



DESIGNING & DEVELOPING CLOUD APPLICATIONS
INDIVIDUAL ASSIGNMENT

Module Code	:	CT071-3-5-3-DDAC
Student ID	:	TP035572
Student Name	:	Ahmed Nabil Ahmed Al-Eryani
Intake Code	:	UC3F1706SE
Project Title	:	Ukraine International Airlines (UIA)
Lecturer Name	:	DR. KALAI ANAND RATNAM
Hand in Date	:	9 th October 2017

Table of Contents

Acknowledgement	3
1.0 Introduction	4
1.1 Project Background	4
1.2 Project Objective.....	6
1.4 Project Specification	6
1.5 Project Deliverables	7
2.0 Project Plan	8
2.1 Gantt chart	8
3.0 Design.....	9
3.1 Architectural Design.....	9
3.2 Use Case Diagram	10
4.0 Implementation	11
4.1 Creating Web App & SQL Database	11
4.2 Publishing Web Application	14
4.3 Traffic Manager.....	16
4.4 Web Application Screenshot.....	18
4.4.1 Home page	18
4.4.2 Register	19
4.4.3 Login.....	19
4.4.4 User Profile Page.....	20
4.4.5 Flights.....	20
4.4.6 Booking	21
4.4.7 Checkout	21
4.4.8 FAQ.....	22
4.4.9 Flight Prices[MYR]	22
4.4.9.1 Flight Prices[USD].....	23
5.0 Test plan.....	24
5.1 Performance Test.....	24
5.2 Functional Test.....	28
6.0 Conclusion	31
References	32
Appendices.....	33

Acknowledgement

I would like to thank to my family for their kind support and reinforcement that help me complete this project. I also would like to thank my friends who offered to help me with their most effort. Also, I would like to extend my gratitude to DR. KALAI ANAND RATNAM for his endless advices on enchanting inspired ideas and more alternatives to make this assignment properly and for his supervision, his suggestions have made my assignment more professional and without his encouragement and the dedication of time, my assignment would not have been a success.

1.0 Introduction

1.1 Project Background

Back in 1992, Ukraine International was co-founded as an international carrier of independent Ukraine by the Ukrainian State Association of Civil Aviation and GPA (thereafter AerCap B.V.), the world's largest aircraft lessor.

Over the 15-year period, the airline has attracted top class strategic and financial investors. In 1995, the Ukrainian Government shareholding was transferred to the State Property Fund. In 1996, Austrian Airlines and Swissair became shareholders with the European Bank for Reconstruction and Development, EBRD, joining shortly – in 2000.

Organizational model of a public-private entity empowered UIA to take advantage of the strengths of all shareholders, to seize a vast selection of business opportunities, as well as to adopt best maintenance, operation, service, and management practices of its international partners.

Initially, UIA was meant to establish non-stop operations between Ukraine and Western Europe and successfully implement the strategy of a point-to-point carrier.

By its 18th anniversary in 2009, UIA became one of Ukraine's key aviation market players with 20% business segment. However, further development required the UIA business strategy revision. The latter coincided with the change of the company's ownership in 2010 and was triggered by international investors' withdrawal from the equity fueled by the shift in their market strategies in Ukraine and expiration of the EBRD investment conventional participation period.

Amid challenging operating environment and intense price-based competition, privatization guaranteed the UIA business flexibility and allowed commencing transformation from a point-to-point into a network carrier.

In 2013, following the business collapse of the main competitor, UIA turned out to be the sole carrier capable of averting market infrastructure collapse. The transition process turned out to be much more dynamic than it had been planned initially. Over twelve months, UIA had to boost operations by doubling fleet and enhancing the team. The latter empowered the airline to resume operations to most destinations "inherited" from the competitor carrier.

Involuntary expansion jump-started the UIA operations' optimization and prioritized the development of Ukraine's transit potential. UIA built up its operations so as to generate

and direct transit passenger traffic from the North to the South and from West to the East via its hub, Kiev Boryspil International Airport.

In 2014, UIA faced the challenge of stabilizing its business amid profound political and economic crisis, devaluation of the national currency, and dramatic decline in effective demand. For the first time ever, UIA had to take drastic contingency measures and optimize its staff, fleet, and route network.

Meantime, to sustain the business and guarantee passengers the service availability, the UIA Management Team decided to modify the operating model and commence systematically decreasing fares by excluding additional services from the ticket price. Eventually, the latter empowered UIA to offer clients competitive fares and become the world's first network low-fare carrier.

In an effort to support further business growth and increase organizational flexibility, Ukraine International Airlines (UIA), is looking at designing and developing an Online Flight Booking System. UIA Management Team looked at both Microsoft Azure and Amazon Web Services and chose Azure.

1.2 Project Objective

1.3 Project Scope

Ukraine International Airlines (UIA) Online Flight Booking System will involve planning, designing, developing and testing of the web application based on the requirements defined by Dmitriy Prudnikov.

The project is broken down into parts and the completion date for each part is recorded by constructing a Gantt chart. The project will be concluded when the web application is tested and deployed to Microsoft Azure. At the end of this project, a documentation that include project plan, web application design, implementation and test plan will be produced.

1.4 Project Specification

Provisioning: To ensure that the Online Flight Booking System web application along with the SQL Server database to the Microsoft Azure Platform.

Maintainability: To ensure the web application is able to be upgraded and other maintenance tasks are able to be performed while multiple users are using it.

Monitoring: To ensure the web application is able to be monitored at all times, including how each user is using the application, in order to identify any problems and to troubleshoot them.

Availability: To ensure the web application is constantly available to users and to ensure the activities of other tenants do not affect the availability of the application.

Scalability: To ensure the web application scales to meet the demand of the application.

Functions provided by Microsoft Azure, which have been **incorporated** into the project:

- Deployment on the cloud
- SQL database
- Traffic Manager
- Performance testing
- Investigation and Analysis of application
- Application Scaling

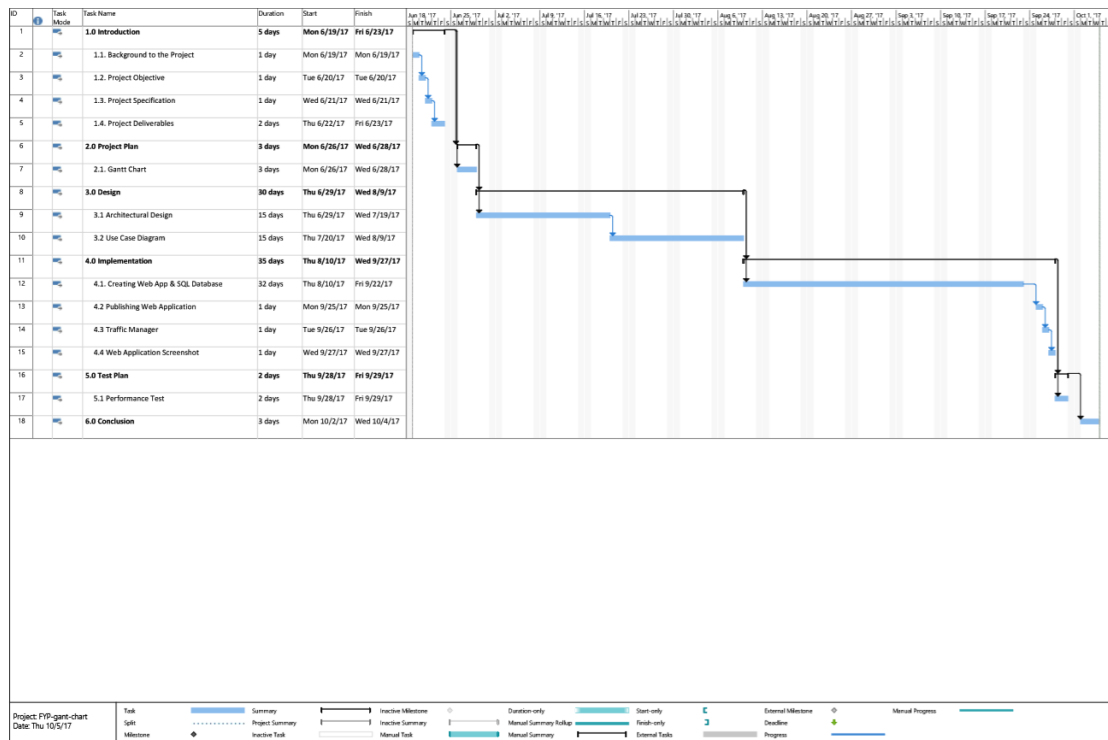
1.5 Project Deliverables

The UIA Online Flight Booking System web application will be able to access by the browsers such as Microsoft Edge, Google Chrome, Mozilla Firefox and Internet Explorer. It allows the users to perform the following criteria with the focus on performance and reliability:

- Manage your entire booking process.
- Creation of customer profile as well as modifying it.
- Based on where the customers are located and the type of trip they are booking the online booking engine will returns customized information.
- Protection from denial-of-service (DOS) attacks.
- Allow visitors from many parts of the world.

