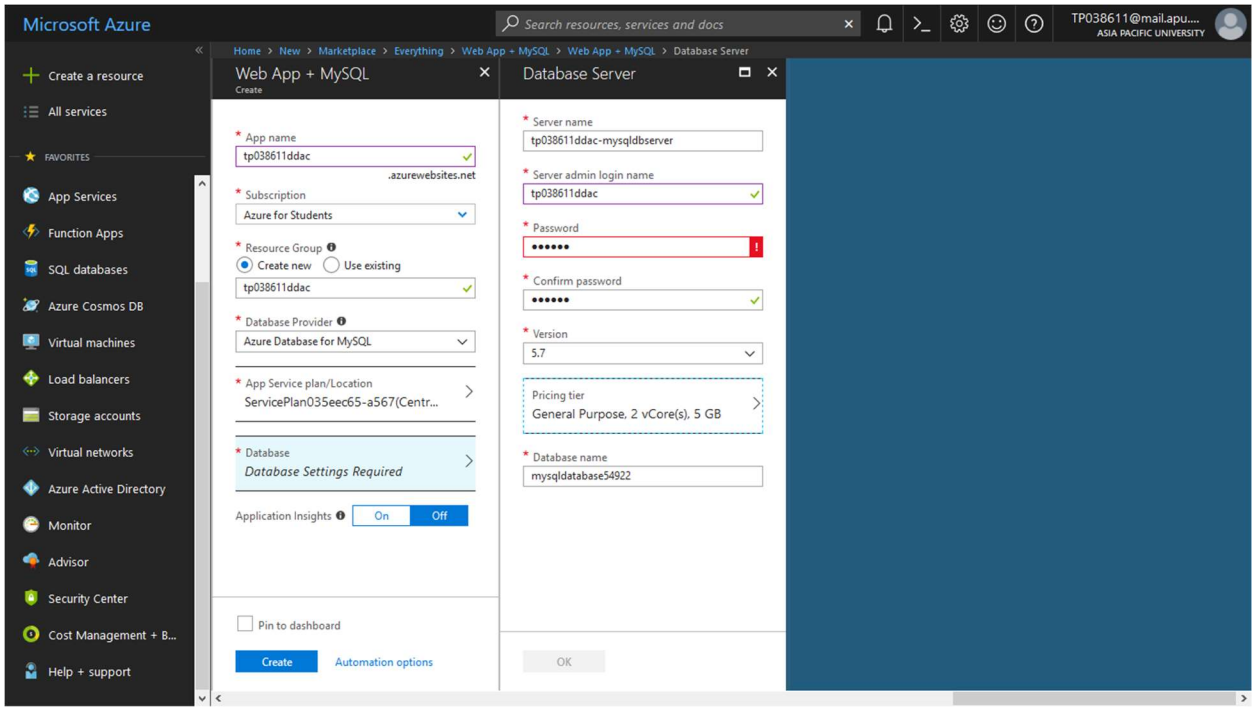
4.0 Implementation

4.1 Deploy on Azure

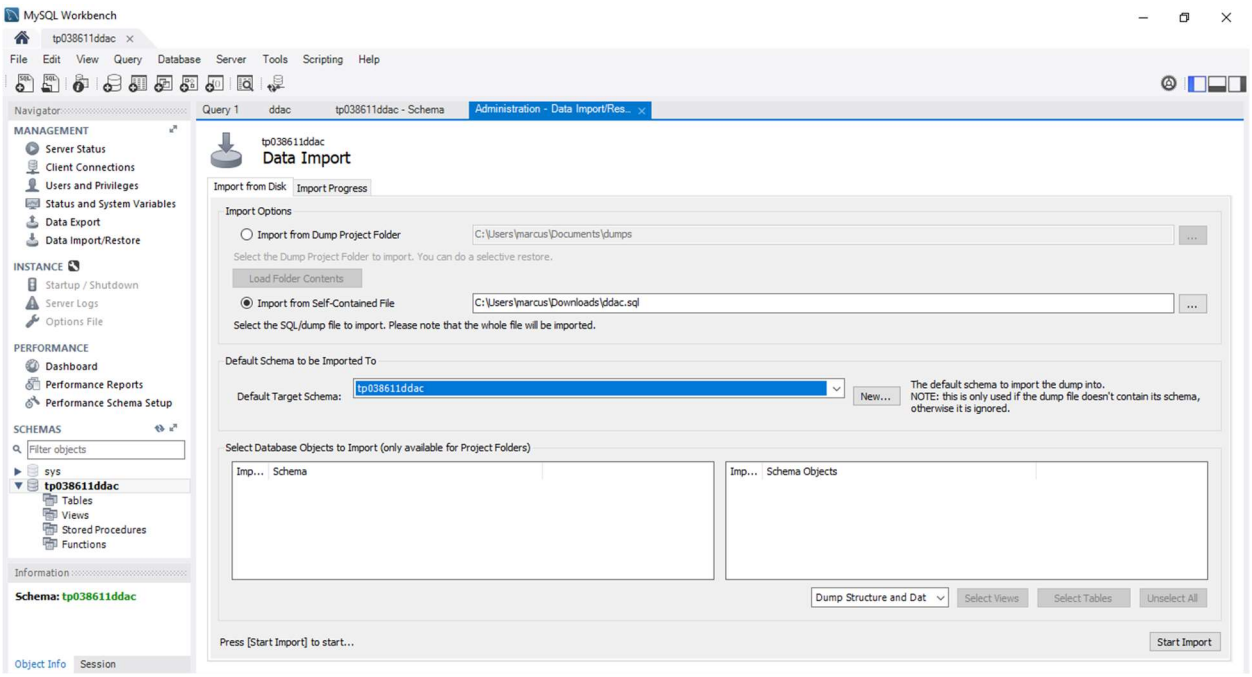
The entire system is firstly published into GitHub and then pushed into Microsoft Azure for automatic deployment. This will allow any changes that will be made into GitHub and will also be deployed directly into the Azure server.



The appropriate Web application service is selected on Azure. In the case of this deployment and development of the container management system, the most suitable will be a web app + ‘SQL since the application uses PHP and MySQL to develop.

