

**INDIVIDUAL ASSIGNMENT**

**CT07-3-3-DDAC**

**DESIGNING AND DEVELOPING APPLICATION ON THE CLOUD**

|  |  |  |
| --- | --- | --- |
| **Name** | **:** | Hong Mei Shean |
| **Intake Code** | **:** | UC3F1706SE |
| **Student ID** | **:** | TP038907 |
| **Lecturer Name** | **:** | Dr. Kalai Anand A/L Ratnam |
| **Hand out Date** | **:** | 13th April 2018 |

# Acknowledgment

Firstly, I would like to thank my lecturer for this module, Dr, Kalai for teaching me this module knowledge and skills and provide students all the useful necessary information which can assist me to accomplish of development a web application on Azure. Furthermore, I would like to thank for my course mate who have supported me in developing the web application on Azure, provided guidance for me and discussed the issues that faced in development.

Contents

[Acknowledgment 2](#_Toc511401849)

[1.0 Introduction 3](#_Toc511401850)

[2.0 Project Plan 4](#_Toc511401851)

[3.0 Design 5](#_Toc511401852)

[3.1 Design Considerations 5](#_Toc511401853)

[3.2 Modelling 5](#_Toc511401854)

[3.2.1 Use Case Diagram 5](#_Toc511401855)

[3.2.2 Use Case Specification 6](#_Toc511401856)

[3.2.3 Sequence Diagram 12](#_Toc511401857)

[3.2.4 Class Diagram 19](#_Toc511401858)

[3.3 Cloud Architectural Diagram 20](#_Toc511401859)

[4.0 Implementation 21](#_Toc511401861)

[4.1 Application Development 21](#_Toc511401862)

[4.2 Set Up Azure serice 21](#_Toc511401863)

[4.3 Azure publishing 30](#_Toc511401866)

[5.0 Testing 33](#_Toc511401871)

[5.1 Unit Testing 33](#_Toc511401872)

[5.2 Performance Testing 41](#_Toc511401873)

[6.0 Conclusion 44](#_Toc511401874)

[7.0 References 45](#_Toc511401875)

# Introduction

Maersk Line was found in year 1928 and now is the global container division and the world’s largest container company for provided shipping to customers via 374 offices in 116 countries. Besides, the company has operated over 600 vessels and 2.6 million TEC capacity. However, due to the businesses grown and improve the organizational flexibility, Maersk has decided to use cloud powered solutions that can solve the issues that faced by company.

Maersk company is looking for creating a Container Management System (CMS) to manage and operate all the containers which can reduces the supply chain operation cost and can manage flexibility and efficiency. After developing the CMS web application, deploying the web application development to the Microsoft Azure platform. The users that involved used in CMS application are Admin and Agent of the company. The Admin can use the CMS application to create schedule for each vessel, register new agent, view the bookings that have made by the agents and so on. For the ability of the agent in the CMS application are make vessel booking and register new customers.

# 2.0 Project Plan

|  |  |  |
| --- | --- | --- |
| WBS | Task Name | Duration |
| **1** | **Maersk Line CMS** | **26 days** |
| **1.1** | **Introduction** | **2 days** |
| 1.1.1 | Background | 1 day |
| 1.1.2 | Objective | 1 day |
| 1.1.3 | Scope | 1 day |
| 1.1.4 | Non-functional Requirement Specification | 1 day |
| 1.1.5 | Functional Requirement Specification | 1 day |
| **1.2** | **Project Plan** | **1 day** |
| 1.2.1 | Work Breakdown Structure | 1 day |
| 1.2.2 | Gantt Chart | 1 day |
| **1.3** | **Design** | **7 days** |
| 1.3.1 | Cloud Architecture design | 1 day |
| 1.3.2 | Design Consideration | 1 day |
| **1.3.3** | **Modelling** | **6 days** |
| 1.3.3.1 | Use Case Diagram | 1 day |
| 1.3.3.2 | Use Case Description | 2 days |
| 1.3.3.3 | Sequence Diagram | 3 days |
| 1.3.3.3 | ………Class Diagram | 0.5h |
| **1.4** | **Implementation** | **13 days** |
| 1.4.1 | ASP.NET Web Application | 10 days |
| 1.4.2 | Azure Resource Group | 1 day |
| 1.4.3 | Azure SQL Server | 1 day |
| 1.4.4 | Azure SQL Database | 1 day |
| 1.4.5 | Azure Web Application Service | 1 day |
| 1.4.6 | Azure Traffic Manager | 1 day |
| 1.4.7 | Azure Web Application Autoscaling | 1 day |
| 1.4.8 | System Interface | 2 days |
| **1.5** | **Testing** | **2 days** |
| 1.5.1 | Performance Testing | 1 day |
| 1.5.2 | Unit Testing | 1 day |
| 1.6 | Conclusion | 1 day |

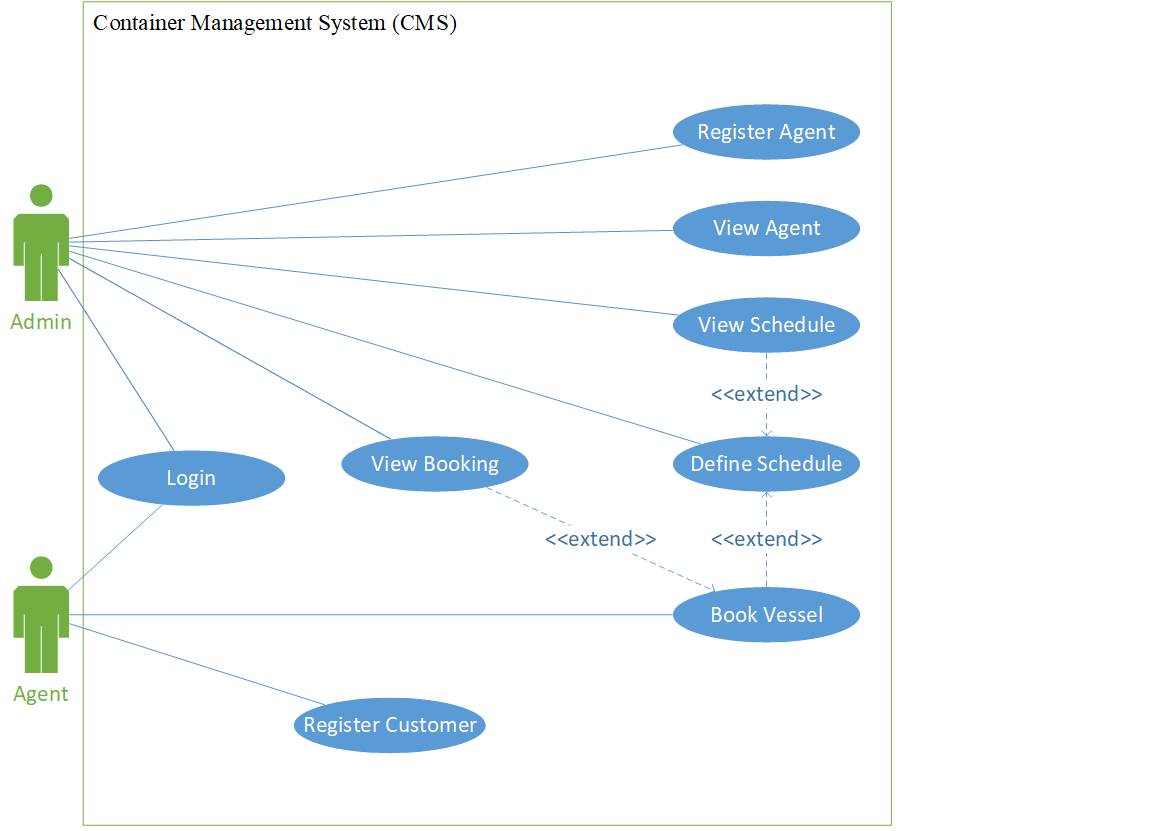
# 3.0 Design

## 3.1 Design Considerations

Before entering the designing stage, there are few assumptions need to be mentioned that the main intention of the Maersk Line company is looking for a Container Management System(CMS) to minimize the cost of overall supply chain and can manage and control their logistics in a efficiency way by creating a cloud-based web application which is deployed in Microsoft Azure to improve the performance. According to the limiting budget, the region that will be cover is only two areas which are Southeast Asia and West Europe region to perform better in supply chain operation to minimize the operation cost.

## 3.2 Modelling

### 3.2.1 Use Case Diagram



### 3.2.2 Use Case Specification

**Admin and Agent**

Login

|  |  |
| --- | --- |
| Use case | Login |
| Summary | User need to key-in the email and password to login to the application. |
| Dependency | None |
| Actor | Users |
| Pre-condition | Users had already an account in the application. |
| Description of main sequence | 1. Application display the login activity page. 2. User key-in email and password. 3. User click on the login button. 4. Application validate the email and password. 5. Application display main activity page for user. |
| Description of alternative sequence | 5(a). if the application displays a message “the email is incorrect”   or “the password is incorrect”, user need to reenter the email   or password to proceed login activity.  5(b). if the application displays a message “the email field cannot   be empty” or “the password cannot be empty”, user need to   full-in all the required field and click on the login button.  5(b). if the application displays a message “please fill-in the  blank” , user need to full-in all the blank field and click on the login button. |
| Post-condition | User can start to chat with other user and start other activity in the application. |

**Admin**

Register Agent

|  |  |
| --- | --- |
| Use case | Register Agent |
| Summary | Admin register new agent into the application. |
| Dependency | None |
| Actor | Admin |
| Pre-condition | Admin need to log-in into application first. |
| Description of main sequence | 1. Admin log-in into application. 2. Admin click on the Register Agent icon. 3. Application display Register Agent page 4. Admin enter new agent details. 5. Admin click on save button 6. Application display “Create successful” message. |
| Description of alternative sequence | 6(a). if the application displays a message “please fill-in the blank” admin need to fill-in all the blanks required.  6(b). if the application displays a message “the password need at  least 6 characters” or “the password required at least one non word or digit character.”, admin need to re-enter the required filed. |
| Post-condition | Admin can use others features in the application. |

Define Schedule

|  |  |
| --- | --- |
| Use case | Define Schedule |
| Summary | Admin create schedule for allow agent to book a schedule. |
| Dependency | <<entend>>View Schedule  <<extend>>Book Vessel |
| Actor | Admin |
| Pre-condition | Admin had an account in the application. |
| Description of main sequence | 1. Admin had log-in into application. 2. Application display home page. 3. Admin click on Create Schedule button. 4. Application display Create Schedule page. 5. Admin enter schedule details. 6. Admin click on the Save button. 7. Application display successful message. |
| Description of alternative sequence | 7(a). if the application displays a message “please fill-in the blank” admin need to fill-in all the blanks required.  7(b). if the application displays a message “the Arrived date must  grater than the Departure date, admin need to choose another date for the arrived date. |
| Post-condition | Admin can continue use others features of the application. |

View Agent

|  |  |
| --- | --- |
| Use case | View Agent |
| Summary | Admin can view agent details such as name and email. |
| Dependency | None |
| Actor | Admin |
| Pre-condition | Admin log-in successful into the application. |
| Description of main sequence | 1. Admin had log-in into the application. 2. Application display home page 3. Admin click on the View Agent icon 4. Application display View Agent page and agents details. |
| Description of alternative sequence | None |
| Post-condition | Admin can continue use others features that provided in the application such as create schedule. |

View Booking

|  |  |
| --- | --- |
| Use case | View Booking |
| Summary | Admin can view what booking had made by agents. |
| Dependency | None |
| Actor | Admin |
| Pre-condition | Admin had log-in into the application |
| Description of main sequence | 1. Application display admin home page. 2. Admin click on the View Booking icon. 3. Application display View Booking page and booking details. |
| Description of alternative sequence | None |
| Post-condition | Admin can continue use others features that provided in the application such as create schedule. |

View Schedule

|  |  |
| --- | --- |
| Use case | View Schedule |
| Summary | Admin can view the schedule. |
| Dependency | None |
| Actor | Admin |
| Pre-condition | Admin had log-in into application. |
| Description of main sequence | 1. Application display the admin home page. 2. Admin click on the navigation bar. 3. Admin choose the View Schedule icon. 4. Application display View Schedule page and schedule details. |
| Description of alternative sequence | None. |
| Post-condition | Admin can view agent or create schedule. |