2.0 Project Plan

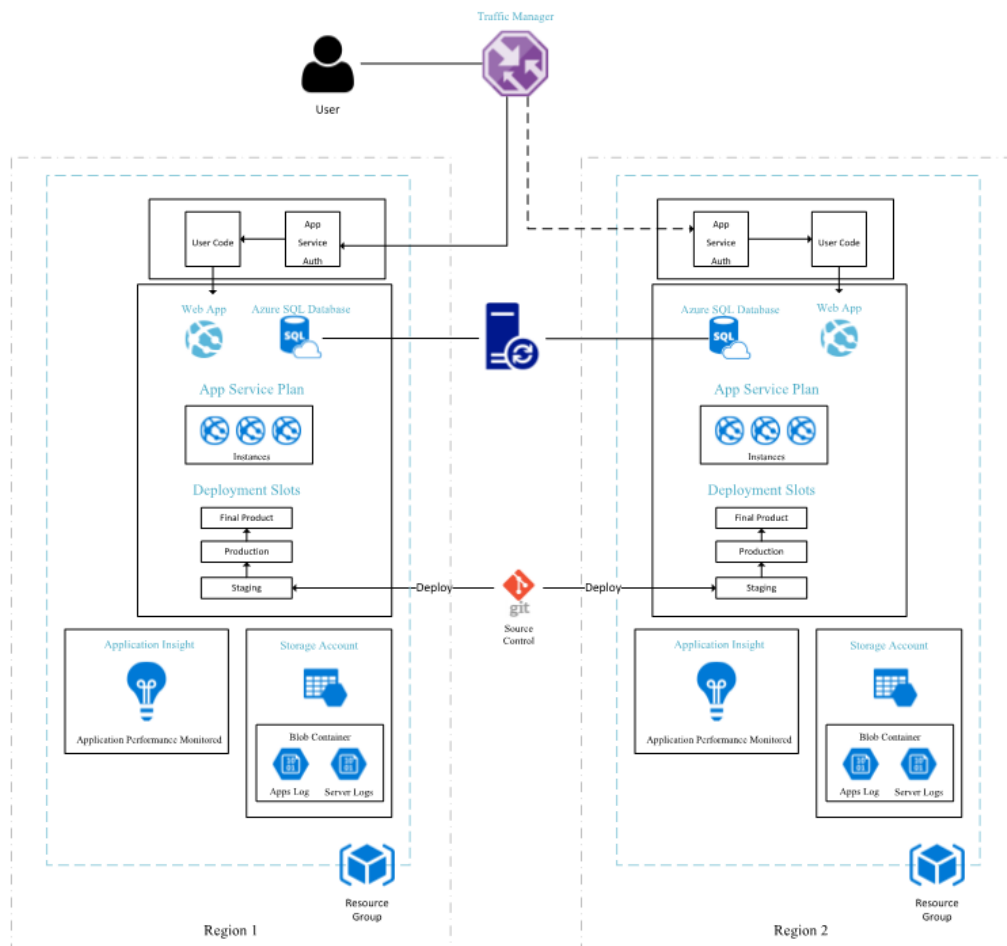
2.1 Gantt chart



ID	Task Mode	Task Name	Duration	Start	Finish
1		1.0 Introduction	5 days	Mon 6/19/17	Fri 6/23/17
2		1.1. Background to the Project	1 day	Mon 6/19/17	Mon 6/19/17
3		1.2. Project Objective	1 day	Tue 6/20/17	Tue 6/20/17
4		1.3. Project Specification	1 day	Wed 6/21/17	Wed 6/21/17
5		1.4. Project Deliverables	2 days	Thu 6/22/17	Fri 6/23/17
6		2.0 Project Plan	3 days	Mon 6/26/17	Wed 6/28/17
7		2.1. Gantt Chart	3 days	Mon 6/26/17	Wed 6/28/17
8		3.0 Design	30 days	Thu 6/29/17	Wed 8/9/17
9		3.1 Architectural Design	15 days	Thu 6/29/17	Wed 7/19/17
10		3.2 Use Case Diagram	15 days	Thu 7/20/17	Wed 8/9/17
11		4.0 Implementation	35 days	Thu 8/10/17	Wed 9/27/17
12		4.1. Creating Web App & SQL Database	32 days	Thu 8/10/17	Fri 9/22/17
13		4.2 Publishing Web Application	1 day	Mon 9/25/17	Mon 9/25/17
14		4.3 Traffic Manager	1 day	Tue 9/26/17	Tue 9/26/17
15		4.4 Web Application Screenshot	1 day	Wed 9/27/17	Wed 9/27/17
16		5.0 Test Plan	2 days	Thu 9/28/17	Fri 9/29/17
17		5.1 Performance Test	2 days	Thu 9/28/17	Fri 9/29/17
18		6.0 Conclusion	3 days	Mon 10/2/17	Wed 10/4/17

3.0 Design

3.1 Architectural Design



The figure illustrates architectural diagram of the project. As it can be observed from the diagram above, once the user tries to access the application using internet, the traffic manager identifies user's location and based on the performance and the region where he is trying to connect from the routing method redirects that user to the closest and most reliable web application server.

Furthermore, there are two web application services connected to the SQL database provided by Microsoft Azure Cloud services, and the web application has in-built function to determine the region as well which identify the user location access and then showing the currency accordingly.

3.2 Use Case Diagram



Functions of the web application system users:

- Function to Register/Login/Logout Users
- Function to View/Book/Purchase Flights Tickets
- Function to Book a Flight Seat
- Function to View/Edit User Profile

4.0 Implementation

4.1 Creating Web App & SQL Database

The screenshot shows the Microsoft Azure portal interface for creating a new resource group. The left navigation pane lists various Azure services, with 'Resource groups' selected. The main content area is titled 'Resource groups' and shows a list of existing resource groups under the 'Visual Studio Dev Essentials' subscription. The right pane displays the 'Resource group' creation form, which includes fields for the resource group name, subscription, and location. The 'Subscription' is set to 'Visual Studio Dev Essentials' and the 'Resource group location' is set to 'Southeast Asia'. A 'Create' button is visible at the bottom right of the form.

NAME	
Deployed	...
UIA_RESOURCE	...
Uiaresource2	...
VS-clt-2aea12df-4ee9-4b21-...	...

Resource group creation form fields:

- * Resource group name:
- * Subscription:
- * Resource group location:

☐ Pin to dashboard

Create

Diagram 4.1 (a): Create a Resource Group

A resource group must be created before deploying resources such as Web App and SQL. In order to create new resource group, click on the resource group icon at the left navigation bar,

then click on Add, then complete the process by filling in the information required. The name given for the resource group must not be the same as the name of existing resource group.

The screenshot shows the Microsoft Azure portal interface for creating a new SQL Database. The left sidebar contains a navigation menu with various services. The main area is divided into two panes. The left pane shows the 'SQL databases' section with an 'Add' button and a list of existing resource groups. The right pane shows the configuration details for the new database.

