

REPORT

Of

Six Week Training

On

Cloud Computing

Submitted By:

Name: Aditya Garg

Roll No.: 11901145

Class: B.Tech. CSE (3CE 6)

Semester: 5th Semester



**Department of Computer Science & Engineering
Punjabi University, Patiala**

PREFACE

In the field of Computer Science & Engineering, the term project refers to a “Computerized work” that is completely performed using a device (computer, mobile, etc.).

Practical Training/Internship for a period of 6 weeks is a part of B.Tech. Computer Science & Engineering degree after 3rd year, according to the syllabus worked out at Punjabi University, Patiala to which my department is affiliated. I took my 6 week course from “**Udacity**” an online learning platform.

This report is a summary of What I learnt and performed there at the time of my Course. It basically includes my project as well as the things I learnt during the training. This also contains my certification in AZ900 which is an azure Fundamental examination with my score card.

DECLARATION

I, Aditya Garg, hereby affirm that the project report titled “Article CMS Deployment”, submitted to Punjabi University, Patiala in partial fulfilment of the requirements for the award of degree of B.Tech. (Computer Science and Engineering) is an authentic record of my own work and that to the best of my knowledge and belief, it contains no written material which has been accepted for the qualification of any other degree or diploma of a university or other institution of higher learning except where due acknowledgement is made.

Signature of student: _____

Name of student: Aditya Garg

Roll no: 11901145

Regd. No.: 7141-19-245

TABLE OF CONTENTS

Sr. No.	Description	Page No.
1	Preface	2
2	Certificate of completion	4
3	AZ-900 Score Card	5
4	AZ-900 Certification	6
5	Company's Profile	7
6	Introduction to Cloud Computing	8
7	What is Cloud Computing	10
8	Benefits Of Cloud Computing	11
9	Models of Cloud Computing	18
10	Types of Cloud Services	23
11	Microsoft Azure	25
12	Azure Portal	26
13	Azure Subscription and Resource Group	27
14	Azure Compute Services	29
15	Azure Storage	37
16	Security and Monitoring	42
17	Introduction to the project	48
18	Parts of the Project	49
19	Screenshots of the Project	50
20	References	59

INTERNSHIP CERTIFICATE



NANO DEGREE PROGRAM

Cloud Developer using Microsoft Azure

CERTIFICATE OF COMPLETION

Awarded to

Aditya Garg

November 03, 2021

Udacity has confirmed the participation of this individual in this program. Confirm program completion at confirm.udacity.com/RKSTH6I

A handwritten signature in black ink that reads "Sebastian Thrun".

SEBASTIAN THRUN
Founder, Udacity



AZ-900 Score Card



EXAM SCORE REPORT

CANDIDATE
Aditya Garg
I dan street Raja cinema road Fazilka Punjab 152123
adityagarg.ag2000@gmail.com

EXAM
AZ-900: Microsoft Azure Fundamentals

Exam reference #: 39480546
Date: June 30, 2021
ID: adityagarg00

RESULTS	100	200	300	400	500	600	700	800	900	1000
Required Score							700			
Your Score							800			

SECTION ANALYSIS	
Describe Cloud Concepts (20-25%)	83%
Describe Core Azure Services (15-20%)	71%
Describe Core Solutions and Management Tools on Azure (10-15%)	91%
Describe General Security and Network Security Features (10-15%)	64%
Describe Identity, Governance, Privacy, and Compliance Features (20-25%)	88%
Describe Azure Pricing and Support (10-15%)	73%

FINAL SCORE	
Required Score	700
Your Score	742

OUTCOME	
Pass	✓

© 2021 Certiport and the Certiport logo are registered trademarks of Certiport, a business of NCS Pearson, Inc. All other trademarks and registered trademarks are the property of their respective holders.

CERTIPORT
A PEARSON VUE BUSINESS

AZ-900 Certificate

Microsoft Certified Azure Fundamentals

Aditya Garg

has successfully completed the requirements of
Azure Fundamentals

Date Issued: June 30, 2021



A handwritten signature in black ink.

