



# Deployment Portfolio Task 3

**SWE40006 - SOFTWARE DEVELOPMENT AND EVOLUTION**

SUMMER - 2024 SUBMITTED ON 16<sup>th</sup> OF JUNE

# Student and Lecturer Details

Name	ID	Lecturer	Class
Vi Luan Dang	103802759	Dr. Thomas Hang Nsam@swin.edu.au	Monday 13:00 PM

# Self-Assessment Details

Declaration ò task level attempted (P/C/D/HD)

	Pass	Credit	Distinction	High Distinction
Self-Assessment				✓

	Included & attempted
Task 3.1: Pass	✓
Task 3.2: Credit	✓
Task 3.3: High Distinction	✓

# Table of Content

TASK 3.1P: ..... 2

TASK 3.2C ..... 5

TASK 3.3HD ..... 8

RESOURCES ..... 10

# Assignment Report

## Task 3.1P:

### 3.1A. Create Azure account

As I already have an Azure account from previous courses and project, there is nothing to do for this task.

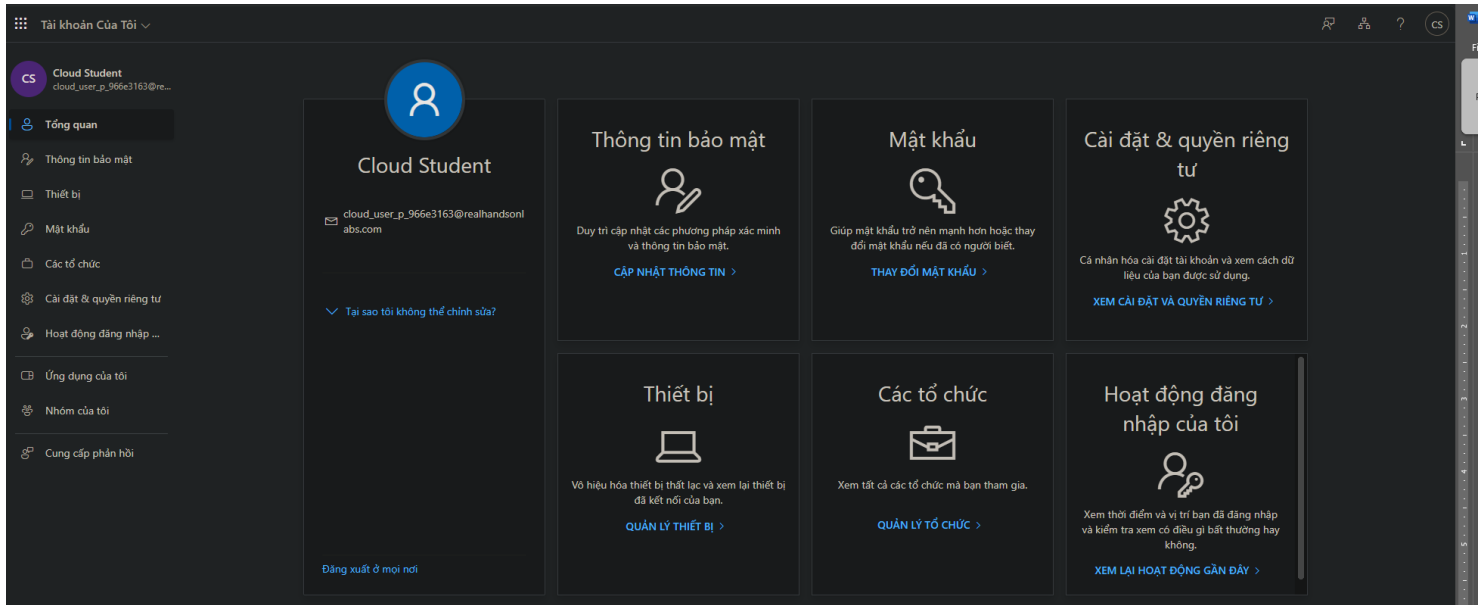


Figure 1: Azure account.

### 3.1B. Install visual Studio

Similar to the above task, I have already had Visual Studio installed when doing the first portfolio assignment, therefore, there is nothing to do.