Configuration Details:

- Database name:** Enter database name
- Subscription:** Visual Studio Dev Essentials
- Resource group:**
 - ☐ Create new
 - ☒ Use existing
 - UIA_RESOURCE
- Select source:** Blank database
- Server:** uia-db (Southeast Asia)
- Want to use SQL elastic pool?**
 - ☐ Yes
 - ☒ Not now
- Pricing tier:** Standard S2: 50 DTU, 250 GB
- Collation:** SQL_Latin1_General_CP1_CI_AS

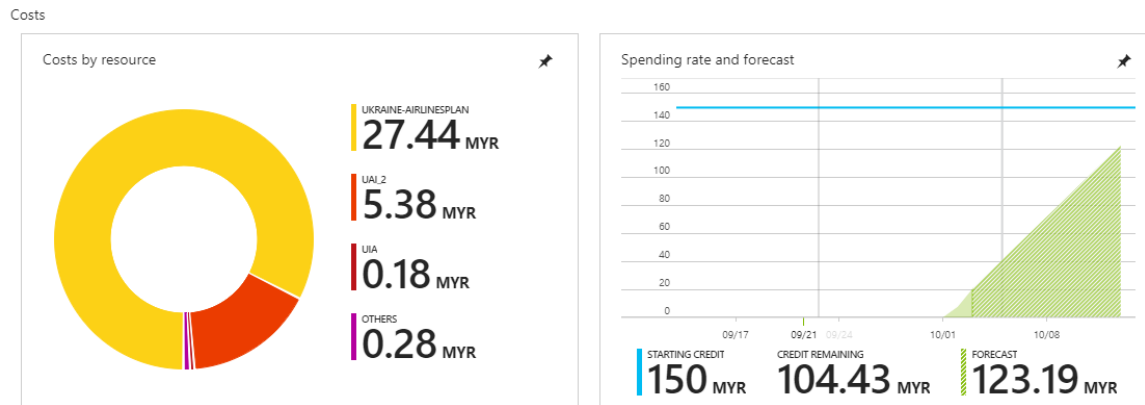
At the bottom right, there is a 'Pin to dashboard' checkbox and a 'Create' button.

Diagram 4.1 (b): Deploy Web App + SQL

After the resource group created, the developer will redirect to the dashboard page. Next, select the created resource group at the dashboard and click on the Add. Select Web App + SQL which falls under the category of Web to deploy it to the resource group which has been

created previously. Diagram 4.2 shows the fields that is necessary to be completed in order to create Web App + SQL.

The App name entered must be unique as it will represent the URL of the web app. The App name of this project is ukraine-airlines and the location of the Web App to be deployed is in Southeast Asia with the URL “http://ukraine-airlines.azurewebsites.net”. Next, the App Service Plan chosen for this Web App is the S0 Standard which is a standard service plan as



the budget allocated is 150 MYR which will be enough as traffic manager will be implemented with two endpoints one on Southeast Asia and one on West Europe as well as two web applications for two different nodes connected with synced database.

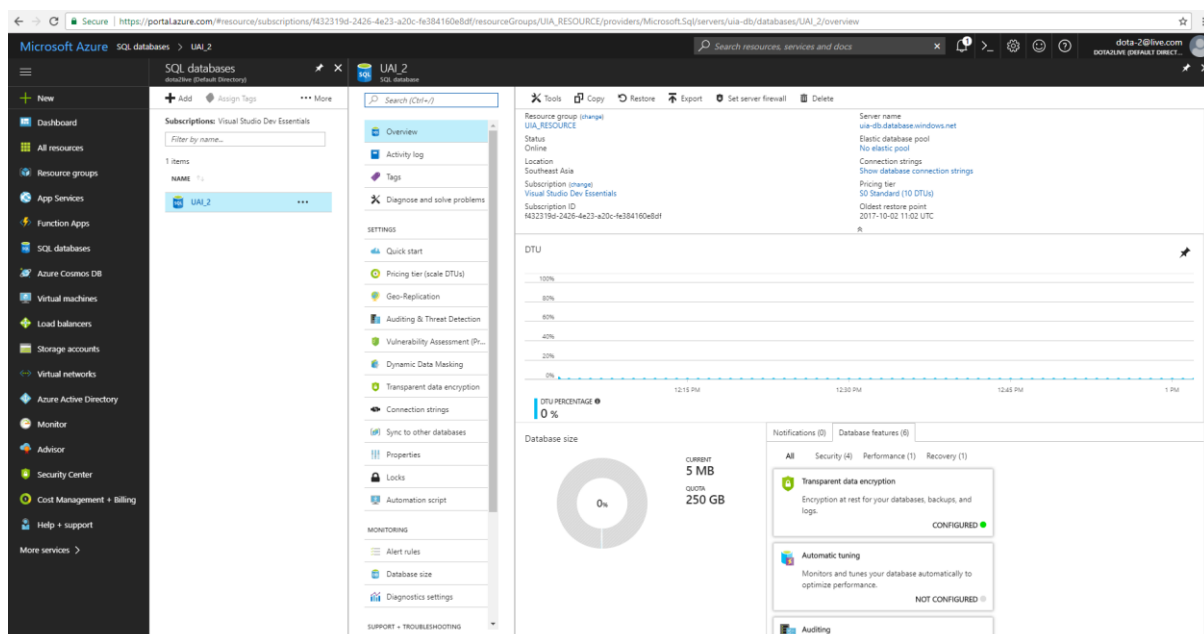


Diagram 4.1 (c): Create SQL Server

Diagram 4.3 shows the page of SQL Server is created to host the SQL Server Database which is created to store information such as user’s details, booking details, flight details etc... Firstly,

click Create new server to open New Server tab. Then, enter a valid name for the server in the Server name text field. Besides, the password field is also required to be filled. Lastly, Southeast Asia is selected at the Location text box and then click Select to create SQL Server.

4.2 Publishing Web Application

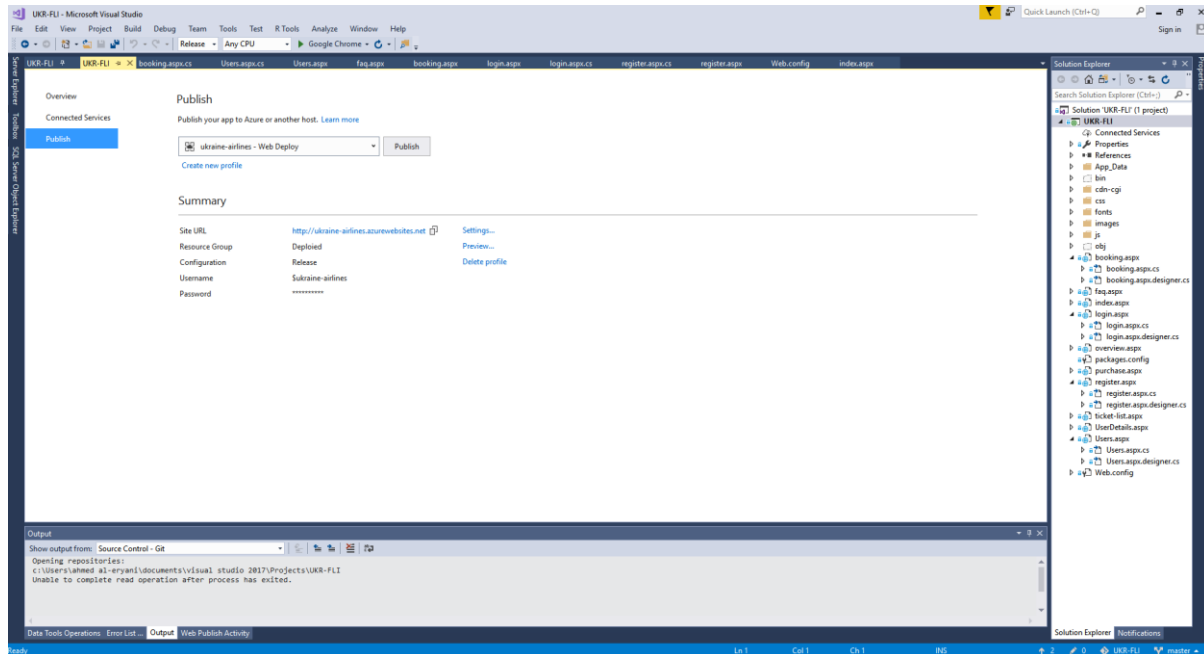


Diagram 4.2 (a): Publish Web Application

The developed System web application can be easily published from Visual Studio 2017. Firstly, right click the solution at the Solution Explorer section and click Publish. Then choose the Microsoft Azure App Service at the profile tab.