Satya Nadella
Chief Executive Officer



[verify.certiport.com: waUQV-H9dJ](https://verify.certiport.com/waUQV-H9dJ)

COMPANY PROFILE

About the Company:

Udacity began as an experiment in online learning, when Stanford instructors Sebastian Thrun and Peter Norvig elected to offer their "Introduction to Artificial Intelligence" course online to anyone, for free. Over 160,000 students in more than 190 countries enrolled. The potential to educate at a global scale was awe-inspiring, and Udacity was founded to pursue a mission to democratize education. It would take several years of intensive iteration and experimentation to clarify our focus on career advancement through mastery of in-demand skills, but today, Udacity proudly offers aspiring learners across the globe the opportunity to participate in—and contribute to—some of the most exciting and innovative fields in the world.

INTRODUCTION TO CLOUD COMPUTING

When you turn on a light, you simply want the light to work. You know you need electricity for that to happen, but in that moment, the details of how the electricity gets to the light bulb aren't important. You might not think about electricity being created in a power plant, traveling through a large network of high-voltage transmission lines to your town, going through a substation, and eventually making its way into your home.



The process of turning on a light is hidden behind the simple act of flipping a switch. At this point, electricity becomes a utility, which has many benefits. First, you only pay for what you need. When you buy a light bulb, you don't pay your electricity provider up front for how long you could possibly use it. Instead, you pay for the amount of electricity that you actually use. Second, you don't worry about how or when power plants upgrade to the latest technology. Finally, you don't have to manage scaling the electricity. For example, as people move to your town, you can rest assured that your light will stay on.

As a technology professional, it would be nice to have these same benefits when developing and deploying applications. Storing data, streaming video, or even hosting a website all require managing hardware and software. This management is an unnecessary obstacle when delivering your application to your users. Luckily there is a solution to this problem: cloud computing.

What is Cloud Computing?

Cloud computing is renting resources, like storage space or CPU cycles, on another company's computers. You only pay for what you use. The company providing these services is referred to as a cloud provider. Some example providers are **Microsoft, Amazon, IBM** and **Google**.

The cloud provider is responsible for the physical hardware required to execute your work, and for keeping it up-to-date. The computing services offered tend to vary by cloud provider. However, typically they include:

- **Compute power** - such as Linux servers or web applications
- **Storage** - such as files and databases
- **Networking** - such as secure connections between the cloud provider and your company
- **Analytics** - such as visualizing telemetry and performance data

Benefits of Cloud Computing

Cloud computing isn't an all-or-nothing service approach which means Companies can choose to use the cloud to store their data and execute logic as much, or as little as necessary for example a company need to deploy a service for 2 hours in on- Premis server they will have to buy all the hardware required to deploy the service but in cloud computing they can rent that same hardware for a particular amount to fulfill their business requirements. Existing businesses might choose a gradual movement to save money on infrastructure and administration costs (referred to as "lift and shift"), while a new company might start in the cloud.

Some of the top benefits of Cloud Computing.

1. It's cost-effective

Cloud computing provides a pay-as-you-go or consumption-based pricing model. Rather than paying upfront for a pre-defined amount of computing resources or hardware, you can rent hardware and pay for the resources that you actually use. This consumption-based model brings with it many benefits, including:

- No upfront costs
- No need to purchase and manage costly infrastructure that you may not use to its fullest
- The ability to pay for additional resources only when they are needed
- The ability to stop paying for resources that are no longer needed

This also allows for better cost prediction. Prices for individual resources and services are provided so you can predict how much you will spend in a given billing period based on your expected usage. You can also perform analysis based on future growth using historical usage data tracked by your cloud provider.

2. It's Scalable

You can increase or decrease the resources and services used based on the demand or workload at any given time. Cloud computing supports both vertical and horizontal scaling depending on your needs.