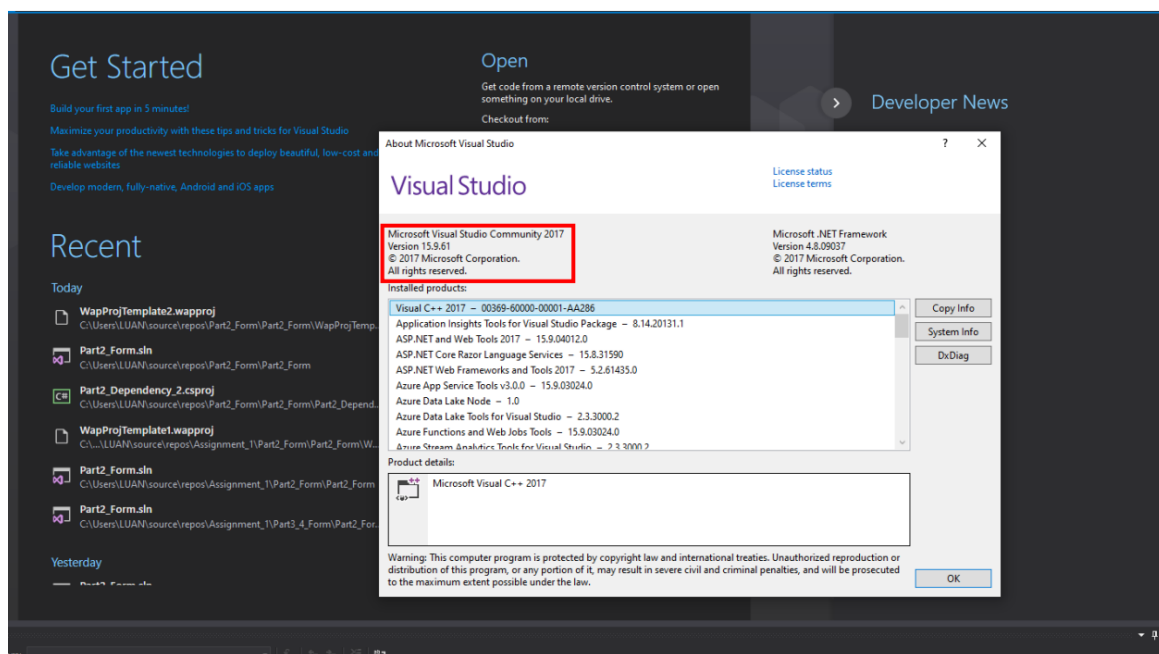
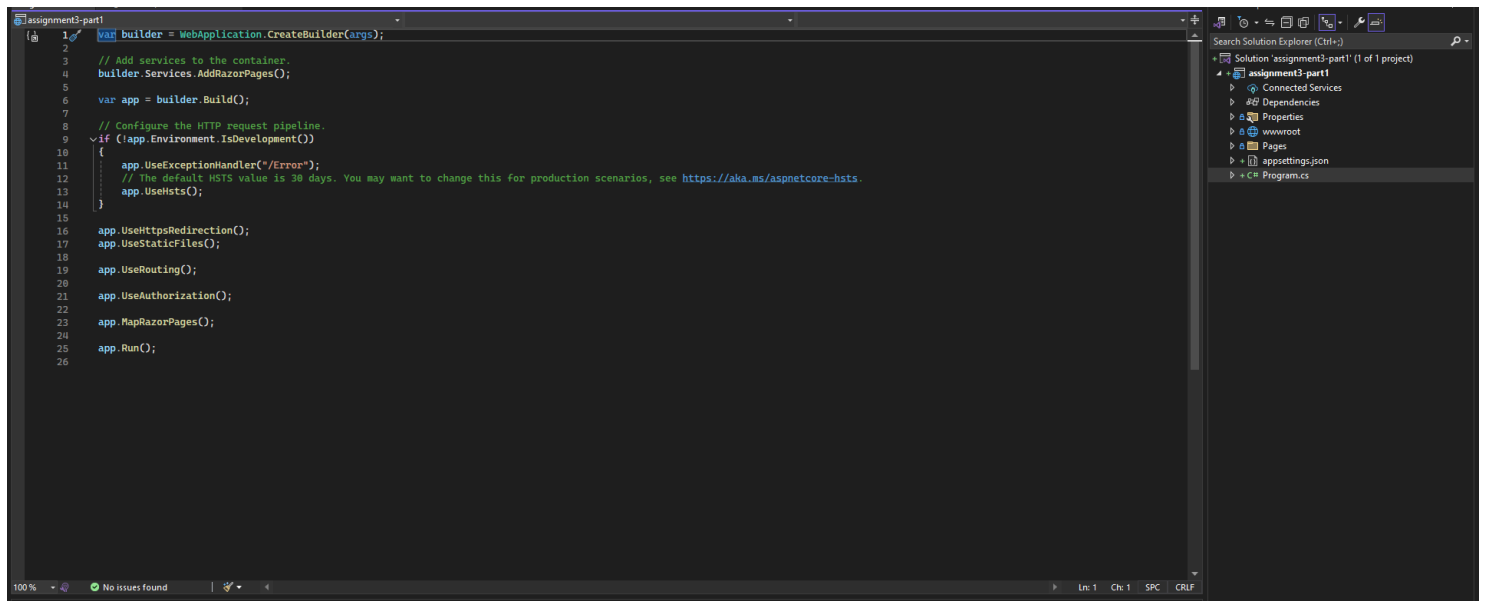


Figure 2: Visual Studio 2017 installed.