The screenshot shows the 'Publish' dialog box in Visual Studio. The title bar is blue with the text 'Publish' and standard window controls. The main area has a blue header with a globe icon and the word 'Publish'. Below this, there are two tabs: 'Connection' (selected, highlighted in blue) and 'Settings'. The 'Connection' tab displays the title 'ukraine-airlines - Web Deploy *'. It contains several fields: 'Publish method:' with a dropdown menu set to 'Web Deploy'; 'Server:' with the text 'ukraine-airlines.scm.azurewebsites.net:443'; 'Site name:' with the text 'ukraine-airlines'; 'User name:' with the text 'ukraine-airlines'; 'Password:' with a masked field of dots and a checked 'Save password' checkbox; and 'Destination URL:' with the text 'http://ukraine-airlines.azurewebsites.net'. A 'Validate Connection' button is located below the 'Destination URL' field. At the bottom of the dialog are four buttons: '< Prev', 'Next >', 'Save' (highlighted in blue), and 'Cancel'.

Diagram 4.2 (b): Connection

Once the publish target is selected and specified, click on the Validate Connection button in the Connection tab shown in Diagram 4.5. The purpose of validating the connection is to check existence of the Microsoft Azure Account by validating credentials such as Server, Site Name, Username and Password. A green tick will appear beside the Validate Connection after validation completed.

4.3 Traffic Manager

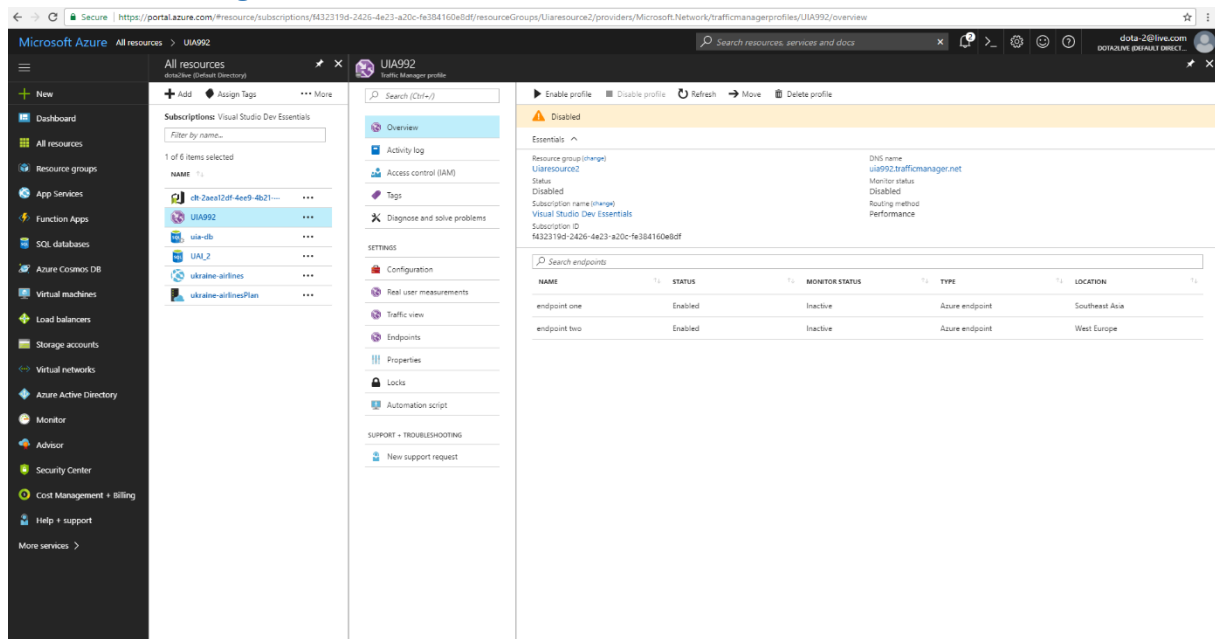


Diagram 4.3 (a): Create Resource Group

Before creating a Traffic Manager Profile, a resource group is required to be created. Next, enter “Traffic Manager Profile” in the search bar located at the top of the Microsoft Azure Portal Page. Then click Add to add new traffic manager profile. It is required to complete all the fields shown in Diagram 4.7 to create traffic manager profile. The name of the traffic manager is specified, the chosen routing method for the traffic manager is performance. After all the fields are filled, click Create to create a new traffic manager profile.

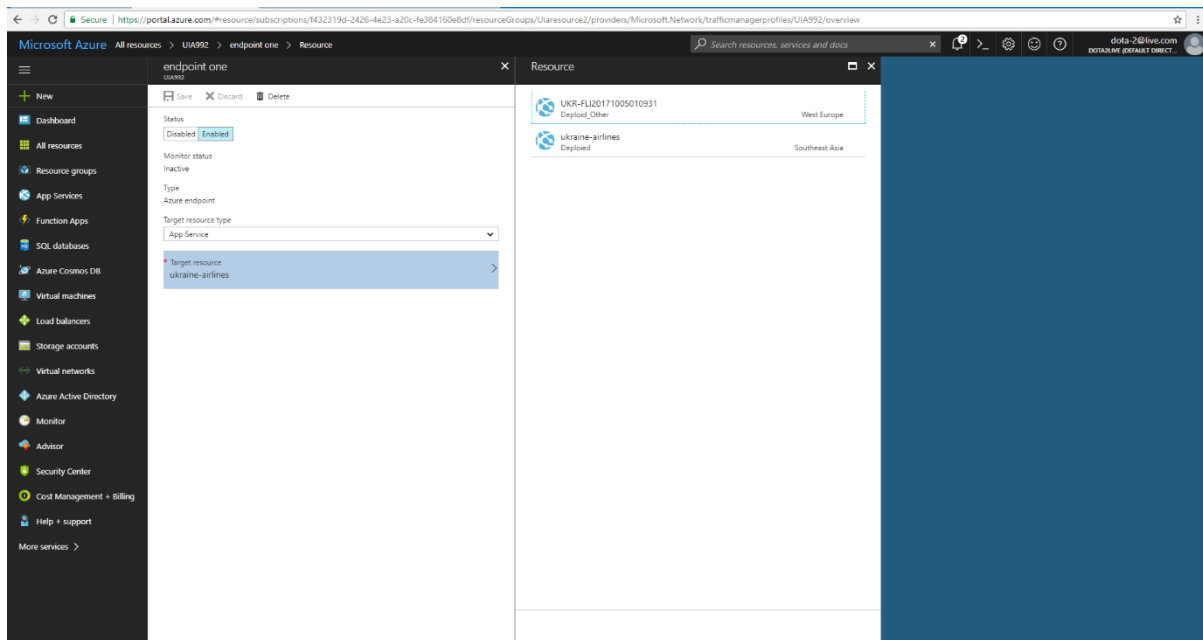


Diagram 4.3 (b): Create a Traffic Manager Profile

After creating traffic manager profile, select the endpoint option to deploy endpoints. By deploying endpoints in two or more locations across the world can improve the responsiveness of the web application by routing traffic to the location that is closest to the client. Diagram 4.8 shows the endpoint page. Firstly, select the type of endpoint, then specify the name of the endpoint. Furthermore, it is required to specify the type of target resource as App Service. Lastly, the developer was required to select an appropriate target resource from a list of App Service that is created.