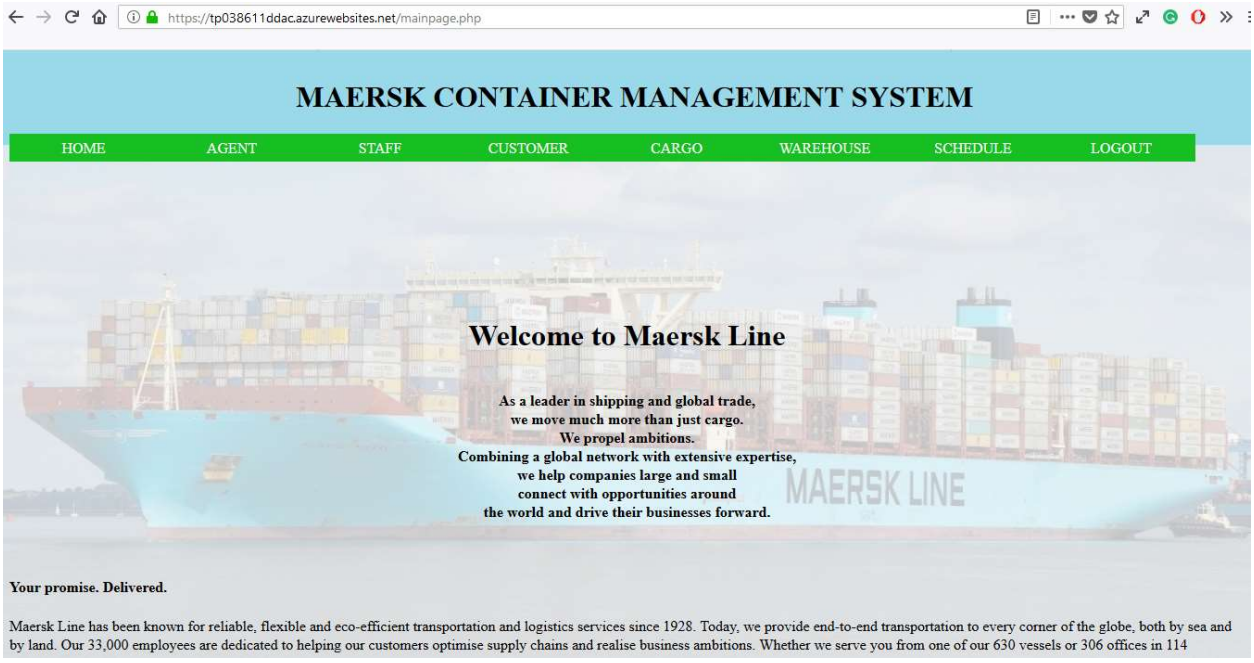


Wait for a few minutes as Azure submit the deployment for Web App + SQL and create the platform for the Container Management system. While waiting MySQL server can be upload into Azure using the help of a software MySQL Workbench. The hostname, username and password could be taken from Azure portal in the database service. After creating the connection with Azure server, the MySQL query then can be uploaded into the platform as well.



After completing the both the Azure Web app service and the MySQL server merging, the system can be said to be successfully deployed into the cloud and could be found in the following link:

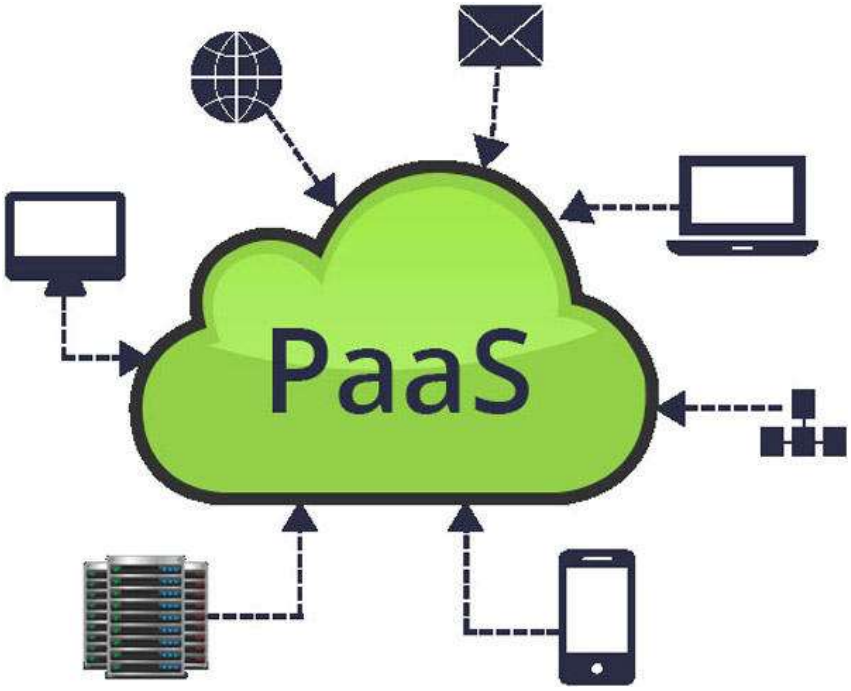
https://tp038611ddac.azurewebsites.net/mainpage.php



4.2 Application Scaling

According to (Azure.microsoft.com, 2018), Azure cloud environment provides the web service for auto scaling which the resource can be dynamically allocated based on the needs to match with performance requirement. This is particularly important when Maersk's Container Management System increase in workload, more resource is needed to maintain the desired performance levels. The scaling to add extra resource is nothing to be worried about as during high peak more resource can be allocated automatically and deallocated when no longer needed. This can help Maersk to minimize the operation cost.