**Agent**

Register Customer

|  |  |
| --- | --- |
| Use case | Register Customer |
| Summary | Agent can register the new customers. |
| Dependency | None |
| Actor | Agent |
| Pre-condition | Agent had already an account into the application. |
| Description of main sequence | 1. Agent log-in into application. 2. Application display agent home page. 3. Agent click on the Customer icon on the navigation bar. 4. Application display Customer page. 5. Agent enter the customer details. 6. Agent click on the Save button. 7. Application display successful message. |
| Description of alternative sequence | 7(a). if the application displays a message “please fill-in the blank” admin need to fill-in all the blanks required. |
| Post-condition | Agent can continue use the application for others action such as book vessel. |

Book Vessel

|  |  |
| --- | --- |
| Use case | Book Vessel |
| Summary | Agent book the vessel for the customer item. |
| Dependency | <<extend>>View booking |
| Actor | Agent |
| Pre-condition | Agent had successful log-in into application. |
| Description of main sequence | 1. Application display agent home page. 2. Agent click on the book vessel icon. 3. Application display Search page for agent enter the place for shipment. 4. Agent enter the location for the shipment load and discharge. 5. Agent click on Search button. 6. Application display the available vessels on the Result page. 7. Agent click on the book action link icon. 8. Application display the Booking Form page. 9. Agent enter the booking details. 10. Agent click on the Save button. 11. Application display successful message. |
| Description of alternative sequence | 6(a).if the application displays a message “please fill-in the blank” agent need to fill-in all the blanks required.  6(b). if the application displays a message “the search data does not  match with the database”, agent need to re-enter available search data.  11(a). if the application displays a message “please fill-in the blank” agent need to fill-in all the blanks required.  11(b). if the application displays a message “please enter the space value between 1 to 40”, agent need to re-enter the space value. |
| Post-condition | Agent can continue use others features in the application. |

### 3.2.3 Sequence Diagram

Login

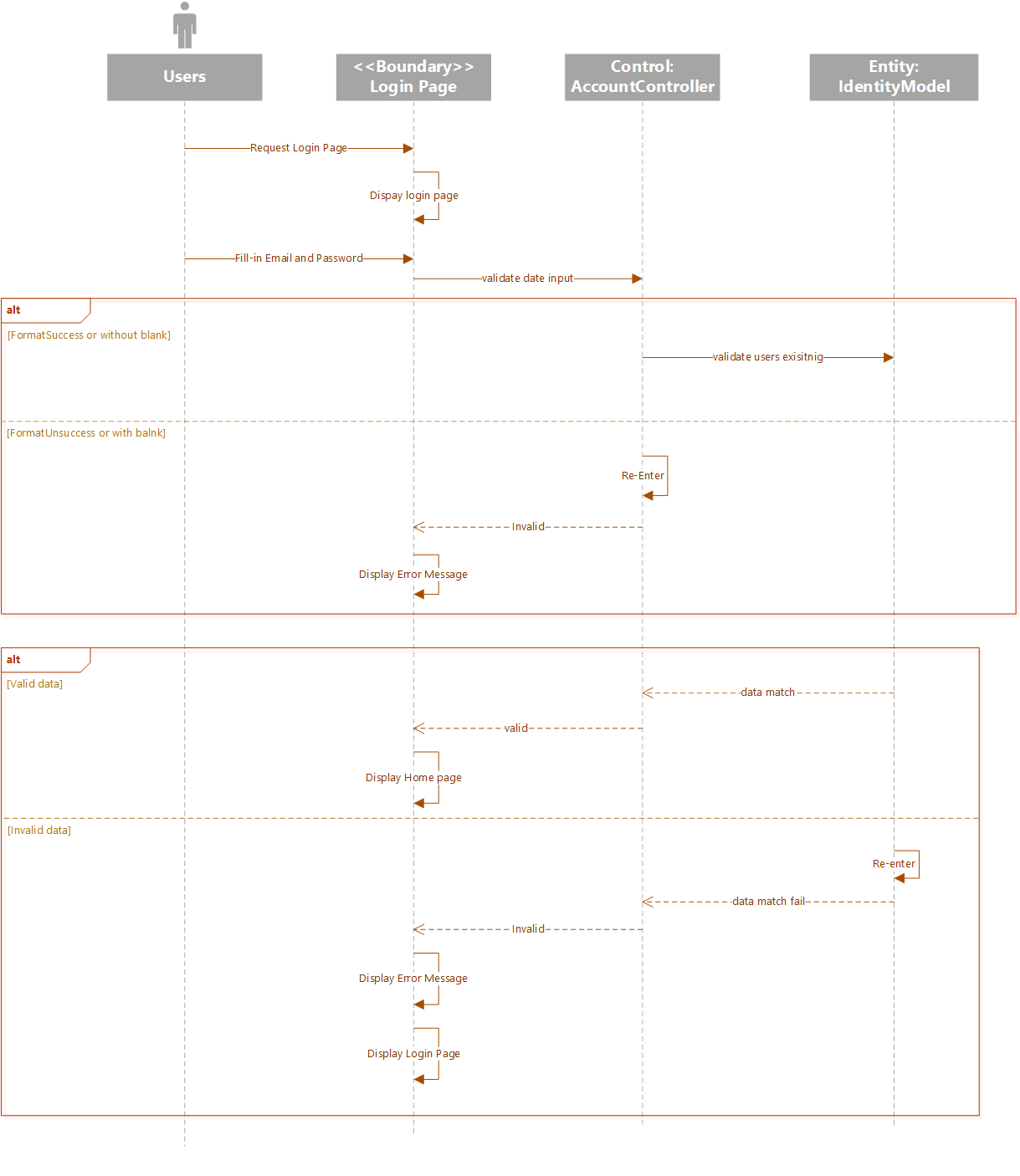
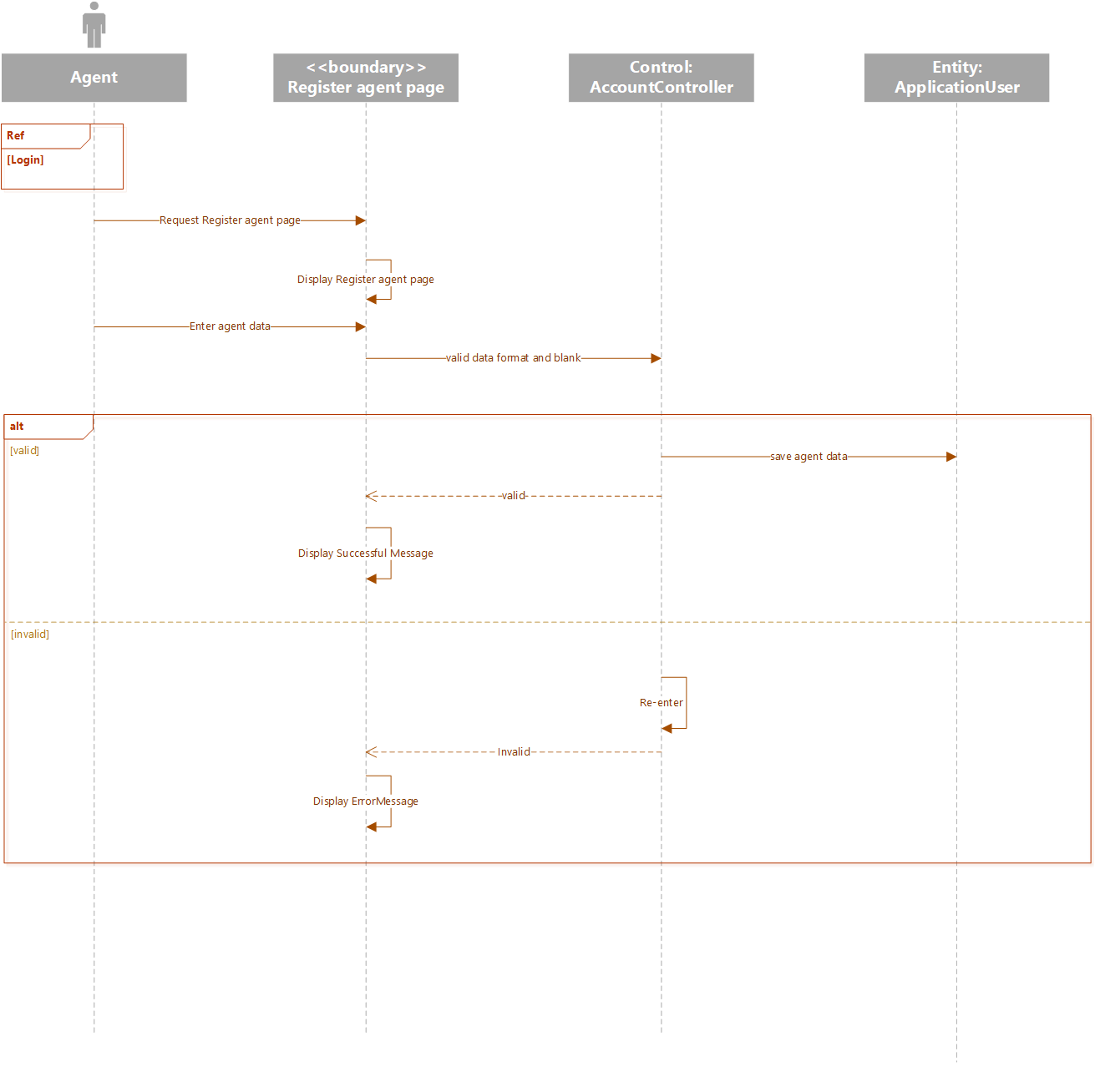