4.4 Web Application Screenshot

4.4.1 Home page

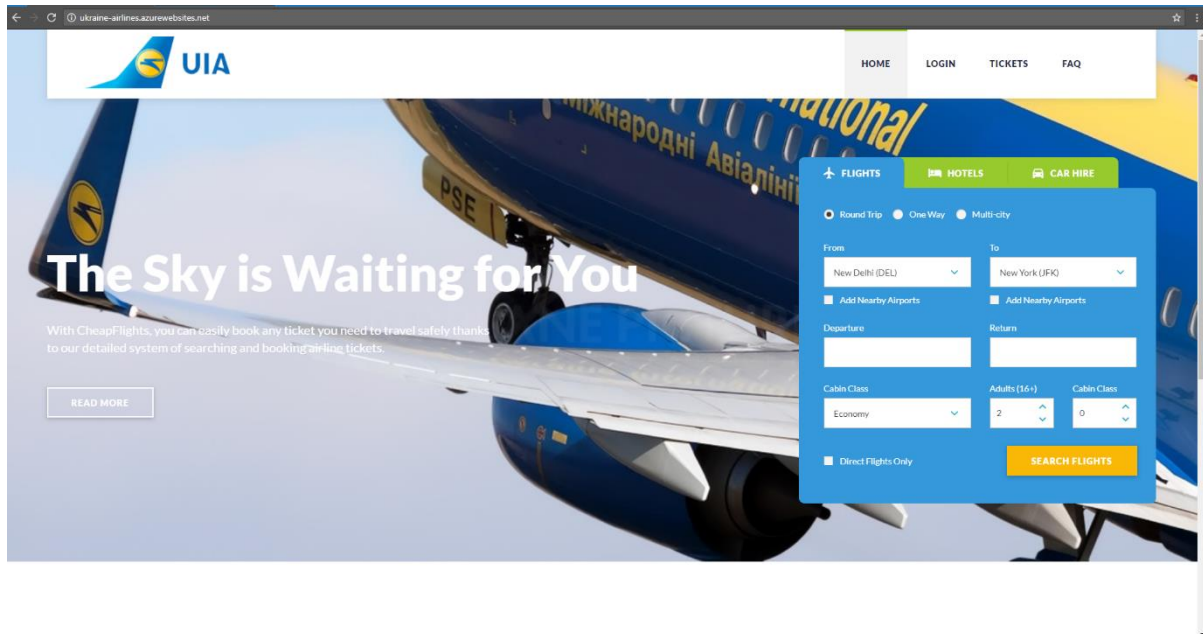


Diagram 4.4 (a): Home page

Diagram 4.4(a) shows the Home Page of the web application of Ukraine International Airlines (UIA) that allow the users to manage the functionalities. The users can view the Flights, FAQ, and Login page by clicking the navigation bar. Register and Log in are also located under LOGIN at the navigation bar also the user can fast search for nearest flights as well as booking a hotel and hiring a care however the last two features are still coming soon. In additional the design that being used is downloaded from a website that provide static html templates which is afterwards used for this project.

4.4.2 Register

Diagram 4.4(b): Register

Diagram 4.4(b) shows the registration page of the web application. The users are required to register themselves before they access into the web application. The user is required to enter details such as e-mail, username, password, passport id, country etc... to register themselves.

4.4.3 Login

Diagram 4.4(c): Login

Diagram 4.4(c) shows the Login page of the web application. After the user has registered, the user will input the registered e-mail/username and password in order to login into the web application.

4.4.4 User Profile Page

Profile Settings

E-Mail:

Username:

Password:

Full Name:

Country:

Date of Birth:

IC Passport ID:

Diagram 4.4 (d): Customer page

Diagram 4.4 (d) shows the user profile page of the web application. In this page, a list of customers details is displayed and can be modified as well by clicking the “Modify”.

4.4.5 Flights

Departure: 15/10/2017 | Return: 22/10/2017 | Sort by: Price low to high

Flight	Departure	Arrival	Price
7:55am Malaysia → 6h 50m → 7:45pm India	7:55am	7:45pm	7 days from MYR 3230
9:50pm India → 11:30am (+1) Malaysia	9:50pm	11:30am (+1)	
7:55am Malaysia → 6h 50m → 7:45pm India	7:55am	7:45pm	7 days from MYR 3290
9:50pm India → 11:30am (+1) Malaysia	9:50pm	11:30am (+1)	
7:55am Malaysia → 6h 50m → 7:45pm India	7:55am	7:45pm	7 days from MYR 3222
9:50pm India → 11:30am (+1) Malaysia	9:50pm	11:30am (+1)	
7:55am Malaysia → 6h 50m → 7:45pm India	7:55am	7:45pm	7 days from MYR 3209
9:50pm India → 11:30am (+1) Malaysia	9:50pm	11:30am (+1)	
7:55am Malaysia → 6h 50m → 7:45pm India	7:55am	7:45pm	7 days from MYR 3216
9:50pm India → 11:30am (+1) Malaysia	9:50pm	11:30am (+1)	

Diagram 4.4 (e): Flights page

Diagram 4.4 (e) shows the flights page of the web application. In this page, a list of flights is displayed along with their details such as prices, date, time and other transit flights.

4.4.6 Booking

The screenshot shows the UIA booking interface. At the top, there's a header with the UIA logo and navigation links. Below the header is a large image of a UIA aircraft. The main content area is titled 'Malaysia to (Any)'. It features a search form with the following fields: 'From' (Malaysia), 'To' (India), 'Departure' (15/10/2017), 'Return' (22/10/2017), 'Cabin Class' (First Class), and 'Adults (16+)' (1). A 'SEARCH FLIGHT' button is located at the bottom right of the form.

Diagram 4.4 (f): Book page

Diagram 4.4 (f) shows the Book page of the web application. In this page, a list of booking is displayed along with the details such as origin of the country, the country going to, date of the depart, date of the return as well as the cabin class which can be First Class, Economy Class or Business Class and if other adults are coming with you. The user is able to select booking after providing the above information then by clicking “Search Flight”. Which will take us to a page similar to the flight page showed above then the user can proceed by pressing the “Select” button beside the required flight details.

4.4.7 Checkout

The screenshot shows the UIA checkout page. At the top, there's a header with the UIA logo and navigation links. Below the header is a large image of a UIA aircraft. The main content area is titled 'Malaysia To India'. It features a confirmation summary for a flight. The summary includes the following details:

Full Name	Passport ID	Depart Time	Return Time	Total Purchase Special Price
Ahmed Nabil Ahmed Al-Eryani	04099137	15/10/2017	22/10/2017	MYR 2844

Below the table, the text 'CHECKOUT MYR 2844' is displayed. The user's name 'Ahmed Nabil Ahmed Al-Eryani' and contact information '5111-2222-3333-4444' are also shown. A 'CHECKOUT' button is located at the bottom right of the form.

Diagram 4.4 (g) Checkout page

Diagram 4.4 (g) shows the checkout page of the web application. In this page, the user is viewing the summary of the flight details along with the details such as price, registered user

full name, date of the flight, Passport ID etc... The user able to confirm the booking by providing his/her credit card information following by clicking “Confirm” button.

4.4.8 FAQ

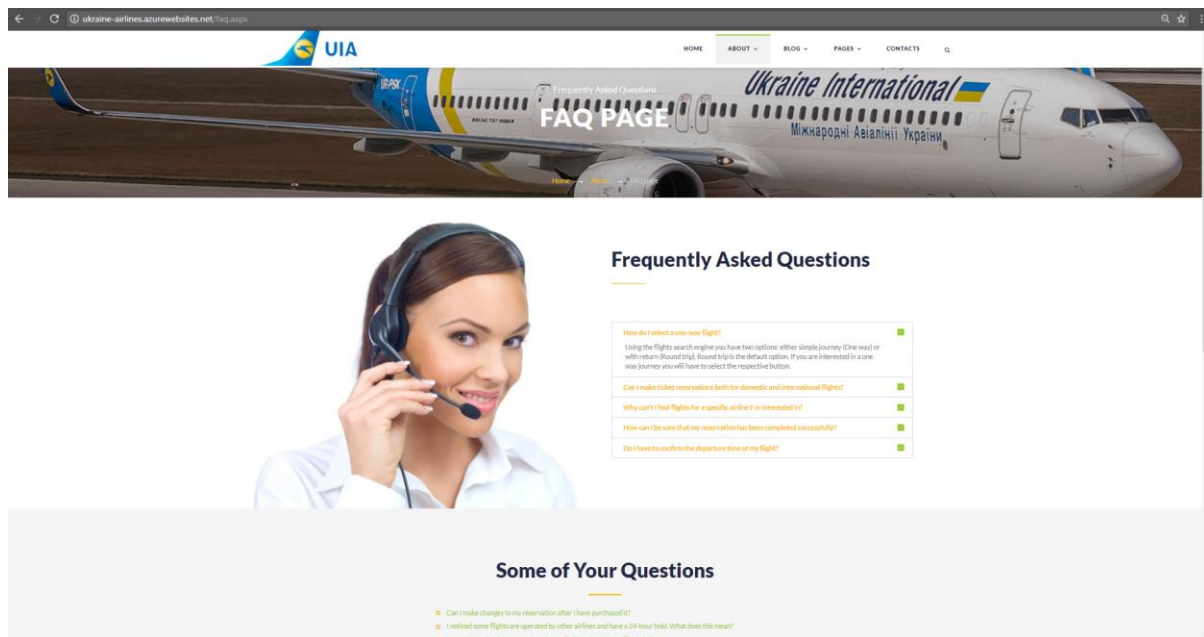


Diagram 4.5 (h) FAQ page

Diagram 4.4 (h) shows the Frequently asked questions page of the web application. In this page, a list of the most frequent questions asked is displayed along with the answers, Also the page shows how to contact the support team in case of inquiry or otherwise.