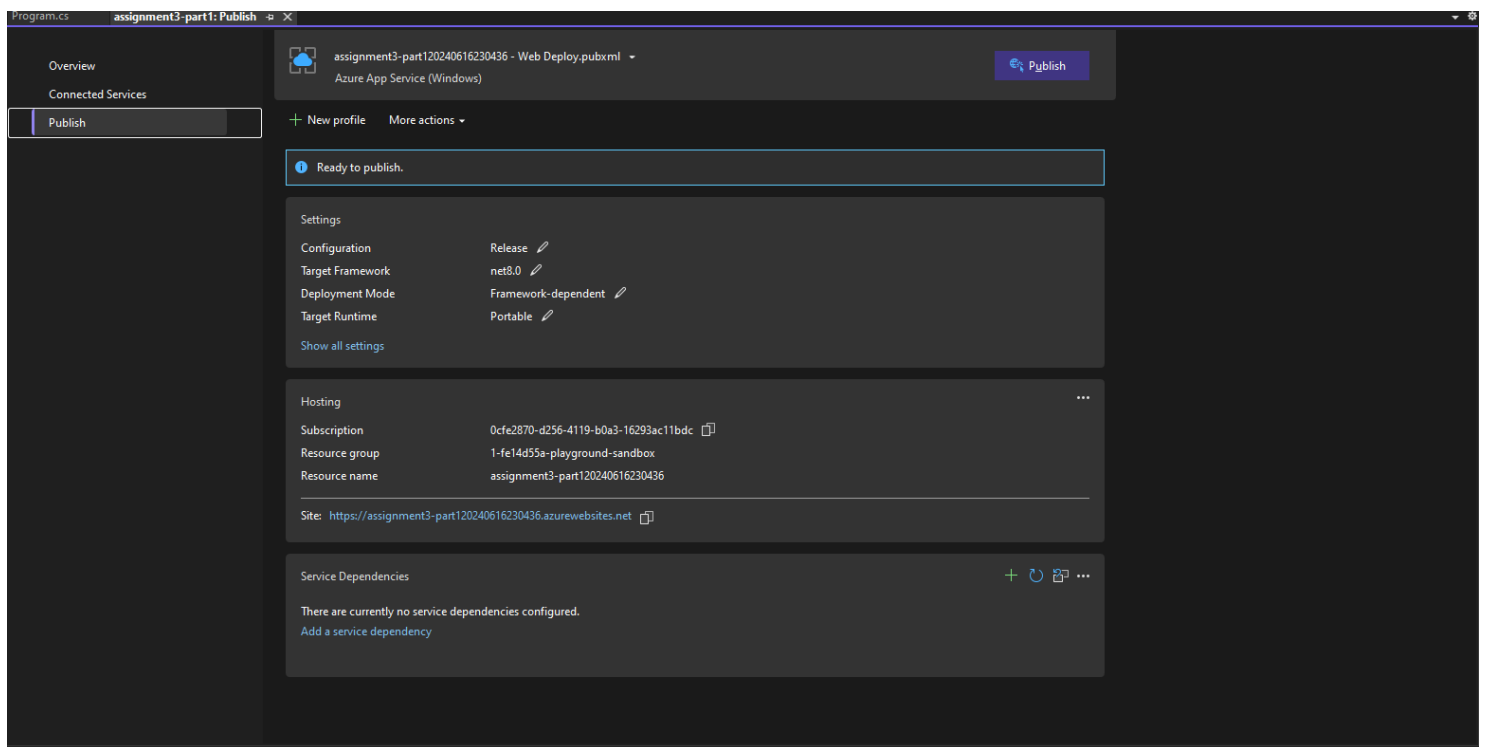
### 3.1C. Deploy any existing app to cloud via Azure

For this task, I will deploy a simple C# web application onto Azure Cloud Environment.