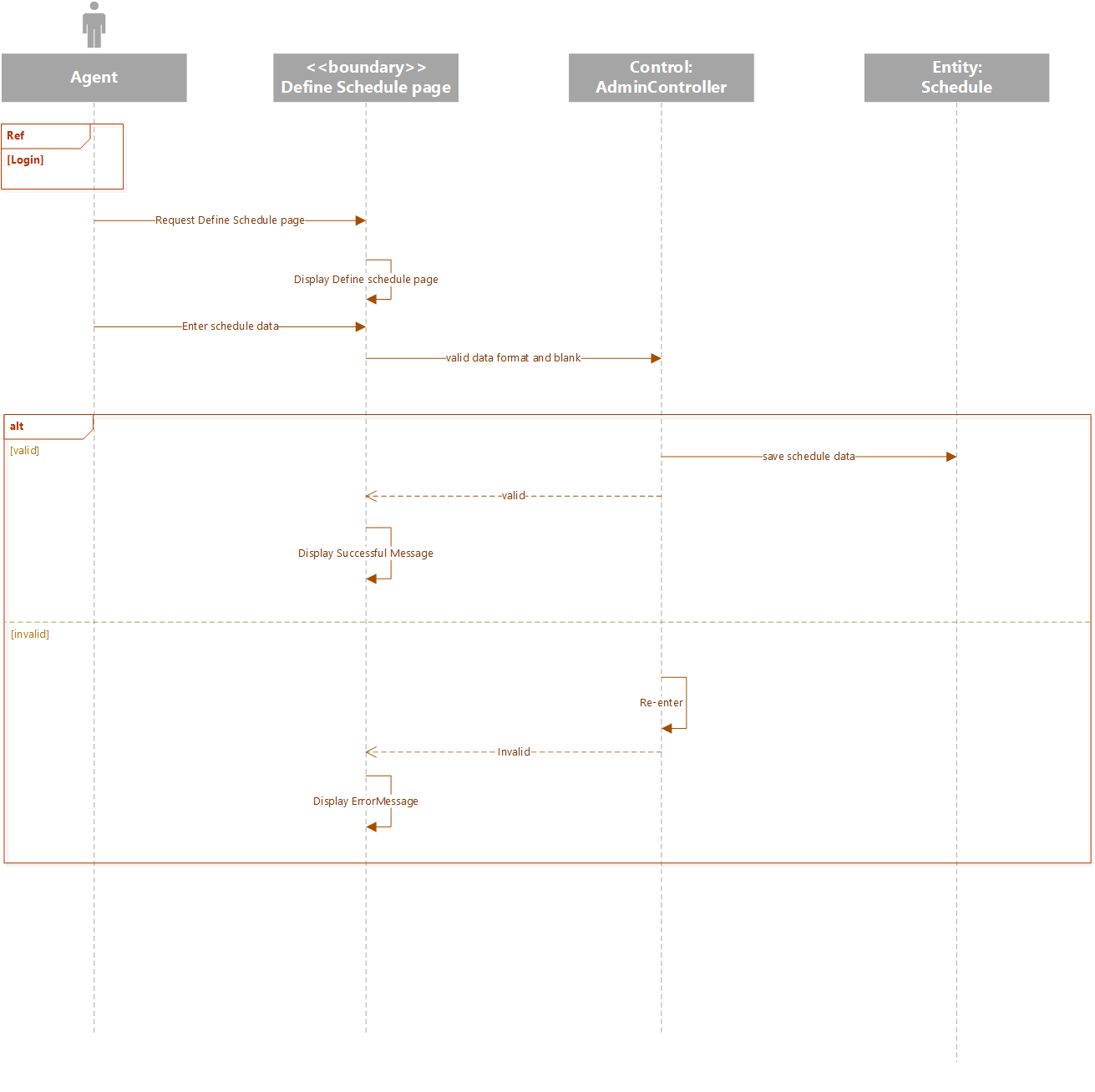
Figure 2: Login Sequence diagram

**Admin**

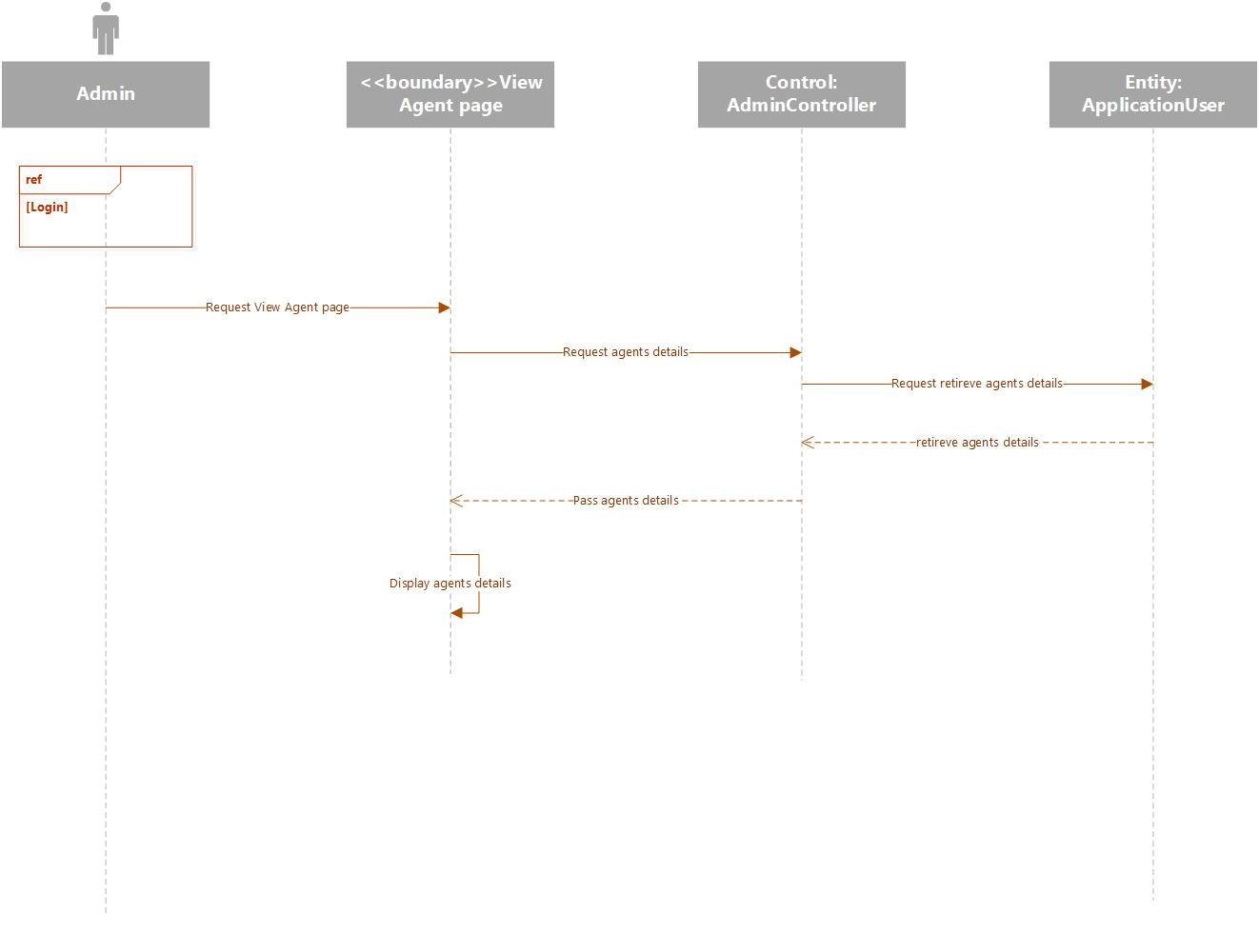
Register Agent



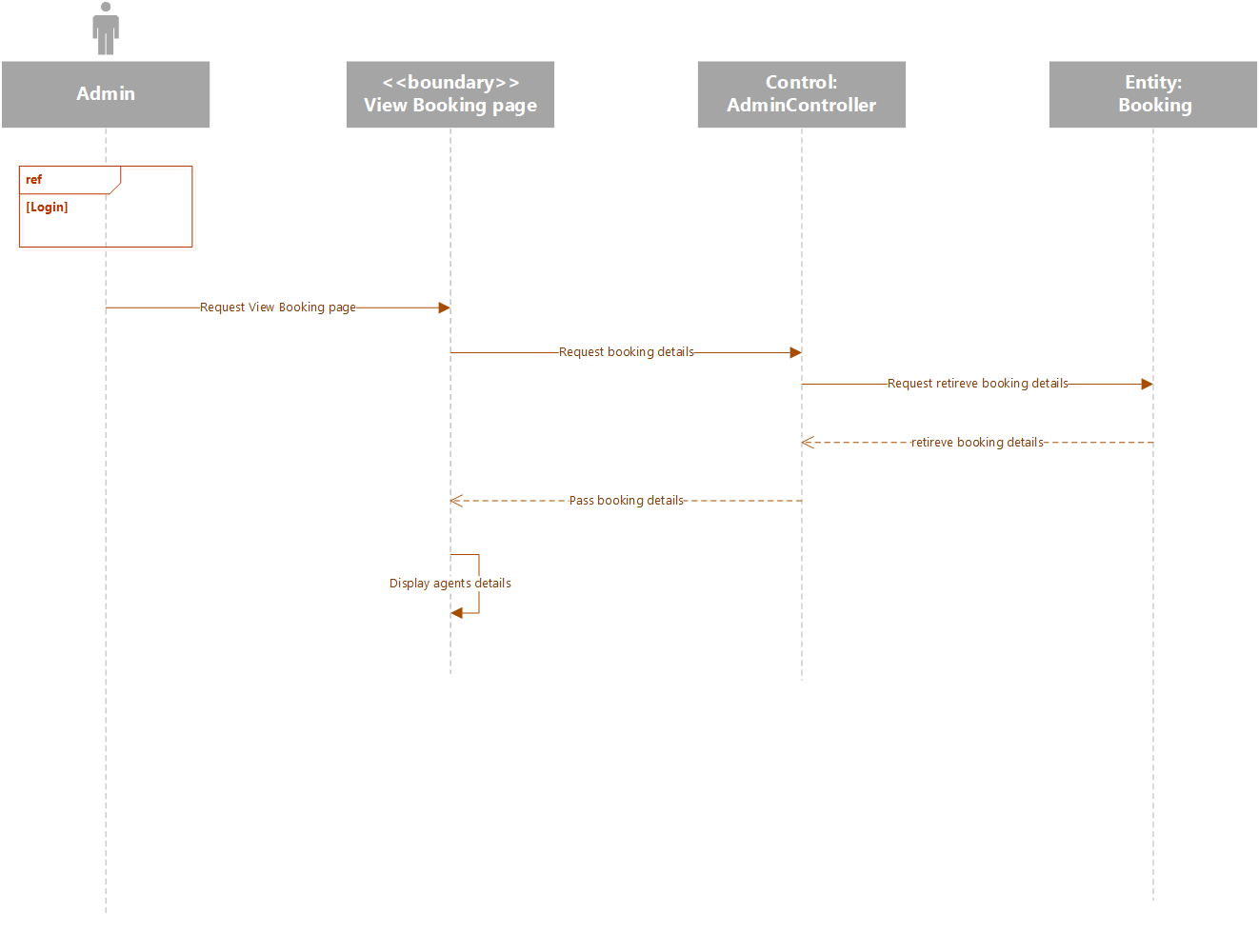
Define Schedule



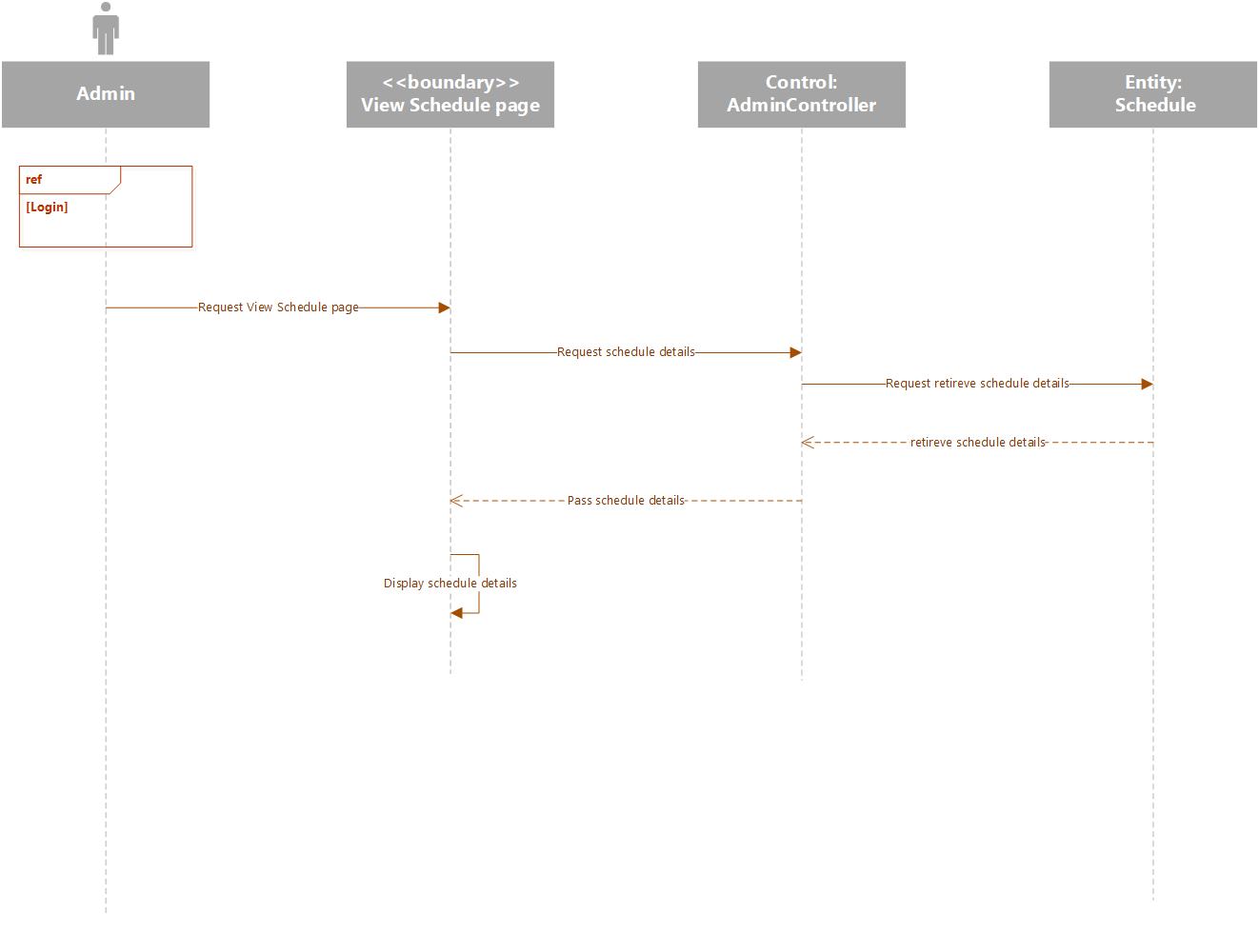
View Agent



View Booking

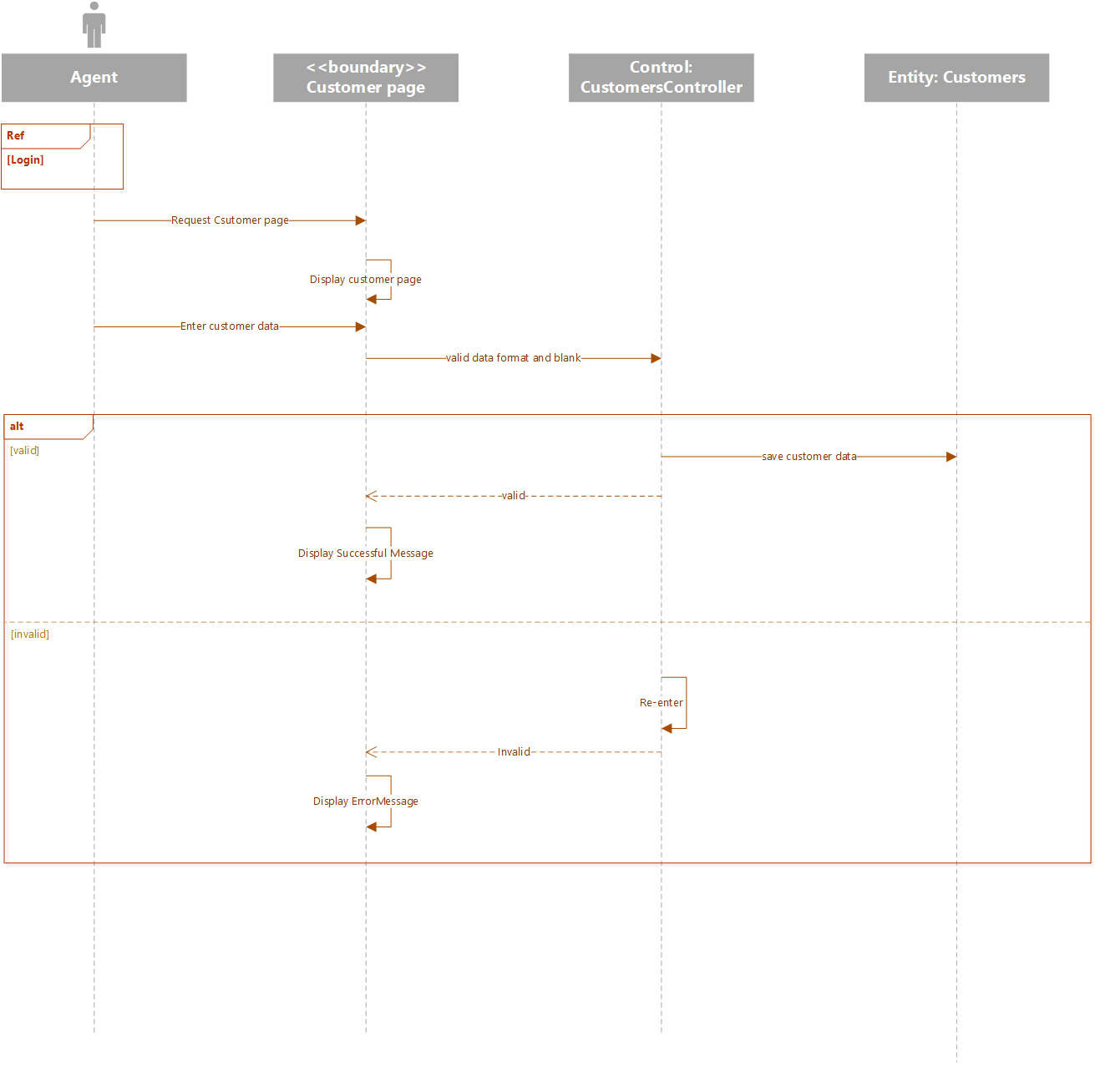


View Schedule

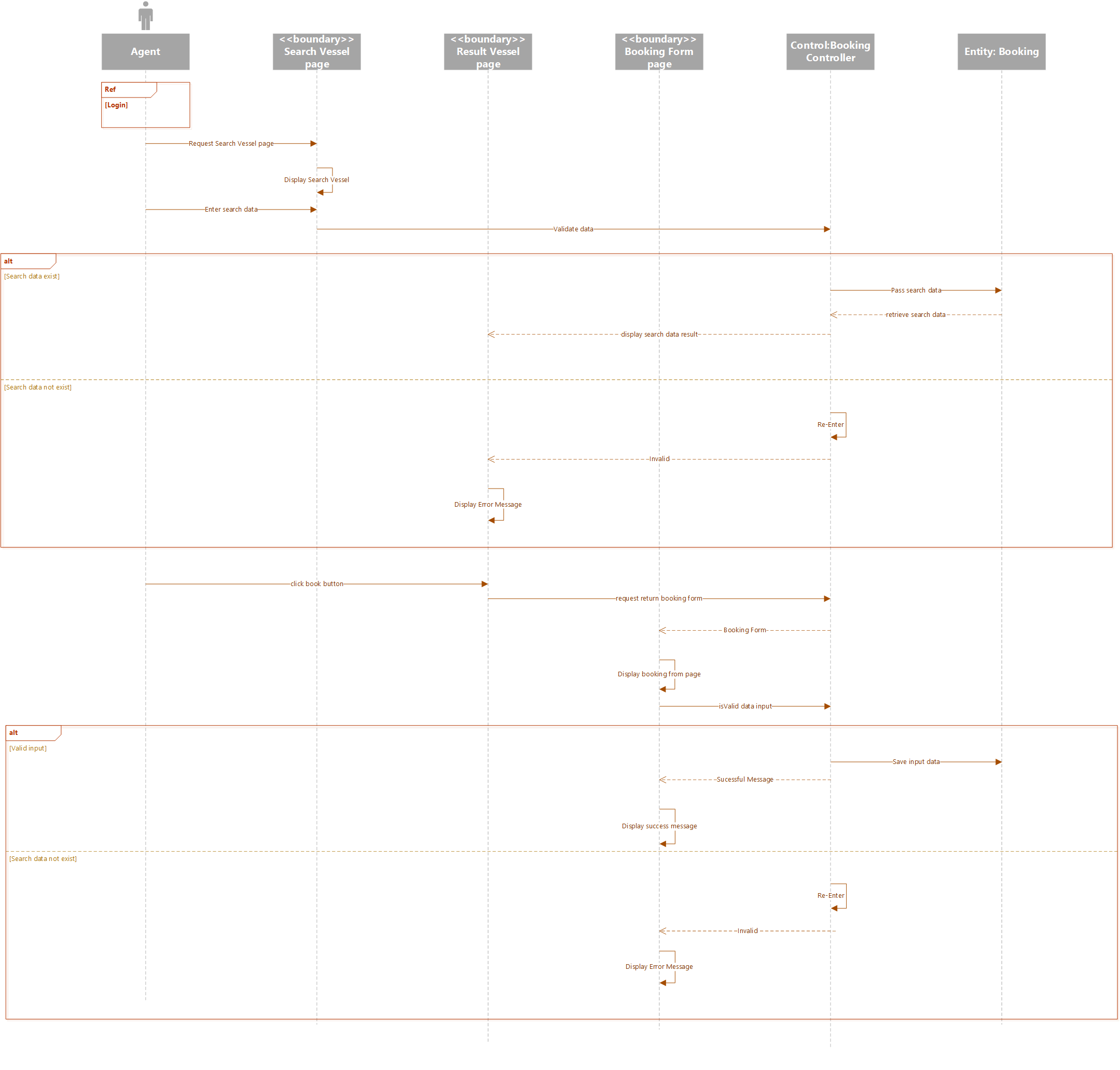


**Agent**

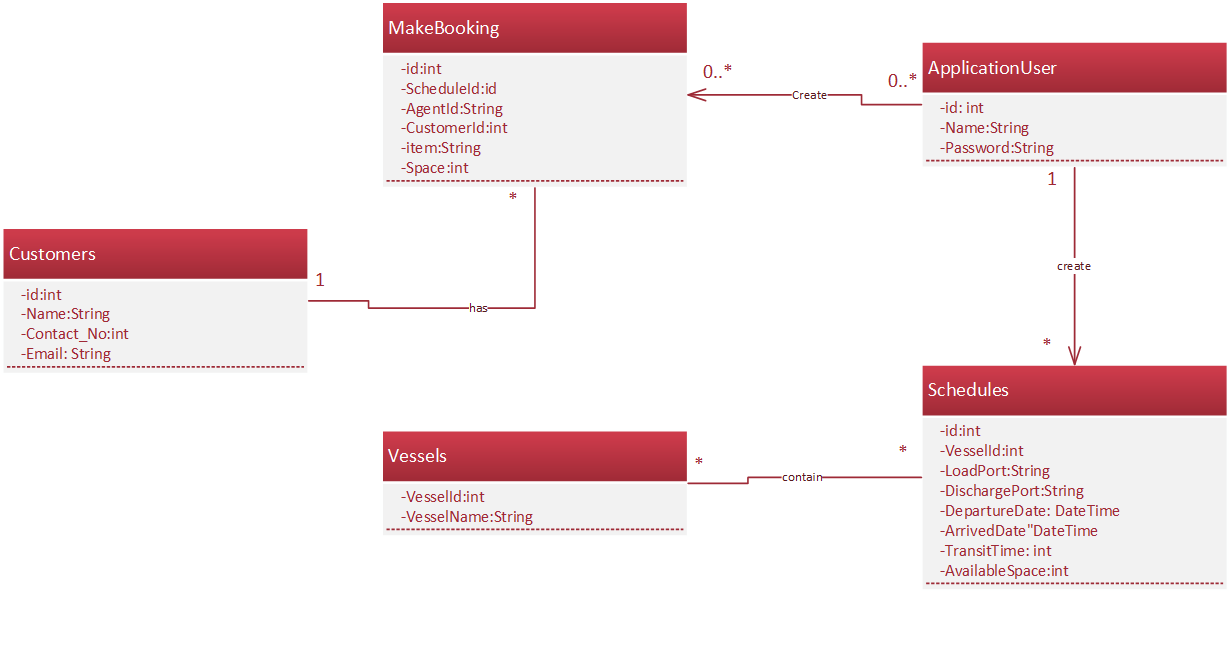
Register Customer



Book Vessel



### 3.2.4 Class Diagram



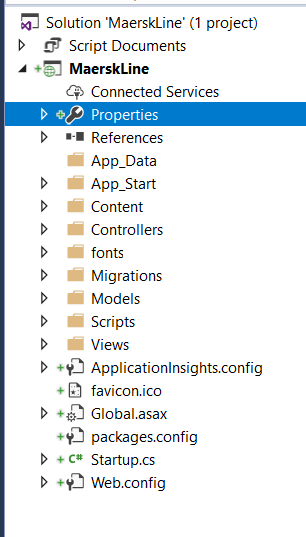
## 3.3 Cloud Architectural Diagram

# 

The cloud design architecture above is cloud platform design for two regions which are for the Southeast Asia and West Europe. The container management system (CMS) after complete developing, it will deploy to the cloud platform. There are set two database which are primary database and secondary database. The primary database is applied in the Southeast Asia and the Secondary database is deploying in West Europe area. Web application is more focusing in the Southeast Asia, therefore, the main web will be located in to Southeast Asia and using the geo-replication to manage and handle the backup server in West Europe. The performance for each region database can be managed by the traffic manager which is ensure the performance for both database is performance well.