4.4.9 Flight Prices[MYR]

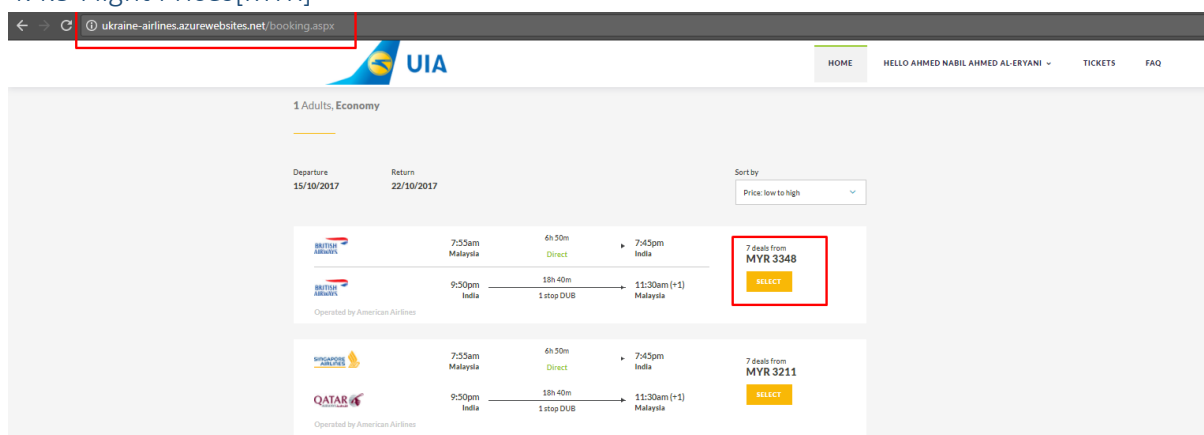


Diagram 4.5 (i) Flight Price(MYR)

Diagram 4.4 (i) demonstrate how the customized information based on the visitor location for example will affect the overall experience of the customers, as the users being routed through the traffic manager to the nearest node or end point the website has in-house function that can

determine or estimate the user's location which in return will show the currency based on that region/city. As you can see from the above screenshot the user visited the website and routed to the Asia region end point which in return showed the currency as MYR.

4.4.9.1 Flight Prices[USD]

The screenshot shows the UTA flight booking interface. At the top, the browser address bar displays 'ukr-fl20171005010931.azurewebsites.net/booking.aspx'. The website header includes the UTA logo and navigation links: HOME, HELLO AHMED NABIL AHMED AL-ERYANI, TICKETS, and FAQ. The main heading is 'Malaysia to India'. Below this, it specifies '1 Adults, Economy'. The flight details section shows a departure on 15/10/2017 and a return on 22/10/2017. A 'Sort by' dropdown menu is set to 'Price: low to high'. Two flight options are listed:

Airline	Departure (Malaysia)	Duration	Arrival (India)	Price
BRITISH AIRWAYS	7:55am	6h 50m Direct	7:45pm	7 deals from \$875
BRITISH AIRWAYS	9:50pm	18h 40m 1 stop DUB	11:30am (+1)	7 deals from \$878
SINGAPORE AIRLINES	7:55am	6h 50m Direct	7:45pm	7 deals from \$878
QATAR AIRWAYS	9:50pm	18h 40m 1 stop DUB	11:30am (+1)	7 deals from \$878

Both flight options are noted as 'Operated by American Airlines'. A red box highlights the '7 deals from \$875' and the 'SELECT' button for the British Airways flight.

Diagram 4.5 (j) Flight Price(USD)

Diagram 4.4 (j) demonstrate how the customized information based on the visitor location for example will affect the overall experience of the customers, as the users being routed through the traffic manager to the nearest node or end point the website has in-house function that can determine or estimate the user's location which in return will show the currency based on that region/city. As you can see from the above screenshot the user visited the website and routed to the US WEST end point which in return showed the currency as USD "\$".

5.0 Test plan

5.1 Performance Test

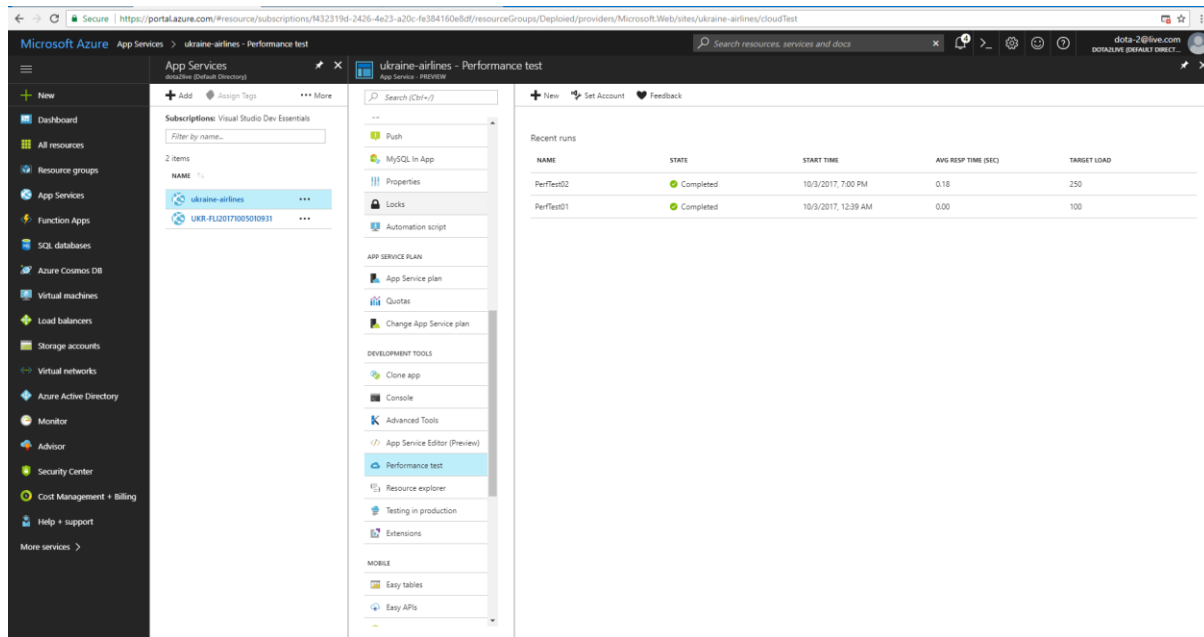


Diagram 5.1 (a): Create New Performance Test

A new performance test can be created by opening the web app which performance is required to be tested on. Next, select the Performance Test under the Development Tools section. After that, click New to create new performance test of the web app. After the New is clicked, a new tab as shown in Diagram 5.1 (a) is shown with all the blank fields are required to be completed.