Vertical scaling, also known as "scaling up", is the process of adding resources to increase the power of an existing server. Some examples of vertical scaling are: adding more CPUs, or adding more memory.

Horizontal scaling, also known as "scaling out", is the process of adding more servers that function together as one unit. For example, you have more than one server processing incoming requests.

Scaling can be done manually or automatically based on specific triggers such as CPU utilization or the number of requests and resources can be allocated or de-allocated in minutes.

3. It's Elastic

As your workload changes due to a spike or drop in demand, a cloud computing system can compensate by automatically adding or removing resources.

For example, imagine your website is featured in a news article, leading to a spike in traffic overnight. Since the cloud is elastic, it automatically allocates more computing resources to handle the increased traffic. When the traffic begins to normalize, the cloud automatically de-allocates the additional resources to minimize cost.

Another example is if you are running an application used by employees, you can have the cloud automatically add resources for the peak operating hours during which most people access the application and remove the resources at the usual end of the day.

4. It's Current

When you use the cloud, you're able to focus on what matters: building and deploying applications. Cloud eliminates the burdens of maintaining software patches, hardware setup, upgrades, and other IT management tasks. All of this is automatically done for you to ensure you're using the latest and greatest tools to run your business.

Additionally, the computer hardware is maintained and upgraded by the cloud provider. For example, if a disk fails, the disk will be replaced by the cloud provider. If new hardware update becomes available, you don't have to go through the process of replacing your hardware. The cloud provider will ensure that the hardware updates are made available to you automatically.

5. It's Reliable

When you're running a business, you want to be confident your data is always going to be there. Cloud computing providers offer data backup, disaster recovery, and data replication services to make sure your data is always safe. In addition, redundancy is often built into cloud services architecture so if one component fails, a backup component takes its place. This is referred to as fault tolerance and it ensures that your customers aren't impacted when a disaster occurs.

6. It's Global

Cloud providers have fully redundant datacenters located in various regions all over the globe. This gives you a local presence close to your customers to give them the best response time possible no matter where in the world they are.

You can replicate your services into multiple regions for redundancy and locality, or select a specific region to ensure you meet data-residency and compliance laws for your customers.

7. It's Secure

Cloud providers offer a broad set of policies, technologies, controls, and expert technical skills that can provide better security than most organizations can otherwise achieve. The result is strengthened security, which helps to protect data, apps, and infrastructure from potential threats.

When it comes to physical security – threats to cloud infrastructure, cloud providers invest heavily in walls, cameras, gates, security personnel, and so on, to protect physical assets. They also have strict procedures in place to ensure employees have access only to those resources that they've been authorized to manage.

Then there's digital security. One thing that makes the cloud unique is that you rent compute and storage resources from a shared pool. Plus, data can travel in many different ways – within a datacenter, between datacenters, and over the internet.

For example, you want only authorized users to be able to log into virtual machines or storage systems running in the cloud. The responsibility for securing these resources is shared between you and the cloud provider. Cloud providers offer tools that help you mitigate security threats but you must use these tools to protect the resources you use.

Cloud deployment models

A cloud deployment model defines where your data is stored and how your customers interact with it – how do they get to it, and where do the applications run? It also depends on how much of your own infrastructure you want or need to manage. There are three different cloud deployment models.

1. Public cloud

This is the most common deployment model. In this case, you have no local hardware to manage or keep up-to-date – everything runs on your cloud provider's hardware. In some cases, you can save additional costs by sharing computing resources with other cloud users.

Businesses can use multiple public cloud providers of varying scale. Microsoft Azure is an example of a public cloud provider.

A common use case scenario is deploying a web application or a blog site on hardware and resources that are owned by a cloud provider. Using a public cloud in this scenario allows cloud users to get their website or blog up quickly, and then focus on maintaining the site without having to worry about purchasing, managing or maintaining the hardware on which it runs.