4.3 Manage Database (PAAS)



Platform as a Service (PaaS) is a cloud platform service. Platform as a service (PaaS) is a complete development and deployment environment in the cloud, with resources that enable you to deliver everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applications. You purchase the resources you need from a cloud service provider on a pay-as-you-go basis and access them over a secure Internet connection. (Azure.microsoft.com, 2018)

It is used for applications and other development, while providing cloud components to software. Developers can use this framework and build upon to develop or customize application. (Apprenda,2018) PaaS can be delivered through public, private, or hybrid clouds. With a public cloud PaaS, the customer controls software deployment while the cloud provider delivers all the major IT components needed to host the applications, including servers, storage systems, networks, operating systems, and databases. (Violino, 2018)

A PaaS provider hosts the hardware and software on its own infrastructure. As a result, PaaS frees users from having to install in-house hardware and software to develop or run a new application. As with most cloud services, PaaS is built on virtualization technology. (SearchCloudComputing, 2018)

Having a platform for the creation of on-demand applications or software delivered over the web bring many different benefits to different parties including the developer, the business and the company. It makes the development, testing and deployment of applications quick simple and cost-effective since the development team can try different configuration, multiple machines and different locations to run test and access performance.

Beside it will reduce the need to expend effort on the maintenance and choice system, giving business and companies to focus more on their core business rather than the technicalities of the development processes. Hence increasing and empowering visionaries and give internal entrepreneurship a boost since the business had set aside a specific sum under the cloud budget.

Other benefits include staying up to date with the latest deployment the soonest, maximizing uptime as vendors have the tools and experience to handle unplanned risk that occurs. Scalability is also very simple and easy with the help of PaaS where the system can be scaled according to the given size of business. Throughout the years, security had been improving and so many vendors offer a great deal of security for your system.

There are many different cloud providers in the market, with the main providers from tech giants such as Google, Amazon and Microsoft. Amazon cloud services is known as Amazon Web Services (AWS), Microsoft call its cloud service Microsoft Azure and Google's cloud services is known as Google Cloud Platform.

AWS, Microsoft Azure and Google Cloud Platform offer largely similar basic capabilities around flexible compute, storage and networking. They all share the common elements of a public cloud: self-service and instant provisioning, autoscaling, plus security, compliance and identity management features.

5.0 Test Plan & Testing Discussion

5.1 Unit Testing

Test ID	Test Case	Test Case Description	Expected Result	Actual Result	Status
T1	Login	1. Enter the credentials as an Admin 2. Click on the login "button" 3. Navigate to the main page with admin privileges	Successfully login to the system as an admin	Successfully login to the system as an admin	Pass
T2	Login	1. Enter the credentials as an agent 2. Click on the login "button" 3. Navigate to the main page with agent privileges	Successfully login to the system as an agent	Successfully login to the system as an agent	Pass
ADMIN					
TAD1	View Agent	1. Login as an admin 2. Click on agent at the navigation bar 3. List of agents will be displayed in a table in the screen	The table of agents is displayed	The table of agents is displayed	Pass
TAD2	Register Agent	1. Login as an admin 2. Click on agent at the navigation bar 3. Click on create button	Agent is added to the database	Agent is added to the database	Pass

		4. Enter the details necessary 5. Click on the submit button 6. Navigate back to the agent page 7. Agent will be added into the table			
TAD3	View Staff	1. Login as an admin 2. Click on staff at the navigation bar 3. List of staff will be displayed in a table in the screen	The table of staff is displayed	The table of staff is displayed	Pass
TAD4	Create Staff	1. Login as an admin 2. Click on staff at the navigation bar 3. Click on create button 4. Enter the details necessary 5. Click on the submit button 6. Navigate back to the staff page 7. Staff will be added into the table	Staff is added to the database	Staff is added to the database	Pass
TAD5	View Schedule	1. Login as an admin 2. Click on schedule at the navigation bar	The table of schedule is displayed	The table of schedule is displayed	Pass