# 4.0 Implementation

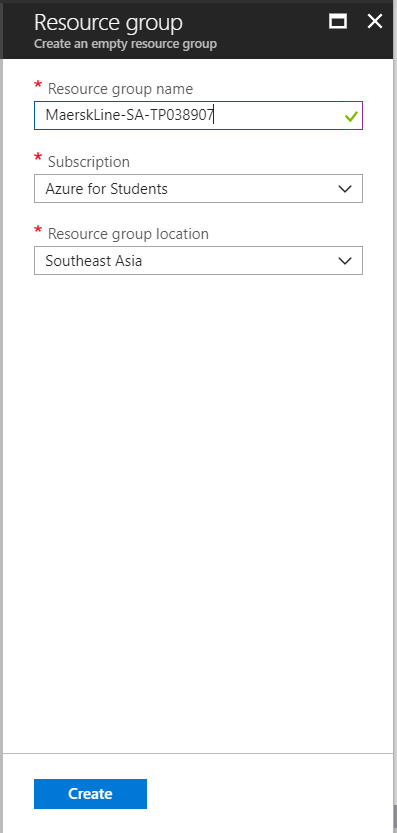
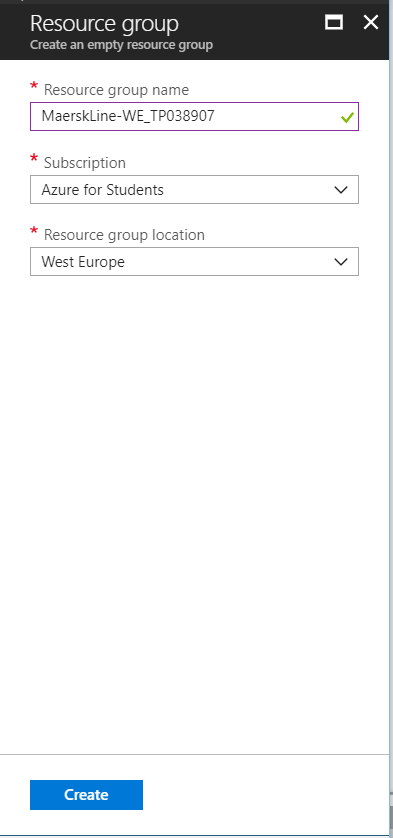
## 4.1 Application Development



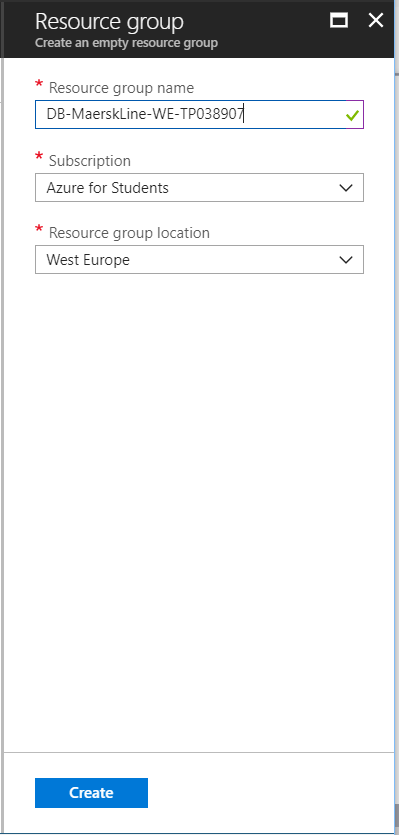
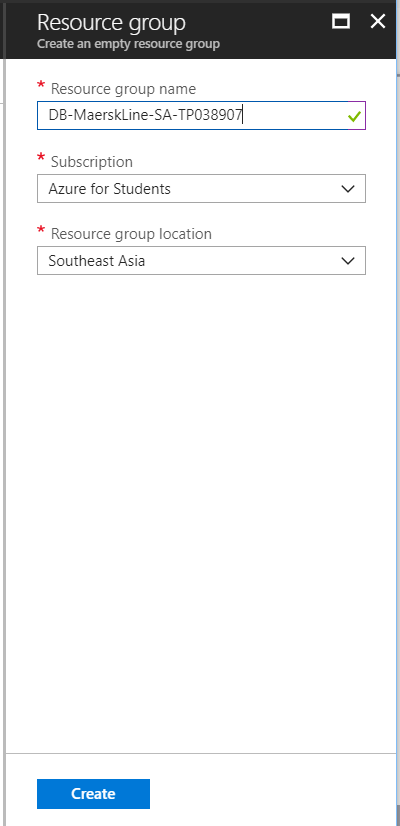
The Maersk Line Web application is developing using Visual Studio 2017 which is using ASP.Net MVC5. ASP.NET MVC is a web application that has implemented the model-view-controller (MVC). It can allow a developer to develop a website in a short time. There are three components that developer is using to create the web application which are the model, controller and the view. The model is act like a class which all the getter and setter for the attributes are put inside the model. For the View is used to display View page to the user. For the controller, it is used to perform something or action to the view. Besides, if using ASP.NET MVC, it has a template already, developer no need to worry about the design, there are provide bootstrap and jQuery. Therefore, developer no need to import bootstrap for design layout.

## 4.2 Set Up Azure serice

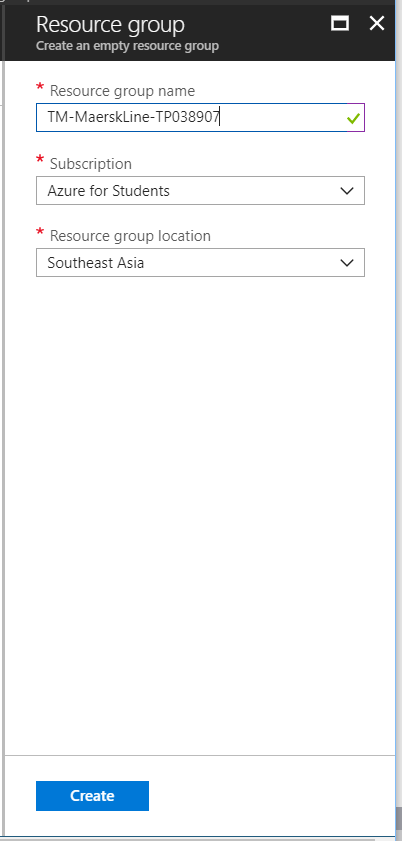
After the developer has done the web application development and ensure all the features of the web application is function well. The developer can start to publish the Web application publish to Microsoft Azure. The first step for the developer is to create Resource Group.

The 

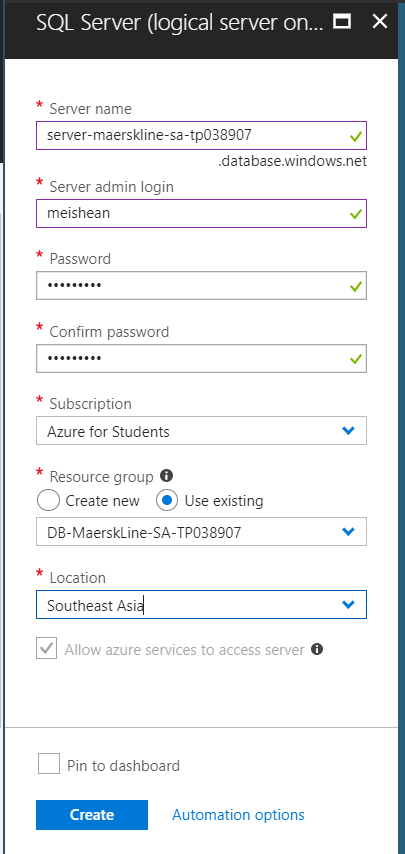
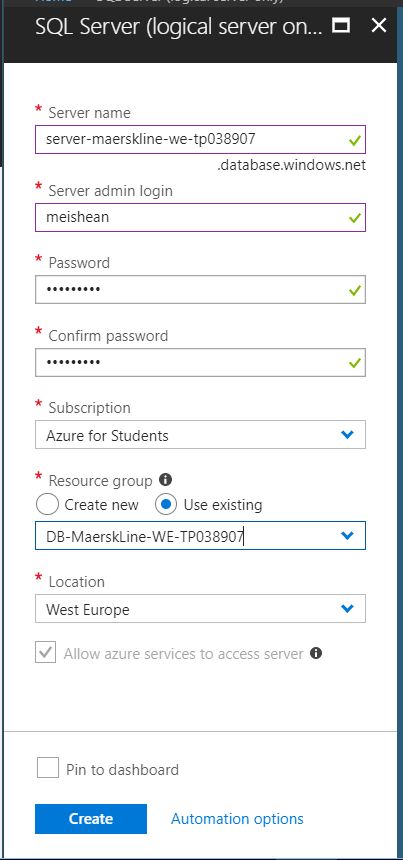
The developer created two resource group for the Southeast Asia region and West Europe region which divided them as primary web app and secondary web app. The name for each web app are differenced which can understand which the primary web app and secondary web app. Are the primary web app which is mean will be more focus on that region users.



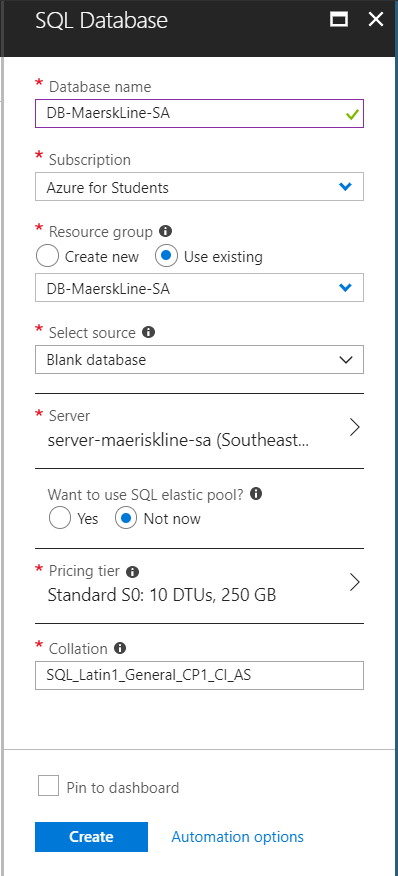
The second action is to create sql database for resource group. The name for given them are difference which can understand which database for which region which the developer does not mix up. The DB word is easy for the developer to understand which the database in the resource group are, since the developer will create more in resource group.