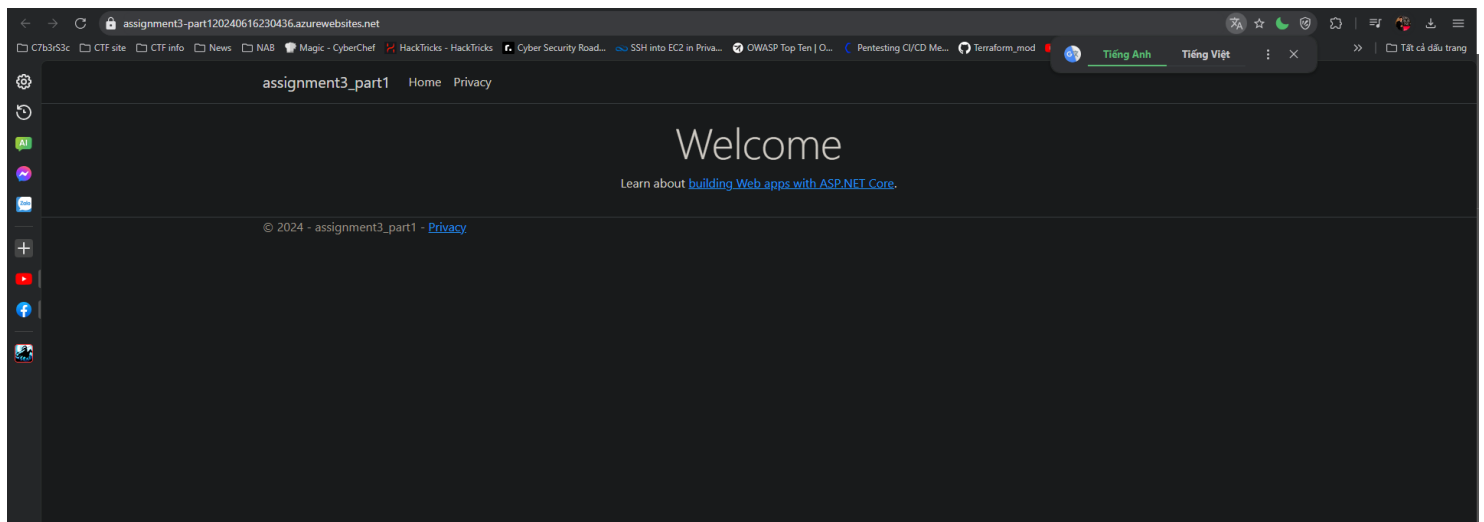
*Figure 3: C# Web application.*

After that I will provision a webserver for our application:



*Figure 4: Web server successfully provisioned.*

Finally, I will publish my application onto Azure Cloud.



*Figure 5: Application successfully published*

This will also conclude the first task of this assignment.

## Task 3.2C:

### 3.2A. Deploy an existing app to cloud via Azure

For this task I will create a C# counter application and deploy it onto Azure cloud.

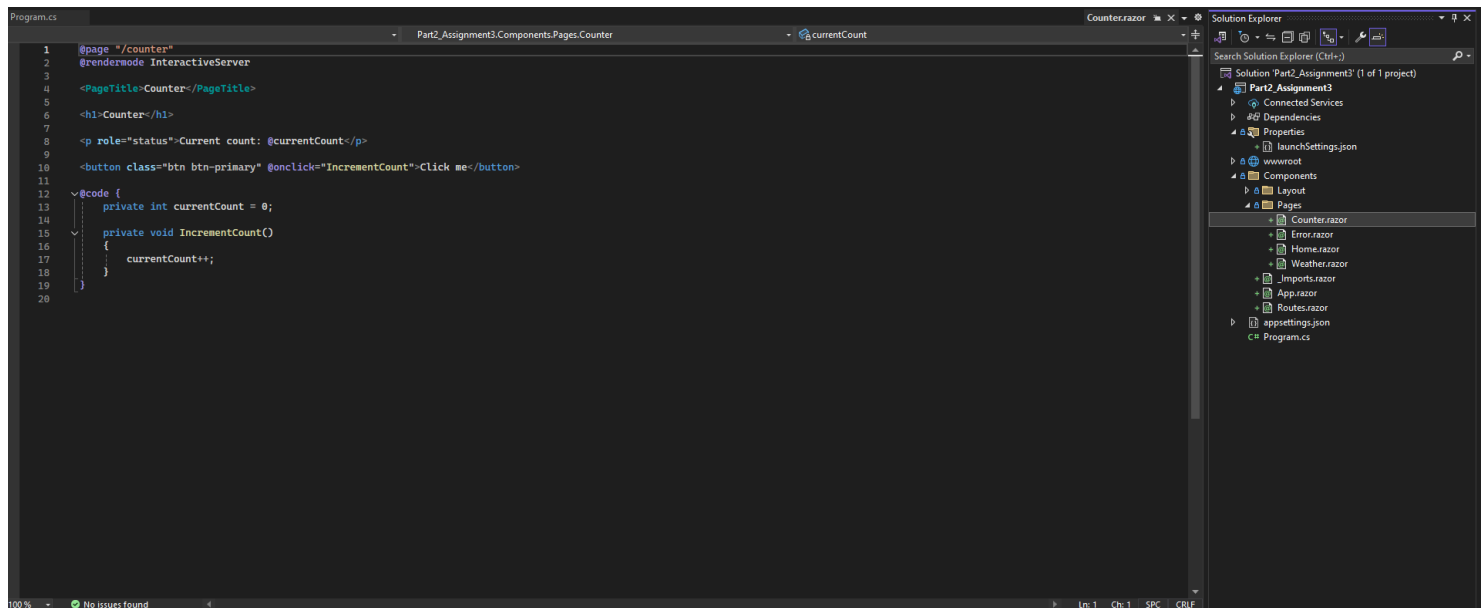


Figure 6: C# Counter application.

We can check the application on the local host of our computer.