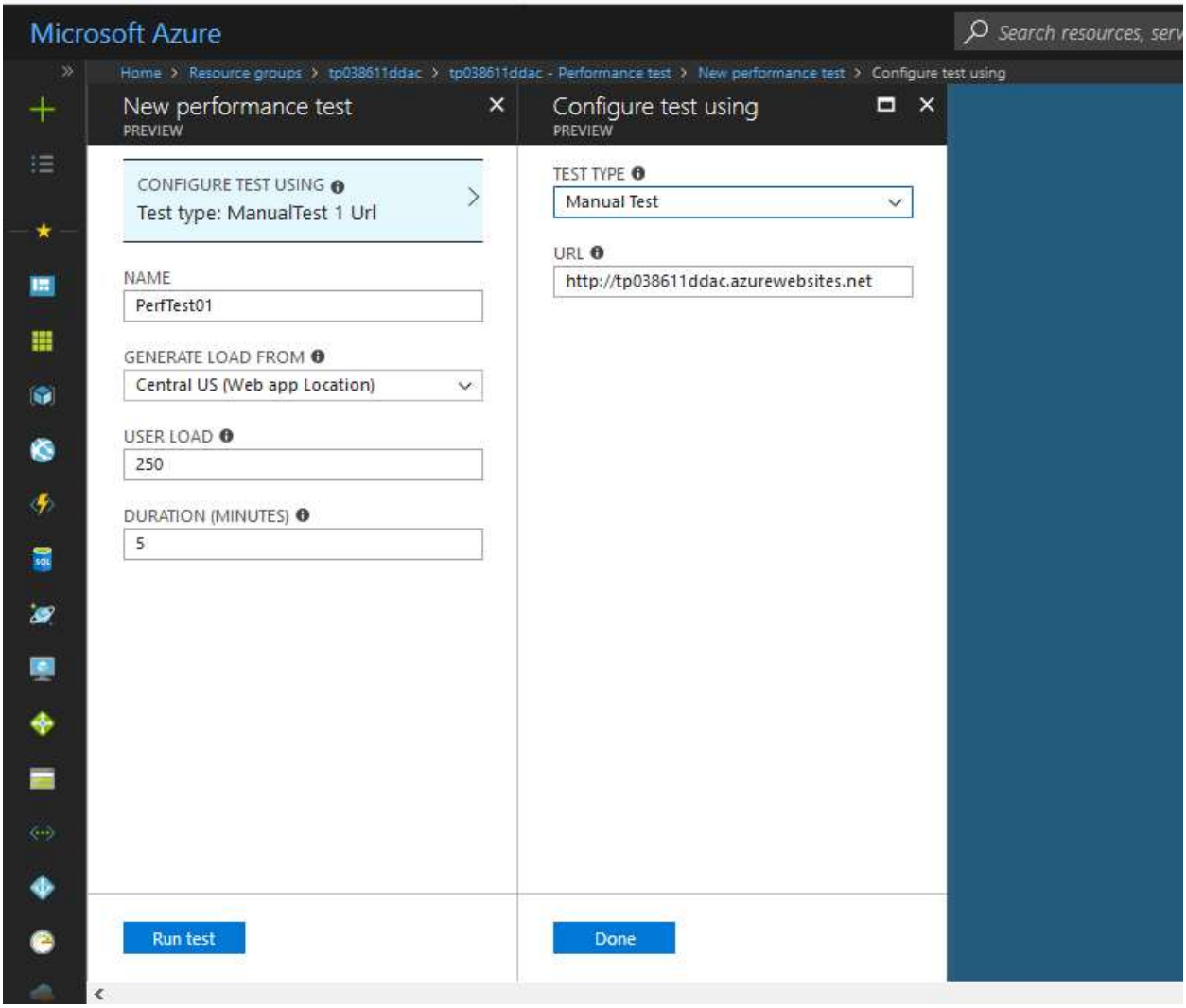
		3. List of schedules will be displayed in a table in the screen			
TAD6	Create Schedule	1. Login as an admin 2. Click on schedule at the navigation bar 3. Click on create button 4. Enter the details necessary 5. Click on the submit button 6. Navigate back to the schedule page 7. Schedule will be added into the table	Schedule is added to the database	Schedule is added to the database	Pass
AGENT					
TAG1	View Customer	1. Login as an agent 2. Click on customer at the navigation bar 3. List of customers will be displayed in a table in the screen	The table of customer is displayed	The table of customer is displayed	Pass
TAG2	Create Customer	1. Login as an agent 2. Click on customer at the navigation bar 3. Click on create button 4. Enter the details necessary 5. Click on the submit button	customer is added to the database	customer is added to the database	Pass