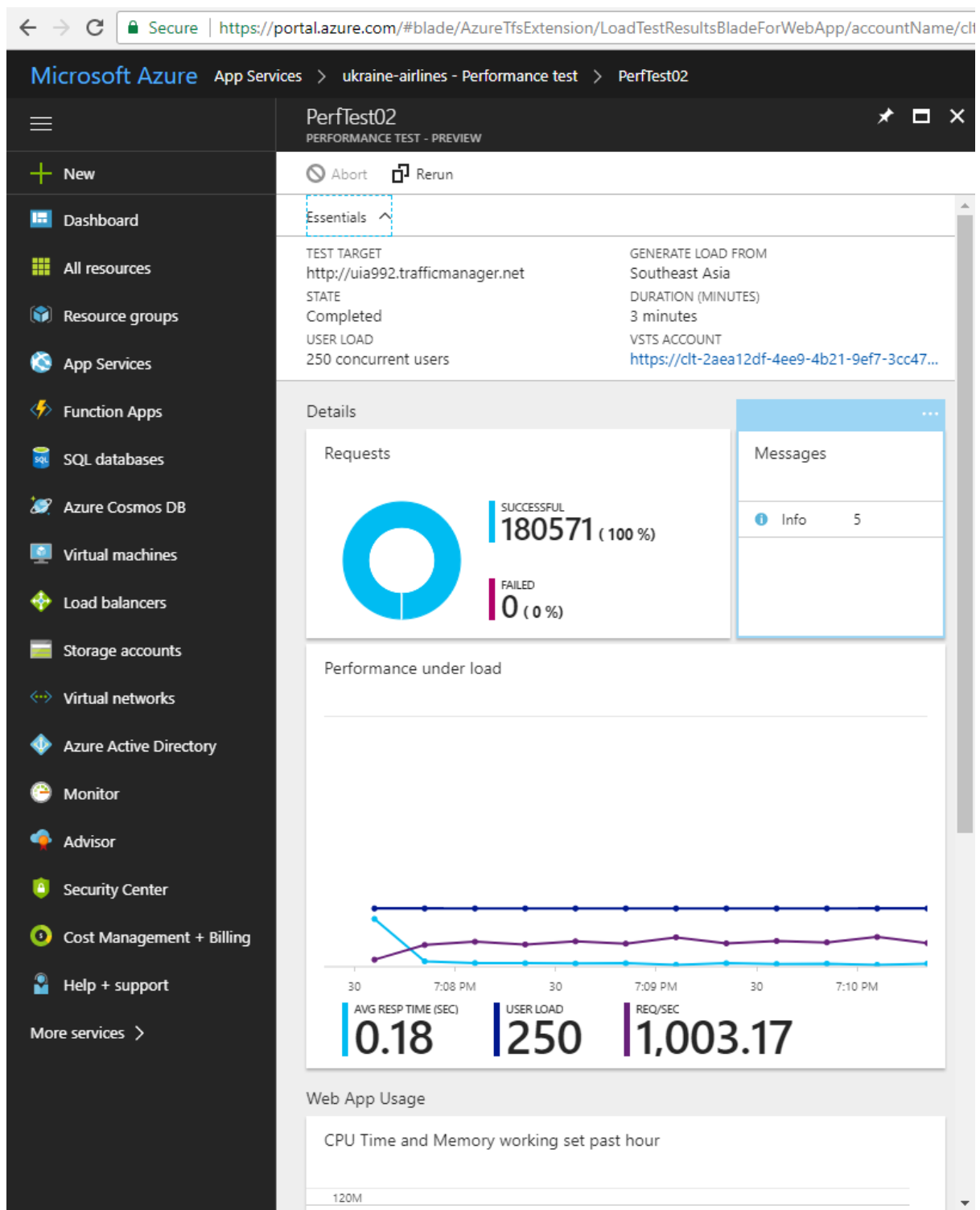


Diagram 5.1 (b): Performance Test Details

The test will take approximately 5 minutes to be completed. After the test is completed, it will show as the Diagram 5.1 (b). The section shows the number of successful and failed request during the performance test.

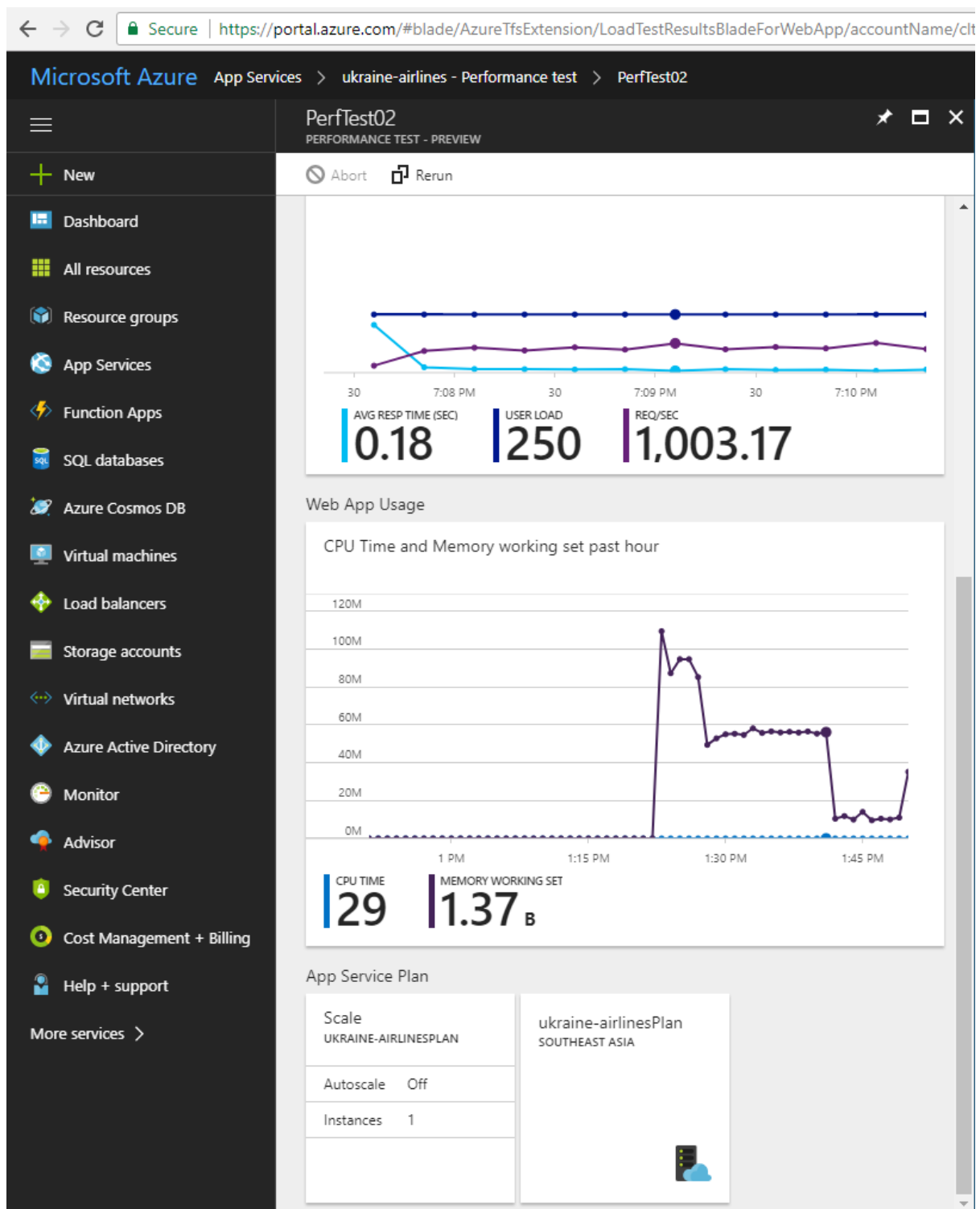


Diagram 5.1 (c): Performance under Load

Diagram 5.1 (c) shows the Performance under Load. It shows the average response time, user load and request per second at the point time indicated as the dots.