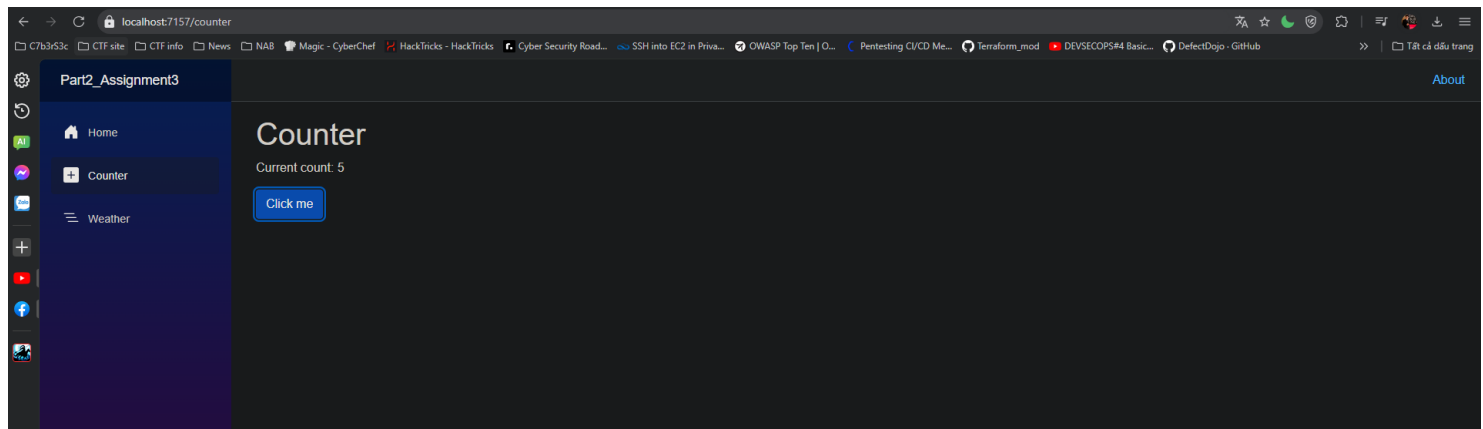


Figure 7: C# counter working on localhost

Next, we will publish this application on to the webserver of Azure Cloud.

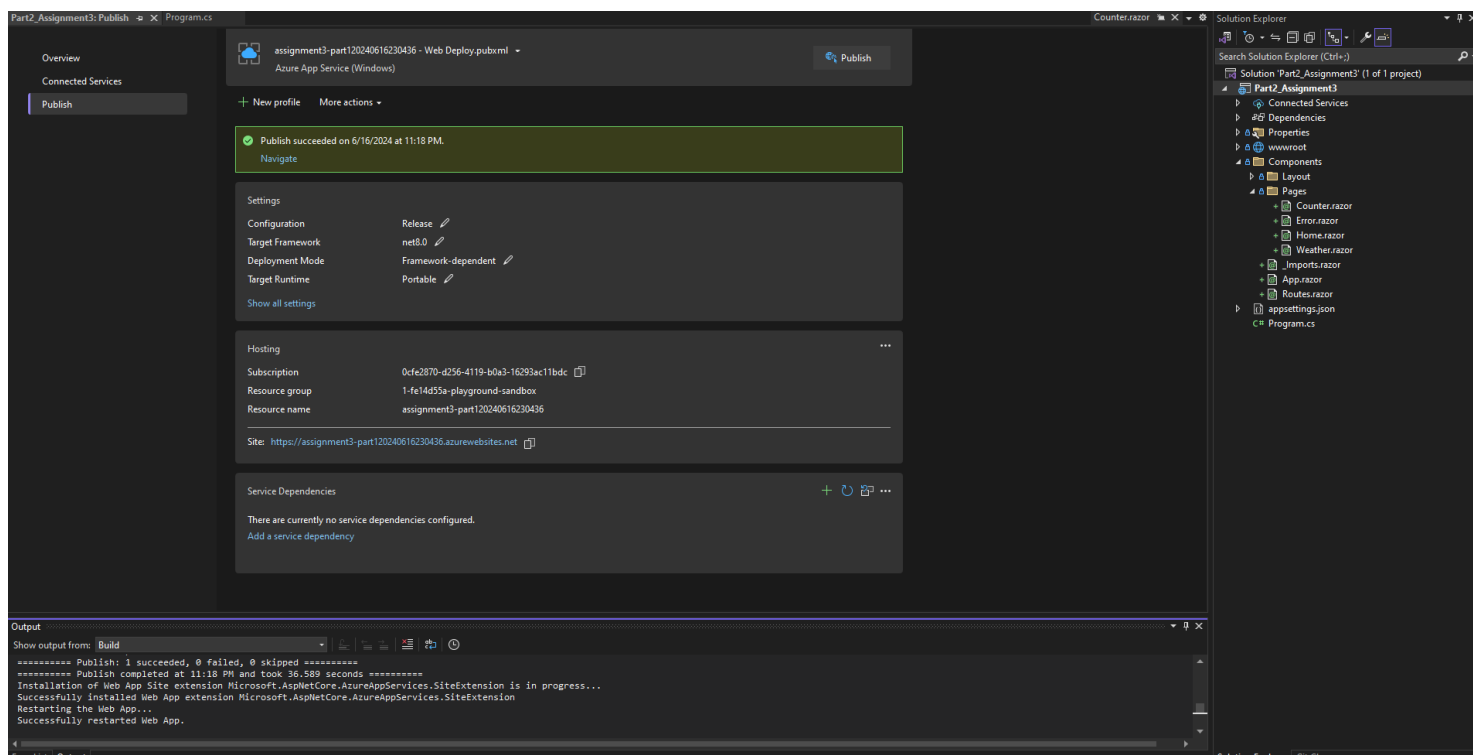


Figure 8: C# counter application successfully published.