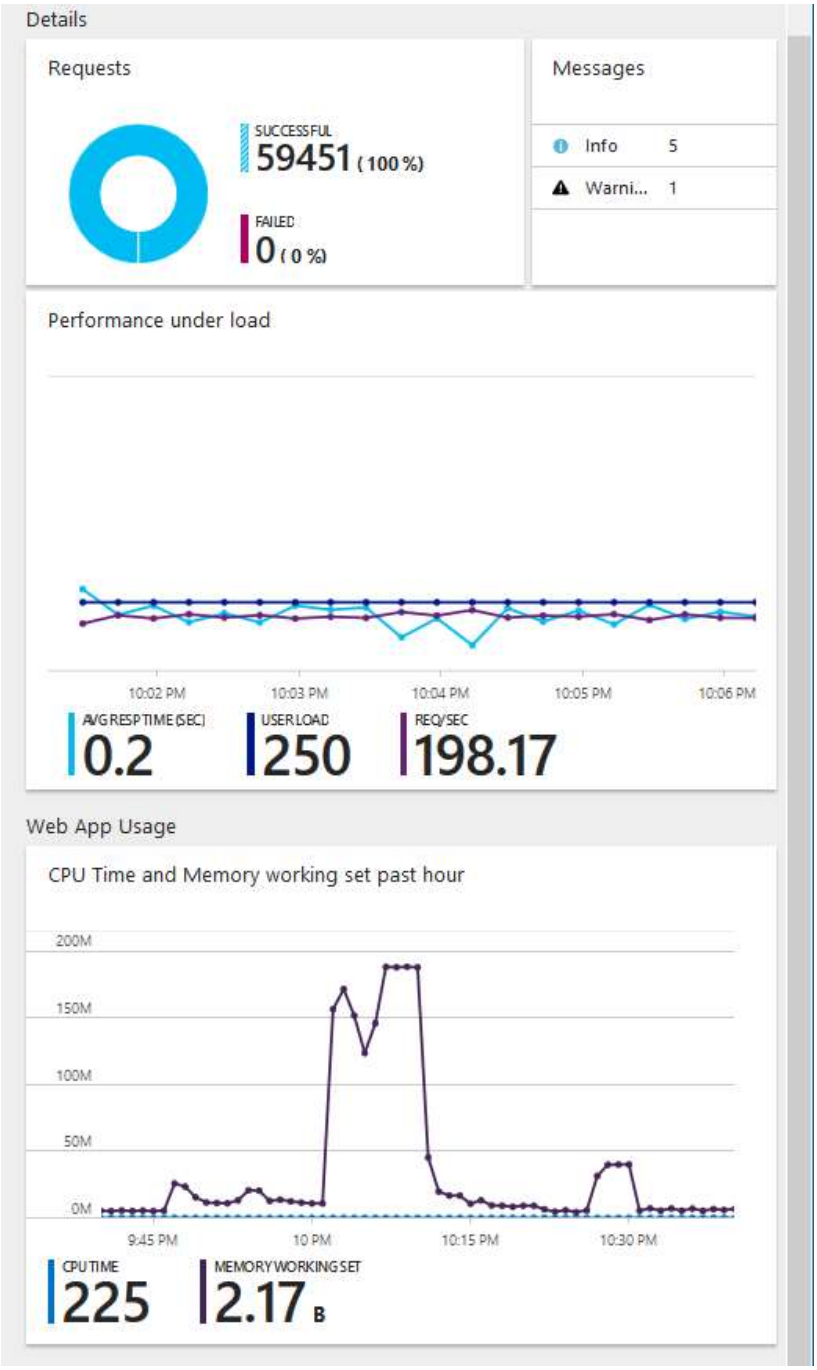
		6. Navigate back to the customer page 7. Customer will be added into the table			
TAG3	View Cargo	1. Login as an agent 2. Click on cargo at the navigation bar 3. List of cargo will be displayed in a table in the screen	The table of cargo is displayed	The table of cargo is displayed	Pass
TAG4	Create Cargo	1. Login as an agent 2. Click on cargo at the navigation bar 3. Click on create button 4. Enter the details necessary 5. Click on the submit button 6. Navigate back to the cargo page 7. Cargo will be added into the table	New cargo is added to the database	New cargo is added to the database	Pass
TAG5	View Warehouse	1. Login as an agent 2. Click on warehouse at the navigation bar 3. List of warehouses will be displayed in a table in the screen	The table of warehouse is displayed	The table of warehouse is displayed	Pass
TAG6	Create Warehouse	1. Login as an agent 2. Click on warehouse at the navigation bar	warehouse is added to the database	warehouse is added to the database	Pass

