

AZ-104 Case Study 0

Bonus points are not essential, but are meant to add another challenge if the main solutions are completed rapidly.

Company Overview

Contoso Coffee is a small coffee house that is opening soon in London and New York, they are looking for a cloud hosted solution for their website and data storage, costs must be kept to a minimum

User accounts

Contoso coffee will require 3 Administrators for the day to day control of the solution.

Bob

Bob will require full admin access as it will be his responsibility to manage and track billing for the Contoso Coffee website and is the owner of Contoso Coffee

Dave

Dave requires admin access to the resources hosted on Azure for the Contoso Coffee website but not overall administration of the Contoso Coffee subscription

Mark

Mark will require read-only access to the Contoso Coffee resources.

Website hosting

The Contoso Coffee website will need to be hosted with a PaaS solution that minimizes costs, dynamic scaling of the website is a consideration but not essential. The website should be able to be updated by developers using standard code hosting solutions.

Difficulty Level 1: Pull a GitHub hosted site via the deployment center with webapps

Difficulty Level 2: Create a Docker Container based solution and deploy to a webapp

Difficulty Level 3: Host the Docker Container solution via Azure Container Registries and pull into an Azure ACI solution.

Difficulty Level 3 example <https://docs.microsoft.com/en-gb/azure/container-instances/container-instances-tutorial-prepare-app> (<https://docs.microsoft.com/en-gb/azure/container-instances/container-instances-tutorial-prepare-app>)

Load Balancing and Geo-redundent access

Contoso Coffee will operate in London and New York, because of this a solution will need to be configured that allows the US users to access a more local server than London and vice versa. Load balancing can be achieved through DNS redirection or a geo-load balanced solution

Contoso Coffee example website

The Contoso Coffee website example is provided at <https://github.com/kramit/CoffeeShopTemplate> (<https://github.com/kramit/CoffeeShopTemplate>) this website is a simple static HTML CSS site that can be forked and edited as required. Bonus points for creating a fancy website

Contoso Coffee data storage

Contoso coffee is planning to store a large amount of image information that will be shared with staff members and clients. These images will be scans of 35 and 120 film for hipster photography that will be displayed around the coffee shop. The storage solution should minimize costs by automatically archiving data onto slower storage when required. A SAS key solution should be created for a future image gallery solution to display images on a TV in the Coffee Shop running a small Raspberry Pi solution locally hosting a web page to display the images. Bonus points for creating an image display web app.

Contoso Coffee ARM template

The deployment of the Contoso Coffee website should be re-deployable through an ARM template (or Bicep/Terraform if you wish as they are common solutions but not part of the course)

GitHub

A GitHub repo should be created for the Contoso Coffee website and ARM template solutions allowing other admins and consultants to fork the solutions as needed

Diagrams

Visual diagrams for the final solution should be created with the standard Azure Icon set found at <https://docs.microsoft.com/en-us/azure/architecture/icons/> (<https://docs.microsoft.com/en-us/azure/architecture/icons/>).

AZ-104 Case Study 1

Bonus points are not essential, but are meant to add another challenge if the main solutions are completed rapidly.

Company Overview

Acme Inc is a small manufacturing company specialising in tartan paint and left handed screwdrivers. The company has been slow to move to modern cloud based ways of working and has proprietary software that cannot run in a PaaS or SaaS solution without a complete re-write of the software.

They are looking to move some of the hosting to a cloud based solution and retiring some of the on-site hardware but do not want a major change in the way that the software is administered.

Hosting requirements

Acme will be manually migrating 2 software applications into the cloud hosted on 2 separate virtual machines.

1x Windows Server based machine: The Windows based machine will need to have installed IIS ready to host an older public facing website to show the latest tartan paint patterns. For bonus points change the IIS default page to a more interesting placeholder page.

1x Linux (Ubuntu) based machine: The Linux machine will be hosting a publicly accessible guest FTP server installed and configured for clients to download the latest CAD designs for left handed screwdrivers

data for these requirements will be provided by Acme internally and configured post-deployment by Acme employees

Security requirements

The FTP and HTTP/S sites will be required to be protected by a Firewall solution restricting external access only to ports 80/21/443.

Remote access cannot be performed over the public internet, only via an additional jump-box or bastion host access.

(For bonus points attempt a remote access solution using a 3rd party tool hosted in Azure e.g.

<https://guacamole.apache.org/>(<https://guacamole.apache.org/>),)

Logs (bonus points)

For bonus points set up logging for the Firewall to a Log Analytics Workspace and quite a Kusto query to create a dashboard widget for network traffic

<https://docs.microsoft.com/en-us/azure/application-gateway/log-analytics> (<https://docs.microsoft.com/en-us/azure/application-gateway/log-analytics>).

Backup

Acme Inc requires full backups of both VM images using an Azure based solution so administrators can restore the full machines in the event of a failure

Storage

Future long term storage should be set up using Azure storage services and linked into the Windows VM by the use of a drive map as the on site admins are more familiar with a drive map than another more complex solution

ARM Templates

ARM Templates are not required for this static non-scalable solution but the deployment method should be as simple as possible for admins that are not very experienced with cloud solutions or any scripting languages.