We can then browse onto our web application to see if it is working correctly.

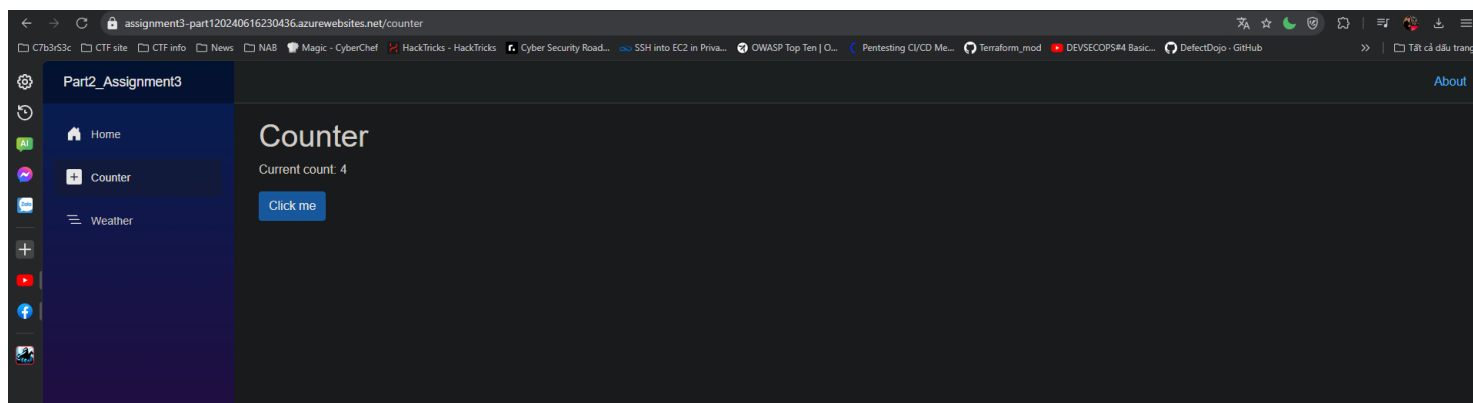
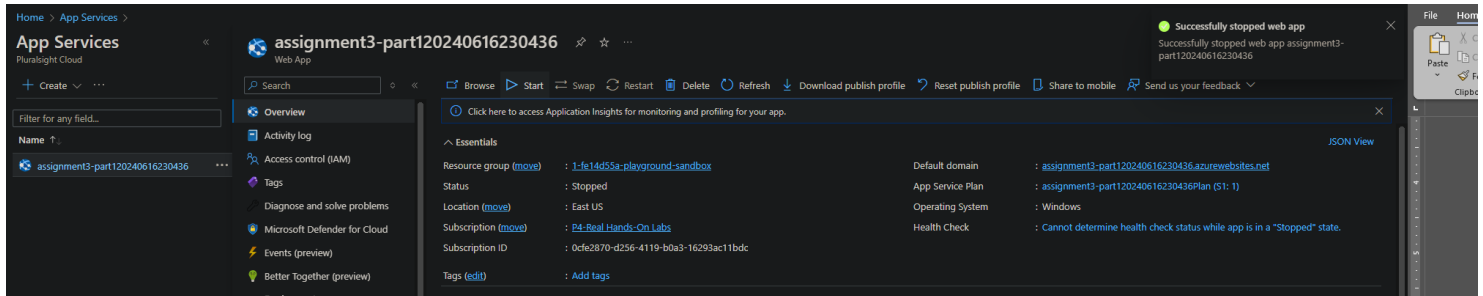


Figure 9: Web application works correctly on Azure Cloud Server.