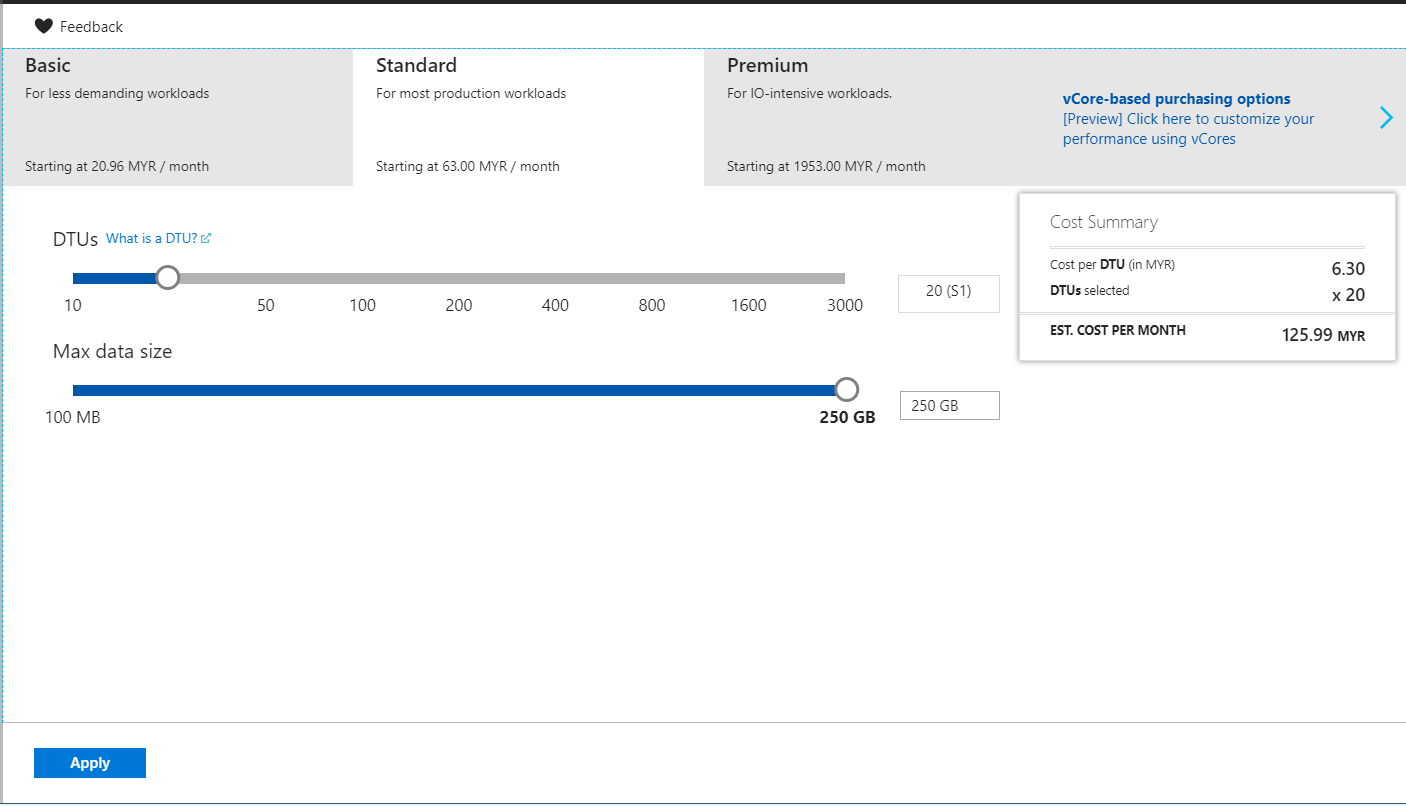
After created the resource group for database, the developer continues create resource group for the traffic manager which can handle and control the traffic endpoint distribution for the users. The It support many services such as cloud service, Web Apps and so on to control the endpoint distributor (Docs.microsoft.com, 2018).



The next action is to create sever for each region. Both server name are using we and se to differential them which se is represent Southeast Asia and we is represent West Europe. The server must be create before the SQL database created. The resource group resources is chosen based on the region and the SQL server that had been created before. The location must be choose correctly according the region that chosen for the server in resource group before.



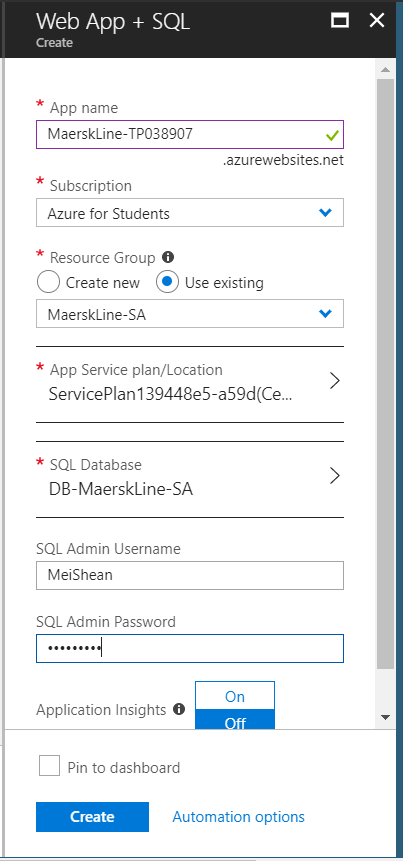
The image above is create Sql database which is primary databaase.The database name as DB-MaerskLine-SA. And select the exiting server. The Pricing tier that had chosen for this database is 20GB which cost 125.99 malaysia ringgit per month. The package is shown in below.



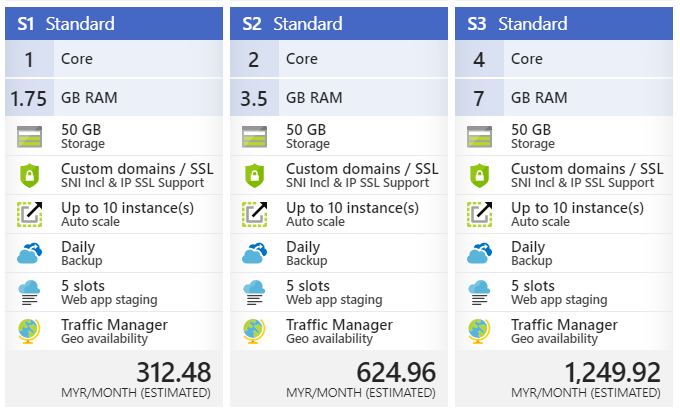
## 

## 

The above two images are showing the secondary database related information. The secondary database is located in West Europe. Using the Geo replication is created copy of database for different region, normally the secondary database is applied for the primary database. When want to use the Geo replication service for secondary database, it had found in SQL database service. The map is used to display the region desired for the secondary database.



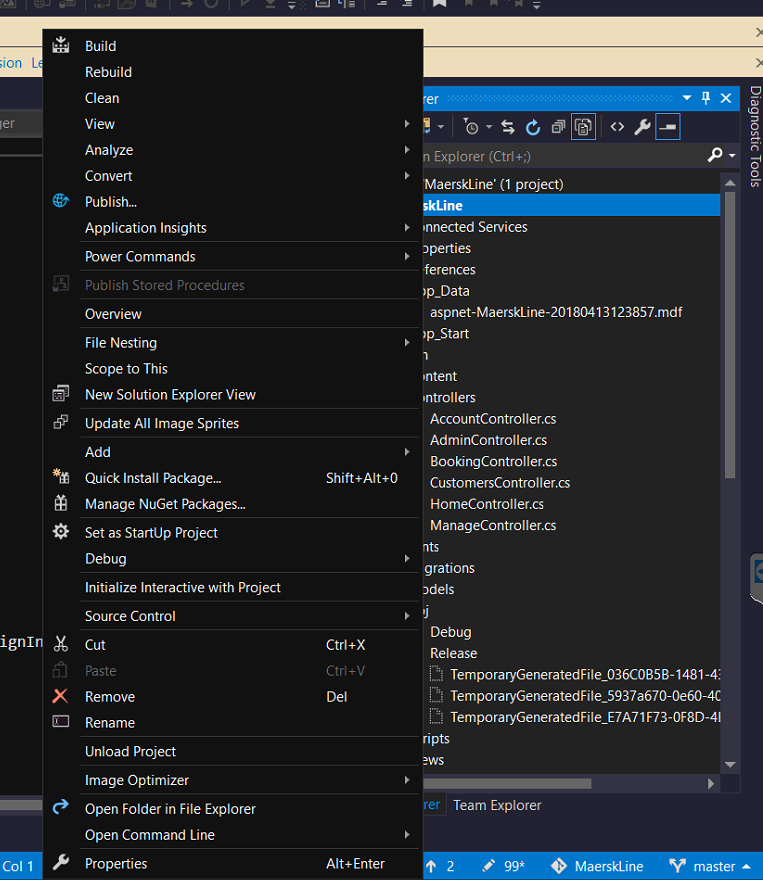
This is application service, the developer is applied the Web app + sql service for creating the web application. The name for the link will be MaerskLine-TP038907



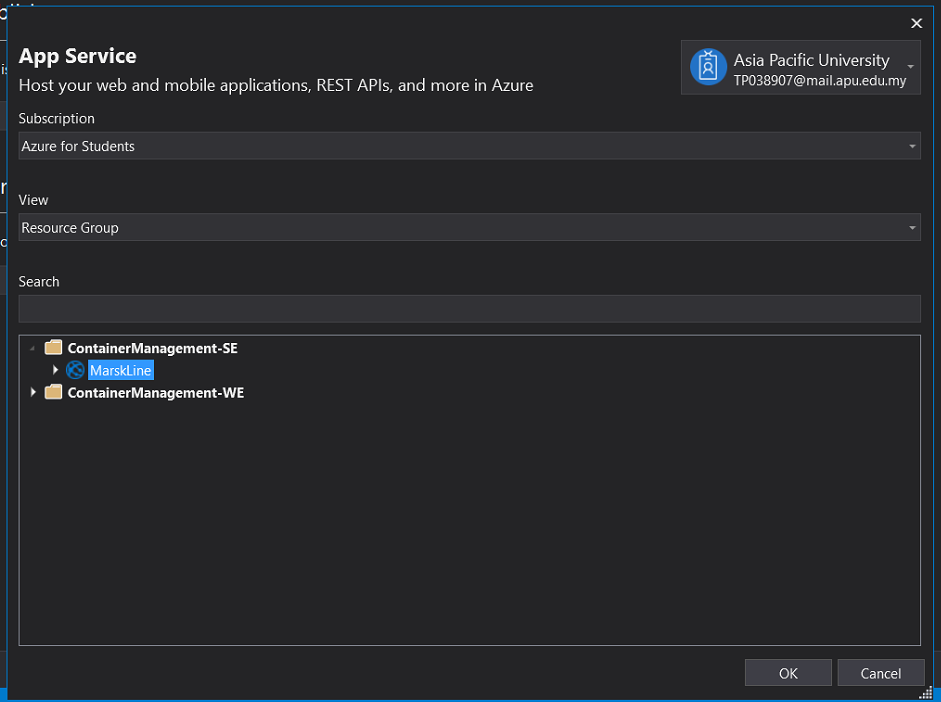
For the service plan, the Standard plan is required for the Maersk Line business. The pricing tier has enough requirements and the support for the Scaling and for Traffic Manager which provide 50GC storage, 1.75 GB RAM and so on.

## 4.3 Azure publishing

After finished all the set up of the Microsoft Azure, the developer can publish the web application through Visual Studio 2017.



One of the benefit of using Visual Studio to publish web application to cloud azure is it is just need to right click the project solution and select the word publish.



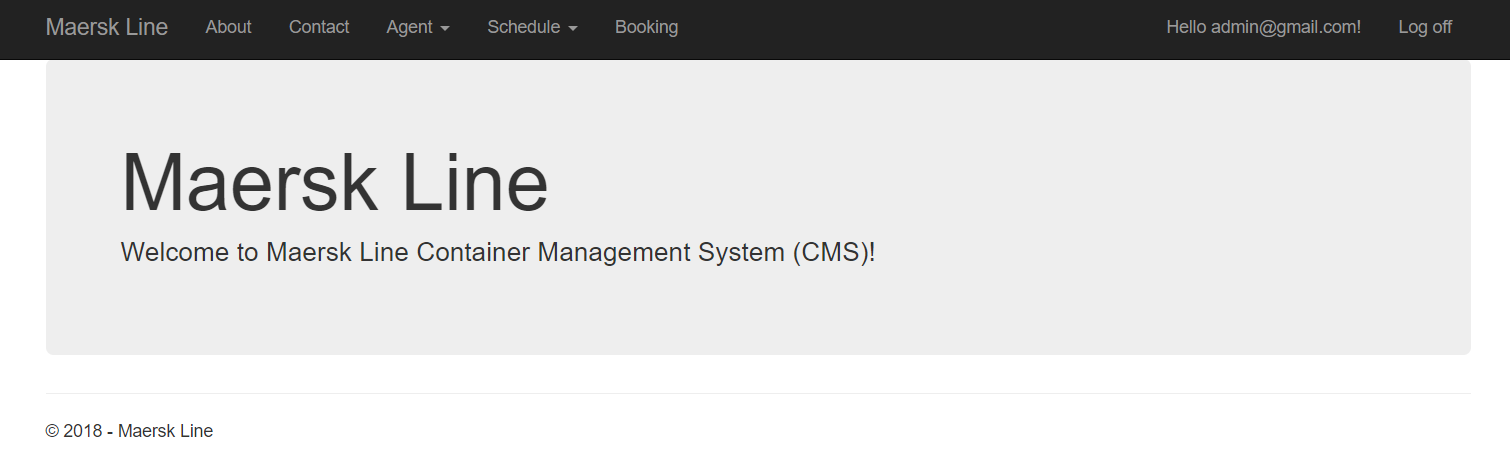
This image is after the developer select the publish icon, it will proceed to this page. Because the developer had create two app service for two region, therefore, there will be two file for different region app service. The developer chosen the MarskLine app service to publish which is the main focus region.