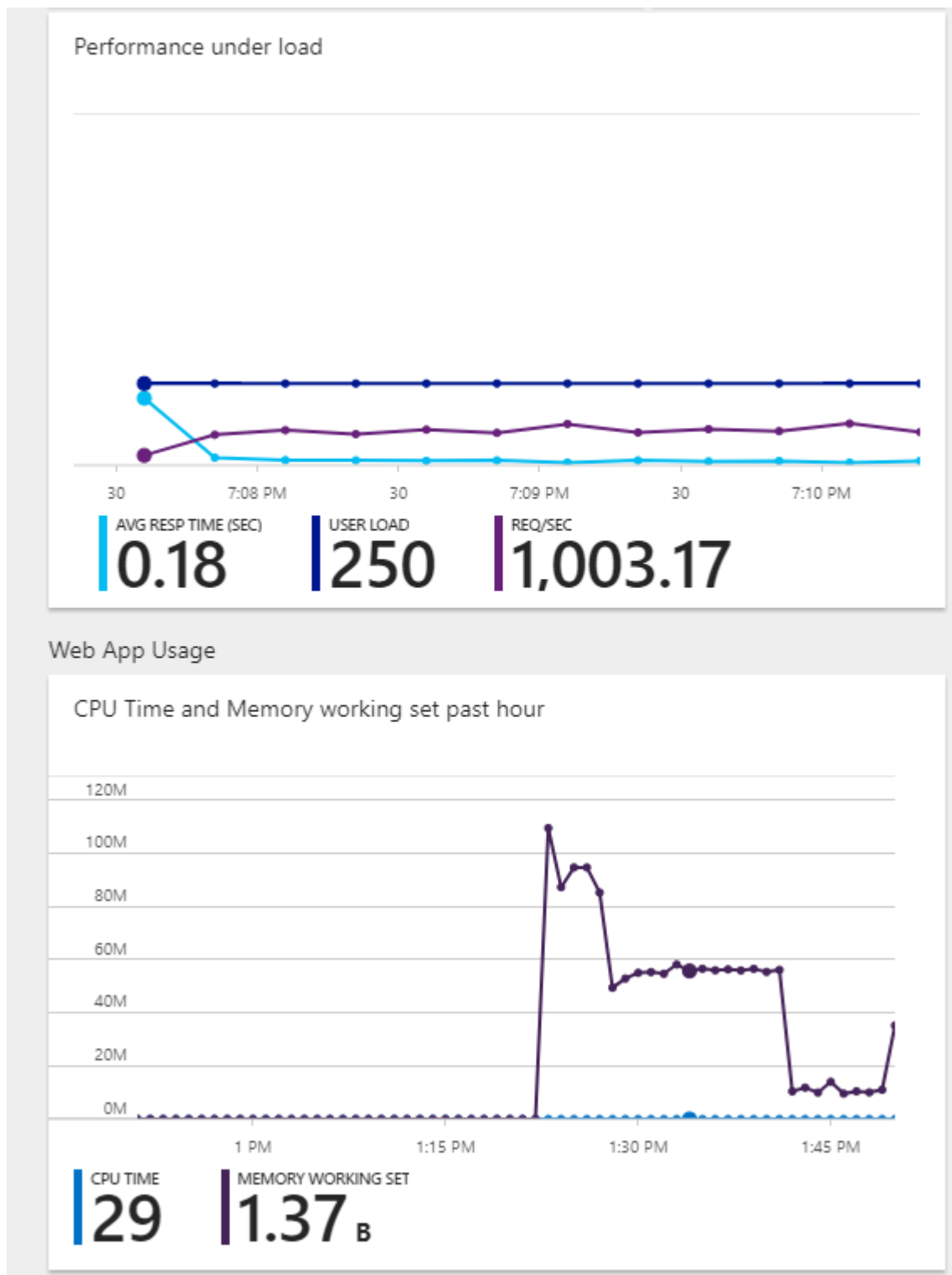


Diagram 5.1 (d): Web App Usage

Diagram 5.1 (d) shows the Web App Usage. It shows the CPU Time and Memory working set past hours.

5.2 Functional Test

This testing phase involve the functional testing which will allow the different features to be tested in terms of their functionality under different scenarios, and the actual result is compared to the expected result. The functional test cases will run after each set of features is developed with its associated functions necessary to complete the feature. This provides an incremental approach to the development and testing, such that the faulty functions are refined and fixed before continuing the development of the other features. For this project, the scope of scripts that are to be tested and those that will be excluded needs to be defined. Furthermore, the test cases will include mostly edge cases, in which the scenarios are bound to fail if the features were not validated properly. The features that will be tested under the functional test are:

- **Login**
- **Registration**
- **Modify Profile**
- **Booking A Flight**

The modules that are out-of-scope for the functional test are:

- **Searching for a flight** – all the functionalities are already part of the Booking a Flight feature, therefore it would be redundant
- **View Flight Details** – The Flight details are available when booking a flight, therefore it would be redundant
- **Book a Seat** – all the functionalities are already part of the Booking a Flight feature, therefore it would be redundant

<i>ID</i>	<i>Feature/Function Tested</i>	<i>Test Procedure/Scenario</i>	<i>Description</i>	<i>Expected Result</i>	<i>Priority</i>
FT1	Login	Insert Wrong username. Execute function.	This test involves checking the Login validation and how it reacts to wrong input that doesn't exist in the database.	A Message indicating wrong username is showed. Redirected to Login page.	High
FT2	Login	Insert Right username but wrong password. Execute function.	This test involves checking the Login validation and how it reacts to wrong input that doesn't exist in the database.	A Message indicating wrong password/username is showed. Redirected to Login page.	High
FT3	Login	Insert Right username but empty password Execute function.	This test involves checking the Login validation and how it reacts to wrong input that doesn't exist in the database.	A Message indicating that the password input is required. Redirected to Login page.	High

<i>ID</i>	<i>Feature/Function Tested</i>	<i>Test Procedure/Scenario</i>	<i>Description</i>	<i>Expected Result</i>	<i>Priority</i>
FT4	Registration	Fill the required information except of critical fields such as passport id. Execute function.	This test involves checking how the validation and verification functions will react to empty or null information when inserted to the database.	Notify the user about the missing field with red highlighted text beside it Refreshing the page without losing the current details.	High
FT5	Modify Profile	Modify the profile with new information after creating a new account Execute function.	This test involves checking how the validating and evaluating model feature functions when modifying the information within same session that is created.	User details have been updated successfully. Redirecting to User Page.	High
FT6	Booking A Flight	Book a flight with missing information such as Cabin Class information Execute function.	This test involves checking how the system will react to searching and then select a flight afterwards retrieving the required session details such as cabin class which will result in Null exception normally.	The user will be prompted to modify/login his account or with new account details to proceed to check out. Continue with booking and checking out the flight ticket.	High

6.0 Conclusion

In conclusion, a web application for Ukraine International Airlines (UIA) has developed to let the users manage the details efficiently. The web application has developed to satisfy the requirements defined by Ukraine International Airlines (UIA) to host the developed web application on Microsoft Azure as App Service has made to meet the needs of Dmitriy Prudnikov. Other than that, by using the cloud services provided by Microsoft Azure is able to accelerate the setup and deployment of the solution. For example, a new server will take approximately 8~16 weeks to deploy in an on-premise environment. However, it only takes few minutes to deploy in the cloud computing environment.

Furthermore, Azure Traffic Manager is implemented to control request from web clients. Performance traffic routing is also done by deploying endpoints in two or more locations across the globe. It will help to improve the responsiveness of the web application by routing traffic to the location that is closest to the client. Lastly, performance test is also done to ensure the web application is capable to handle certain user load and make sure it's up for scalability in the near the future without impacting the performance. In this assignment, the developer had gained valuable experience and knowledge of hosting web application on Microsoft Azure as App Service.

References

Dwivedi, K., 2017. *Overview of Traffic Manager*. [Online]

Available at: <https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

Labeler, C., 2017. *What is the Azure SQL Database service?*. [Online]

Available at: <https://docs.microsoft.com/en-us/azure/sql-database/sql-database-technical-overview>

Lin, C., 2017. *Web Apps overview*. [Online]

Available at: <https://docs.microsoft.com/en-us/azure/app-service-web/app-service-web-overview>

Appendices

URL: <http://ukraine-airlines.azurewebsites.net>

Login ID: ahmed_nabil992@hotmail.com

Password: 123

GitHub: [https://github.com/AhmedAlEryani/DDAC TP035572](https://github.com/AhmedAlEryani/DDAC_TP035572)

Microsoft Stream: <https://web.microsoftstream.com/video/fffc7642-5091-4b15-a15b-035c9f82a511>