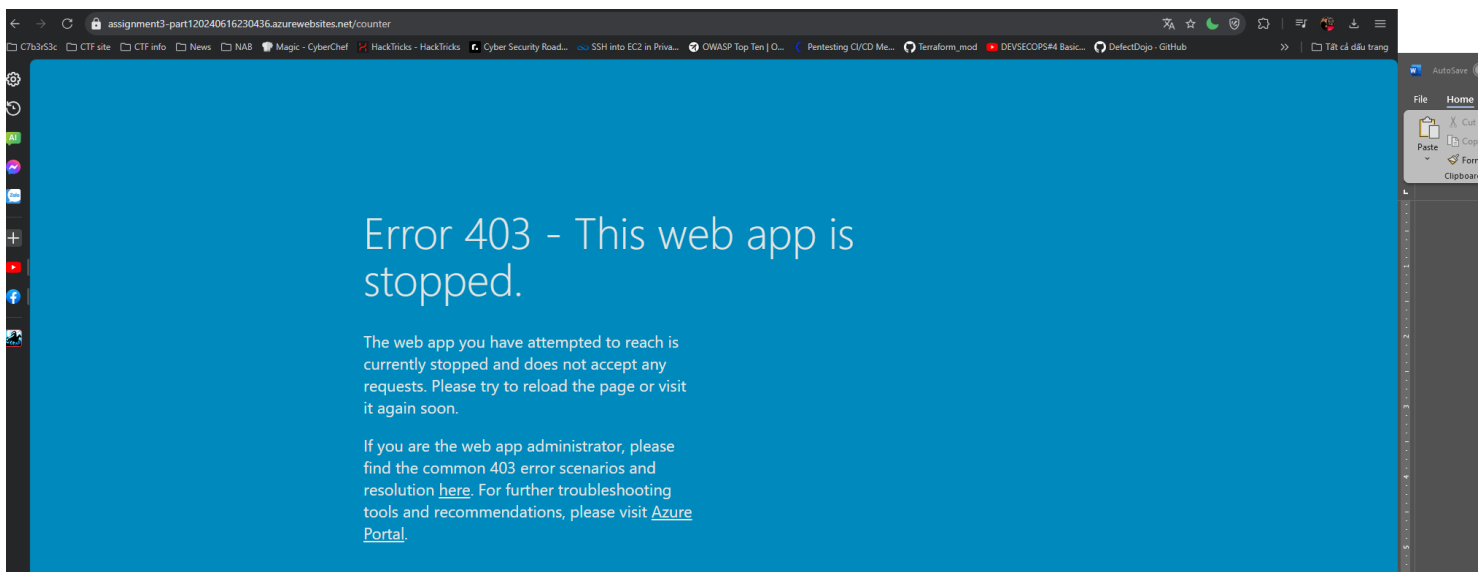
## 3.2B. Deactivate web application

We can deactivate our application on the Azure GUI as shown in the figure below:



*Figure 10: Azure application deactivated*

We can also check this by browsing our applications.



*Figure 11: Web application deactivated*

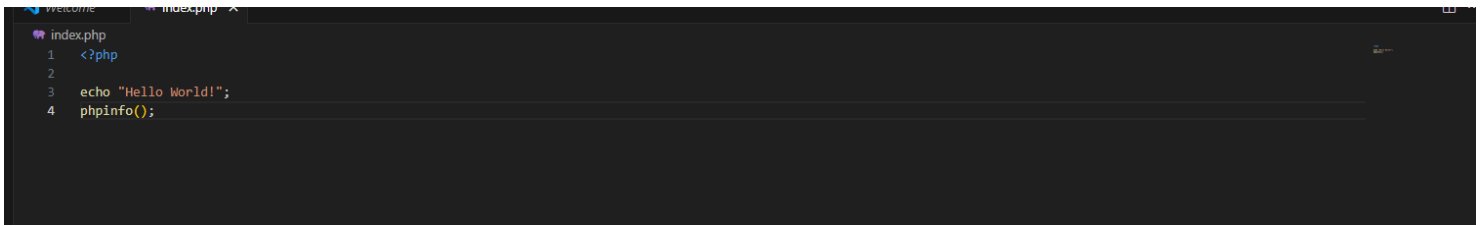
As our application has been stopped, we can confirm that our tasks have been completed. This will also conclude the second task of this assignment.



## Task 3.3HD:

### 3.3A. Deploy an PHP web application to Azure Cloud Environment

For this task I will use Visual Studio Code to write a simple PHP application and deploy it onto Azure, there is no need to transfer files between two different IDEs. The application will only be a simple “Hello-World” PHP and PHP information application.



```
index.php
1  <?php
2
3  echo "Hello World!";
4  phpinfo();
```

Figure 12: Simple PHP web application

I will then create a new web server and deploy my code onto it.

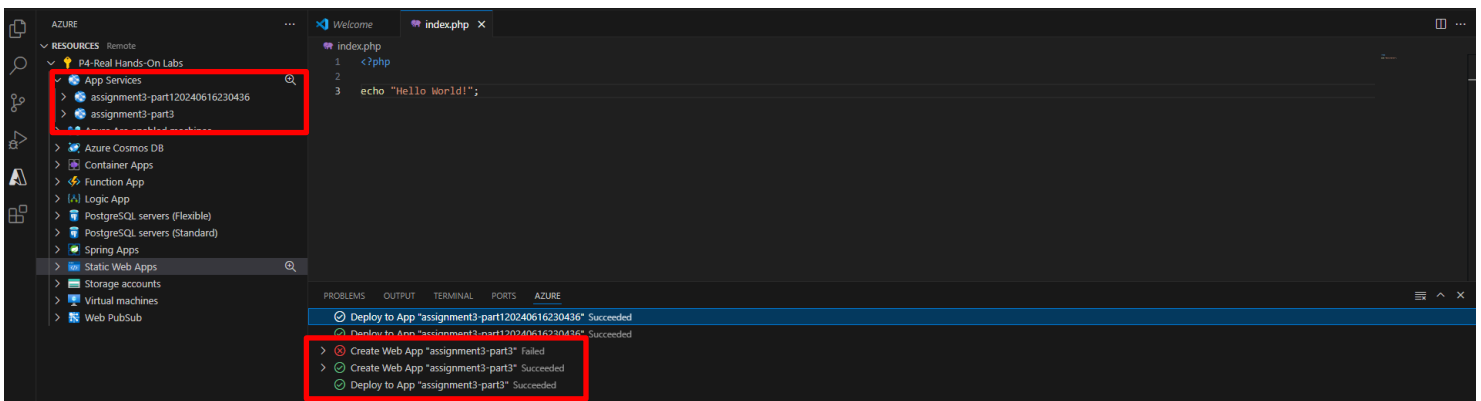


Figure 13: Web server successfully created and deployed

We can then browse our website to see if it works as expected.