		3. Click on create button 4. Enter the details necessary 5. Click on the submit button 6. Navigate back to the warehouse page 7. Warehouse will be added into the table			
TAG7	View Schedule	1. Login as an agent 2. Click on schedule at the navigation bar 3. List of schedules will be displayed in a table in the screen	The table of schedule is displayed	The table of schedule is displayed	Pass

5.2 Performance

The performance analysis testing for the CMS can be achieve through conducting the performance test on server with simulation of user visiting the server. This is a feature of Azure Portal and the testing here will be conducted based on three different standard plans and the number of user load within the given period.





From the report, it shows that the system will be stable, and performance will still be able to maintain even when 250 users access the system concurrently over the course of 5 minutes. It is acceptable, and the business can continue to run smoothly without to worry about downtime or the possible of system crash due to overloading of user acting upon the system concurrently.

6.0 Conclusion

In conclusion, the development of a container management system that is cloud based had been an educative and changeling journey. Developing the web application and then to deploy it onto Microsoft Azure required careful calculations and planning as it provides many different options to customize the cloud system as per required. Microsoft Azure is a fundamental and needful knowledge when it comes to cloud computing. This assignment had given me the opportunity to practice new and relevant skill that will be valuable in the cloud technology age where everything is to be done on cloud. Cloud computing is a great euphemism for centralization of computer services under one server.

References

1) Apprenda. (2018). *IaaS, PaaS, SaaS (Explained and Compared)*. [online] Available at: https://apprenda.com/library/paas/iaas-paas-saas-explained-compared/ [Accessed 6 Apr. 2018].

2) Azure.microsoft.com. (2018). *What is PaaS? Platform as a Service | Microsoft Azure*. [online] Available at: https://azure.microsoft.com/en-us/overview/what-is-paas/ [Accessed 5 Apr. 2018].

3) SearchCloudComputing. (2018). *What is Platform as a Service (PaaS)? - Definition from WhatIs.com*. [online] Available at: https://searchcloudcomputing.techtarget.com/definition/Platform-as-a-Service-PaaS [Accessed 4 Apr. 2018].

4) Violino, B. (2018). *What is PaaS? Software development in the cloud explained*. [online] InfoWorld. Available at: https://www.infoworld.com/article/3223434/paas/what-is-paas-software-development-in-the-cloud.html [Accessed 13 Apr. 2018].