# 

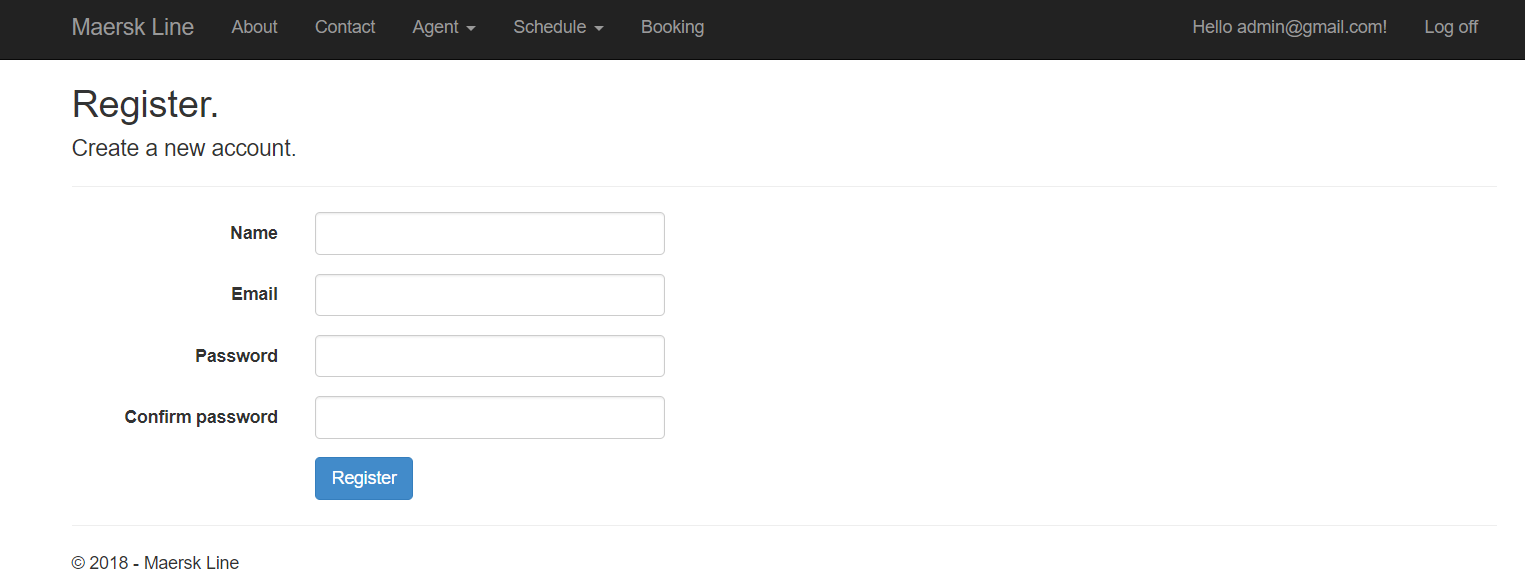
# After the user click on okay button, it will proceed to this window. The developer can click on the publish button to publish the web application.

# 

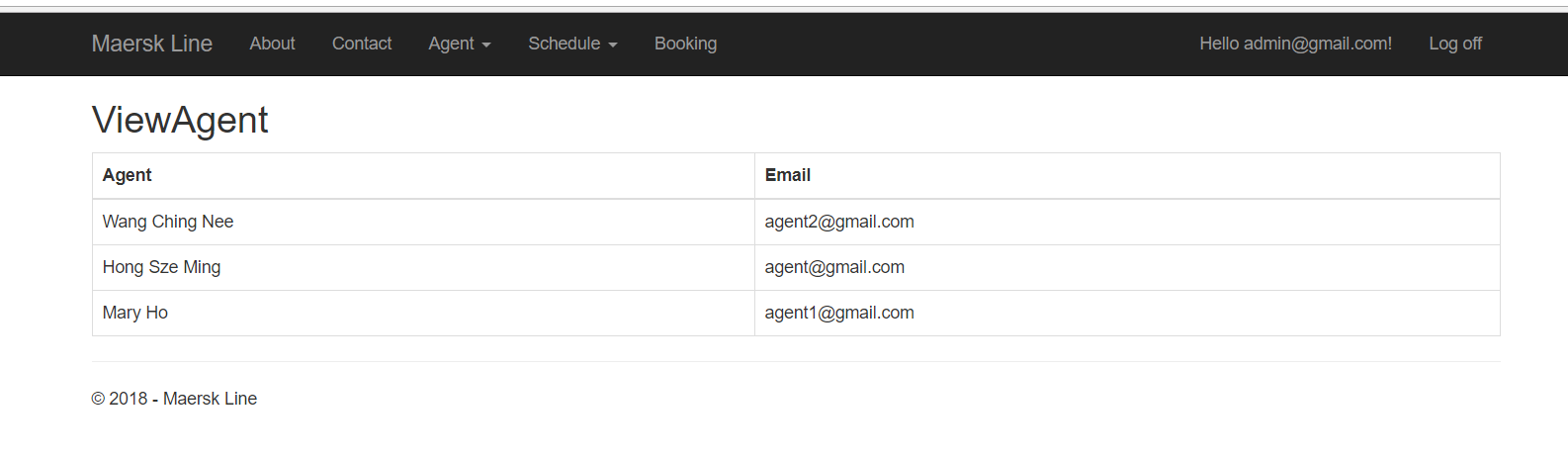
# This web application had successfully publish in cloud platform.

Admin home page

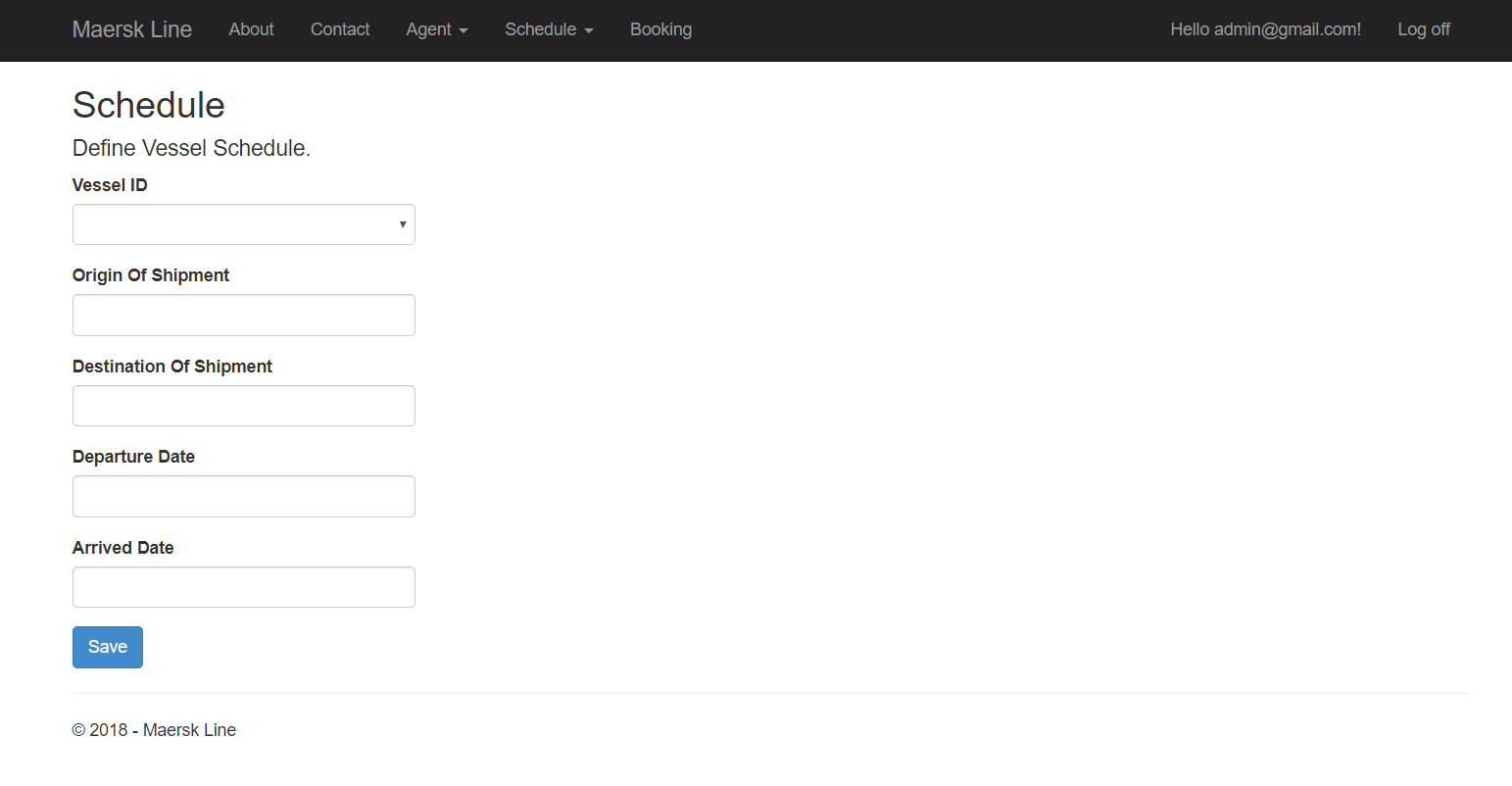
Register agent page



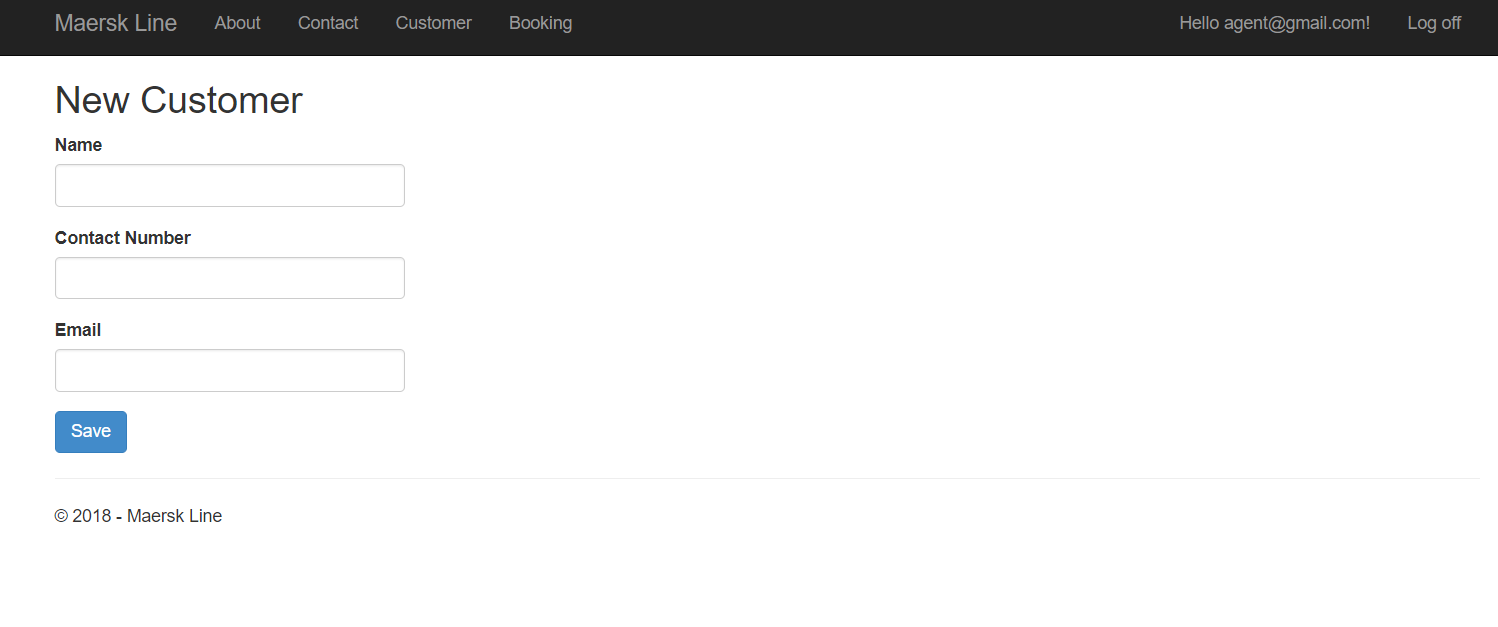
View Agent page



Schedule page



Customer Registration



# 5.0 Testing

## 5.1 Unit Testing

1) Login

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Title | Description | Expected Result | Actual Result |
|  | Email, | 1. Do not enter any value 2. Press Login button | Display error message “The Email field is required”. | Expected result |
|  | Password | 1. Do not enter any value 2. Press Login button | Display error message “The password field is required”. | Expected result |
|  | Email, Password | 1. Enter valid email. 2. Enter invalid password. 3. Press Login button | Display error message “Please enter the valid email or password”. | Expected result |
|  | Email,  Password | 1. Enter invalid email. 2. Enter valid password. 3. Press Login button | Display error message “Please enter the valid email or password”. | Expected result |
|  | Email, Password | 1. Enter invalid email. 2. Enter invalid password. 3. Press Login button | Display error message “Please enter the valid email or password” | Expected result |
|  | Email, Password | 1. Enter valid email. 2. Enter valid password. 3. Press Login button | Application display login user home page. | Expected result |