Diagrams

Visual diagrams for the final solution should be created with the standard Azure Icon set found at <https://docs.microsoft.com/en-us/azure/architecture/icons/> (<https://docs.microsoft.com/en-us/azure/architecture/icons/>).

AZ-104 Case Study 2

Bonus points are not essential, but are meant to add another challenge if the main solutions are completed rapidly.

Company Overview

Pineapple Pizza is a small pizza chain that wants to set up a new website at minimal cost to allow an online blogging platform for sharing new and interesting ways to integrate pineapples into pizza based delivery solutions for the 21st century.

User accounts

Pineapple Pizza will require 3 Administrators for the day to day control of the solution.

Bob

Bob will require full admin access as it will be his responsibility to manage and track billing for the Pineapple Pizza website and is the owner of Pineapple Pizza

Dave

Dave requires admin access to the resources hosted on Azure for the Pineapple Pizza website but not overall administration of the Pineapple Pizza subscription

Mark

Mark will require read-only access to the Pineapple Pizza resources.

Website hosting

Pineapple Pizza has decided that they want to use an off the shelf static-site solution to generate their platform. They wish to use either Gatsby, Hugo, VuePress or Jekyll with a preference to a Jekyll based solution.

Deployment must be configured to use a github based repo with github actions for automated deployment of the static site.

The free tier for Azure static web apps should suffice for the solution

<https://docs.microsoft.com/en-us/azure/static-web-apps/> (<https://docs.microsoft.com/en-us/azure/static-web-apps/>).

Load Balancing and Geo-redundent access

Pineapple Pizza will operate in London, Oslo, Singapore and Tokyo, because of this a solution will need to be configured that allows users around the world to access local region cached solutions by using an Azure Front Door based CDN solution.

<https://docs.microsoft.com/en-us/azure/static-web-apps/front-door-manual> (<https://docs.microsoft.com/en-us/azure/static-web-apps/front-door-manual>).

Pineapple Pizza example website

Pineapple Pizza, like any good client has no idea what they actually want from a website, but expects a templated site with the following pages that they can edit in the future.

- Home
- About Us
- Contact Us
- Updates (this will be for blog posts)

Create some example posts in the blog and example text in each of the pages

Pineapple Pizza Alerting

Pineapple Pizza also wishes to be alerted when the hits to the site via the Front Door solution hit 100 requests worldwide, the alert generated from Front Door should send an email to the owner of Pineapple Pizza (Bob, but the email address only needs to be a placeholder), you can force 100 requests for testing with the following powershell code

```
1..100 | % {invoke-webrequest -Uri PineapplePizzaUri -DisableKeepAlive}
```

Pineapple Pizza ARM template

The deployment of the Pineapple Pizza website should be re-deployable through an ARM template (or Bicep/Terraform if you wish as they are common solutions but not part of the course)

<https://docs.microsoft.com/en-us/azure/static-web-apps/publish-azure-resource-manager?tabs=azure-cli>
(<https://docs.microsoft.com/en-us/azure/static-web-apps/publish-azure-resource-manager?tabs=azure-cli>).

GitHub

A GitHub repo should be created for the Pineapple Pizza website that includes the ARM template solutions along with the static site code allowing other admins and consultants to fork the solutions as needed

Diagrams

Visual diagrams for the final solution should be created with the standard Azure Icon set found at <https://docs.microsoft.com/en-us/azure/architecture/icons/> (<https://docs.microsoft.com/en-us/azure/architecture/icons/>).

Case Study Readme

There are 3 case studies here, you only have choose 2 out of the 3 to do. It does not matter which 2.

The point of the case studies is to show that you can take a clients needs and design and azure solution that would meet the needs of the client, you will then need to present this solution to some members of Microsoft and the rest of the MSU group.

It will not be too stressful and I have attempted to keep the case studies to solution that we will cover in class.

I really advise that you have Github repos created for your projects, this will make things much simpler when showing them off.

Take a look at an example from the MSU2021 completed by a student on that course that uses all the azure icon sets and a well formatted Github repo for his project <https://github.com/kramit/contosocoffee>
(<https://github.com/kramit/contosocoffee>).

You can start working on these Case Studies whenever you feel like it, there are dedicated days to work on them too so we will have plenty of time, but you really do need to have them completed and squared away by the end of the course.

When giving the presentations, the Microsoft guys love it when you use the azure icon sets in your powerpoint presentations <https://docs.microsoft.com/en-us/azure/architecture/icons/> (<https://docs.microsoft.com/en-us/azure/architecture/icons/>).

us/azure/architecture/icons/), they love a good Visio type diagram, try using something like this tool <https://online.visual-paradigm.com/diagrams/features/azure-architecture-diagram-tool/> (<https://online.visual-paradigm.com/diagrams/features/azure-architecture-diagram-tool/>).