C763f3c

CTF site

CTF info

News

NAB

Magic - CyberChef

HackTricks - HackTricks

Cyber Security Roadmap

SSH into EC2 in Private

OWASP Top Ten | ...

Pentesting C/UD Me...

Terraform, mod

Tiếng Anh

Tiếng Việt

Hello World!

PHP Version 8.3.4

System	Linux assignment3-part3_d7df829d 5.15.145 2-1 cm2 #1 SMP Wed Jan 17 15:39:07 UTC 2024 x86_64
Build Date	Apr 30 2024 14:38:16
Build System	Linux build@sandbox 5.15.0-1060-azure #69-20.04.1-Ubuntu SMP Tue Mar 19 22:14:45 UTC 2024 x86_64 GNU/Linux
Configure Command	"/configure" "--build=x86_64-linux-gnu" "--with-config-file-path=/usr/local/etc/php/" "--with-config-file-scan-dir=/usr/local/etc/php/conf.d/" --enable-option-checking=fatal" "--with-mhash!" --enable-fpm --enable-mbstring"--enable-mysqlnd" "--with-password-argon2" "--with-sodium-shared" "--with-pdo-sqlite=libsqlite3" "--with-curl" "--with-ibm-db2" "--with-openssl" "--with-zlib" "--with-pear" --with-libdir=libx86_64-linux-gnu" --enable-fpm --with-fpm-user=www-data" "--with-fpm-group=www-data" "--disable-cgi" "ac_cv_func_mmap=no" "build_alias=x86_64-linux-gnu"
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/usr/local/etc/php
Loaded Configuration File	(none)
Scan this dir for additional .ini files	/usr/local/etc/php/conf.d
Additional .ini files parsed	/usr/local/etc/php/conf.d/20-opcache.ini, /usr/local/etc/php/conf.d/20-pdo_sqlsrv.ini, /usr/local/etc/php/conf.d/docker-php-ext-bcmath.ini, /usr/local/etc/php/conf.d/docker-php-ext-calendar.ini, /usr/local/etc/php/conf.d/docker-php-ext-exif.ini, /usr/local/etc/php/conf.d/docker-php-ext-gd.ini, /usr/local/etc/php/conf.d/docker-php-ext-gettext.ini, /usr/local/etc/php/conf.d/docker-php-ext-gmp.ini, /usr/local/etc/php/conf.d/docker-php-ext-imap.ini, /usr/local/etc/php/conf.d/docker-php-ext-intl.ini, /usr/local/etc/php/conf.d/docker-php-ext-ldap.ini, /usr/local/etc/php/conf.d/docker-php-ext-mongodb.ini, /usr/local/etc/php/conf.d/docker-php-ext-mysqli.ini, /usr/local/etc/php/conf.d/docker-php-ext-odbc.ini, /usr/local/etc/php/conf.d/docker-php-ext-openssl.ini, /usr/local/etc/php/conf.d/docker-php-ext-pcntl.ini, /usr/local/etc/php/conf.d/docker-php-ext-pdo_mysql.ini, /usr/local/etc/php/conf.d/docker-php-ext-pdo_odbc.ini, /usr/local/etc/php/conf.d/docker-php-ext-pdo_pgsql.ini, /usr/local/etc/php/conf.d/docker-php-ext-readline.ini, /usr/local/etc/php/conf.d/docker-php-ext-shmop.ini, /usr/local/etc/php/conf.d/docker-php-ext-soap.ini, /usr/local/etc/php/conf.d/docker-php-ext-sockets.ini, /usr/local/etc/php/conf.d/docker-php-ext-sodium.ini, /usr/local/etc/php/conf.d/docker-php-ext-sysvmsg.ini, /usr/local/etc/php/conf.d/docker-php-ext-sysvsem.ini, /usr/local/etc/php/conf.d/docker-php-ext-synchoniz..., /usr/local/etc/php/conf.d/docker-php-ext-tidy.ini, /usr/local/etc/php/conf.d/docker-php-ext-xsl.ini, /usr/local/etc/php/conf.d/docker-php-ext-zip.ini, /usr/local/etc/php/conf.d/opcache-recommended.ini, /usr/local/etc/php/conf.d/php.ini
PHP API	20230831
PHP Extension	20230831
Zend Extension	420230831
Zend Extension Build	API420230831.NTS
PHP Extension Build	API20230831.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
Zend Max Execution Times	disabled

*Figure 14: PHP application work successfully*

This will also conclude the final task of assignment 3.

## Resources:

The code and configuration files of this assignment can be found here via [this link](#)