Register Agent

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Title | Description | Expected Result | Actual Result |
|  | Name | 1. Do not enter any value 2. Press Register button | Display error message “The Name field is required”. | Expected result |
|  | Email | 1. Do not enter any value 2. Press Register button | Display error message “The Email field is required”. | Expected result |
|  | Password | 1. Enter valid email. 2. Enter invalid password. 3. Press Register button | Display error message “The Password field is required”. | Expected result |
|  | Confirmation Password | 1. Enter invalid email. 2. Enter valid password. 3. Press Register button | Display error message “The password and confirmation password do not match”. | Expected result |
|  | Password, Confirmation Password. | 1. Enter invalid email. 2. Enter invalid password. 3. Press Register button | Display error message “The password and confirmation password do not match”. | Expected result |
|  | Name, Email, Password, Confirmation Password. | 1. Enter all valid data for required field. 2. Press Register button. | Display message “Created successfully” on register agent page. | Expected result |

View Agent

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Function | Description | Expected Result | Actual Result |
|  | View Profile | 1. Login as Admin user. 2. Click on the drop-down Agent icon. 3. Click on the View Agent icon. | Application display the all agents information. | Expected result |

Create Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Title | Description | Expected Result | Actual Result |
|  | Vessel Id | 1. Do not select any id from drop down list 2. Press Save button | Display error message “The Vessel Id field is required”. | Expected result |
|  | Origin of shipment | 1. Do not enter any value 2. Press Save button | Display error message “The Original of Shipment field is required”. | Expected result |
|  | Destination of shipment | 1. Do not enter any value 2. Press Save button | Display error message “The Destination of shipment field is required”. | Expected result |
|  | Departure date | 1. Do not select any date from the date picker 2. Press Save button | Display error message “The Departure date field is required”. | Expected result |
|  | Arrived date | 1. Do not select any date from the date picker 2. Press Save button | Display error message “The Arrived date field is required”. | Expected result |
|  | Arrived date | 1. Select a date from the date picker. 2. Press Save button. | Display message “Please select the data is greater than the departure data”. | Expected result |
|  | Vessel Id, Origin of shipment, Destination of shipment, Departure date, Arrived date | 1. Enter all valid data. 2. Press Save button | Display message “Created successful” |  |

View Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Function | Description | Expected Result | Actual Result |
|  | View Schedule | 1. Login as Admin user. 2. Click on the drop-down Schedule icon. 3. Click on the View Schedule icon. | Application display the all schedule information. | Expected result |

View Booking

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Function | Description | Expected Result | Actual Result |
|  | View Booking | 1. Login as Admin user. 2. Click on the Booking icon | Application display the all booking information. | Expected result |

Customer Registration

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Parameter | Description | Expected Result | Actual Result |
|  | Name | 1. Do not enter any value 2. Press Save button | Display error message “The Name field is required”. | Expected result |
|  | Contact Number | 1. Do not enter any value 2. Press Save button | Display error message “The Contact Number field is required”. | Expected result |
|  | Email | 1. Enter invalid email. 2. Enter valid password. 3. Press Save button | Display error message “The Email field is required”. | Expected result |
|  | Contact Number | 1. Enter alphabet value into the contact number field. 2. Press save button | Display error message “The Contact Number field is not a valid phone number.”. | Expected result |
|  | Email | 1. Enter value that not follow email format. 2. Press save button | Display error message “The Contact Number field is not a valid e-mail address.”. | Expected result |
|  | Name, Contact Number, Email | 1. Enter all the valid data. 2. Press save button. | Display message “Created successfully” on Register Customer page. | Expected result |

Search Vessel

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Parameters | Description | Expected Result | Actual Result |
|  | Name | 1. Do not enter any value 2. Press Register button | Display error message “The Name field is required”. | Expected result |
|  | Email | 1. Do not enter any value 2. Press Register button | Display error message “The Email field is required”. | Expected result |
|  | Password | 1. Enter valid email. 2. Enter invalid password. 3. Press Register button | Display error message “The Password field is required”. | Expected result |
|  | Confirmation Password | 1. Enter invalid email. 2. Enter valid password. 3. Press Register button | Display error message “The password and confirmation password do not match”. | Expected result |
|  | Password, Confirmation Password. | 1. Enter invalid email. 2. Enter invalid password. 3. Press Register button | Display error message “The password and confirmation password do not match”. | Expected result |
|  | Name, Email, Password, Confirmation Password. | 1. Enter all valid data for required field. 2. Press Register button. | Display message “Created successfully” on register agent page. | Expected result |

Display Vessel Result

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Function | Description | Expected Result | Actual Result |
|  | Display search result from the Search Vessel page | 1. Click on the Search button after entering required data. | Display all the related information about vessel and schedule. | Expected |

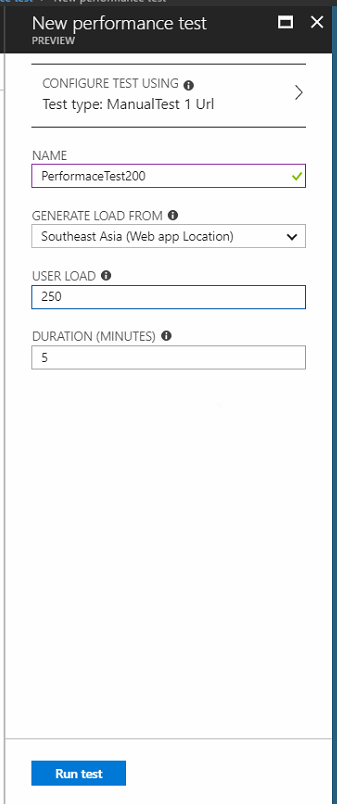
Booking Form

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Function/ Parameters | Description | Expected Result | Actual Result |
|  | Display related data of schedule. | 1. After Agent press the search button. | Display booking form which contained the vessel and schedule related data and a form for booking. | Expected result |
|  | Item | 1. Do not enter any value 2. Press Save button | Display error message “The Item field is required.” | Expected result |
|  | Customer | 1. Do not enter any value 2. Press Save button | Display error message “The Customer field is required.” | Expected resu |
|  | Space | 1. Select one number from the drop down list 2. Press Save button | Display error message “The space is full, please selected another vessel.” | Expected result |
|  | Item, Customer, Space | 1. Enter all valid data 2. Press Save button | Display message “Book successful” | Expected result |

## 5.2 Performance Testing

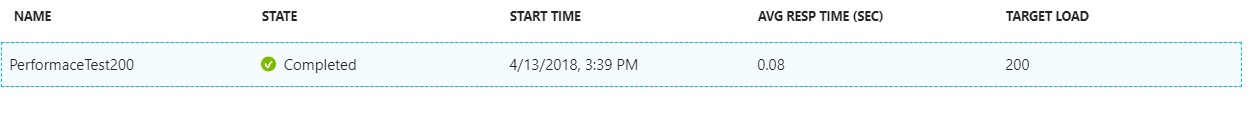
The performance testing of web application can perform by Azure, which allow users to perform testing on the web application. Besides, it can also can allow to understand the performance of web application in difference region.

This feature can search in the app service which click on the desired app service and find the word “Performance Test”.

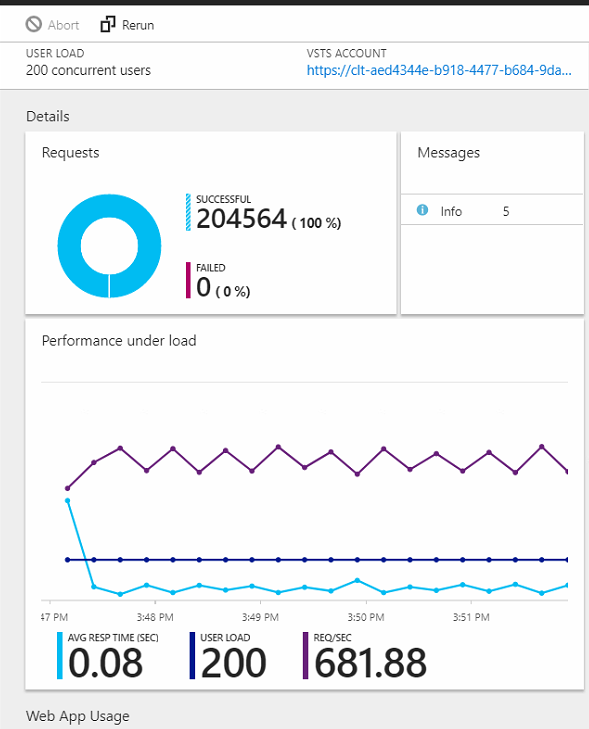


The user can select how many user load for the web application, in this test performance the developer had chosen 200 for the user load.

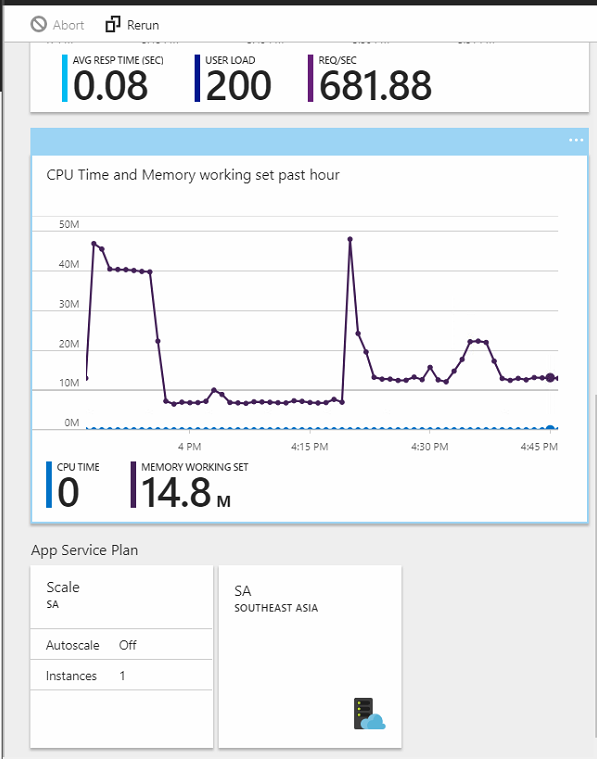
This performance test take about 15 minutes to completed.



The result will show on the pie chart whether the performance test fail or not.



The performance result show as above and below.



# 6.0 Conclusion

As a conclusion, the Container Management System(CMS) has been deploy in Microsoft Azure. Microsoft Azure is a new service for the developer used, therefore, it is new experience and challenges for the developer while the developer also can gain new knowledge and skill about this area which is more understand about Microsoft Azure. The CMS web application has been deploying by the developer to the Microsoft Azure to improve the supply chain performance and can achieve the goal of reduce the operation cost in overall supply chain management. There are few tools and services that provided by the Microsoft Azure had been utilized by the developer to deploy the CMS web application to the cloud platform which are the Azure web service application, SQL server, SQL database, performance test and so on to allow the web application can have improve it performance in supply chain management.

# 7.0 References

Docs.microsoft.com. (2018). *What is Traffic Manager*. [online] Available at: https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview [Accessed 13 Apr. 2018].