Advantages

- High scalability/agility – you don't have to buy a new server in order to scale
- Pay-as-you-go pricing – you pay only for what you use, no CapEx costs
- You're not responsible for maintenance or updates of the hardware
- Minimal technical knowledge to set up and use - you can leverage the skills and expertise of the cloud provider to ensure workloads are secure, safe, and highly available .

Disadvantages

- There may be specific security requirements that cannot be met by using public cloud
- There may be government policies, industry standards, or legal requirements which public clouds cannot meet
- You don't own the hardware or services and cannot manage them as you may want to
- Unique business requirements, such as having to maintain a legacy application might be hard to meet

2. Private Cloud

In a private cloud, you create a cloud environment in your own datacenter and provide self-service access to compute resources to users in your organization. This offers a simulation of a public cloud to your users, but you remain completely responsible for the purchase and maintenance of the hardware and software services you provide.

A use case scenario for a private cloud would be when an organization has data that cannot be put in the public cloud, perhaps for legal reasons. An example scenario may be where government policy requires specific data to be kept in-country or privately.

A private cloud can provide cloud functionality to external customers as well, or to specific internal departments such as Accounting or Human Resources.

Advantages

- You can ensure the configuration can support any scenario or legacy application
- You can control (and responsibility) over security
- Private clouds can meet strict security, compliance, or legal requirements
- Economies at scale and integration with Azure Security Center

Disadvantages

- You have some initial CapEx costs and must purchase the hardware for startup and maintenance
- Owning the equipment limits the agility - to scale you must buy, install, and setup new hardware
- Private clouds require IT skills and expertise that's hard to come by

3. Hybrid Cloud

A hybrid cloud combines public and private clouds, allowing you to run your applications in the most appropriate location. For example, you could host a website in the public cloud and link it to a highly secure database hosted in your private cloud (or on-premises datacenter).

This is helpful when you have some things that cannot be put in the cloud, maybe for legal reasons. For example, you may have some specific pieces of data that cannot be exposed publicly (such as medical data) which needs to be held in your private datacenter. Another example is one or more applications that run on old hardware that can't be updated. In this case, you can keep the old system running locally, and connect it to the public cloud for authorization or storage.

Advantages

Some advantages of a hybrid cloud are:

- You can keep any systems running and accessible that use out-of-date hardware or an out-of-date operating system
- You have flexibility with what you run locally versus in the cloud
- You can take advantage of economies of scale from public cloud providers for services and resources where it's cheaper, and then supplement with your own equipment when it's not
- You can use your own equipment to meet security, compliance, or legacy scenarios where you need to completely control the environment

Disadvantages

Some concerns you'll need to watch out for are:

- It can be more expensive than selecting one deployment model since it involves some CapEx cost up front
- It can be more complicated to set up and manage

Types of cloud services

When talking about cloud computing, there are three major categories.

Infrastructure as a Service (IaaS)

Infrastructure as a Service is the most flexible category of cloud services. It aims to give you complete control over the hardware that runs your application (IT infrastructure servers and virtual machines (VMs), storage, networks, and operating systems). Instead of buying hardware, with IaaS, you rent it. It's an instant computing infrastructure, provisioned and managed over the internet.

Note

When using IaaS, ensuring that a service is up and running is a shared responsibility: the cloud provider is responsible for ensuring the cloud infrastructure is functioning correctly; the cloud customer is responsible for ensuring the service they are using is configured correctly, is up to date, and is available to their customers. This is referred to as the **shared responsibility model**. IaaS is commonly used in the following scenarios:

- **Migrating workloads.** Typically, IaaS facilities are managed in a similar way as on-premises infrastructure and provide an easy migration path for moving existing applications to the cloud.
- **Test and development.** Teams can quickly set up and dismantle test and development environments, bringing new applications to market faster. IaaS makes scaling development and testing environments, fast and economical.
- **Website hosting.** Running websites using IaaS can be less expensive compared to traditional web hosting.
- **Storage, backup, and recovery.** Organizations avoid the capital outlay and complexity of storage management, which typically requires skilled staff to manage data and meet legal and compliance requirements. IaaS is useful for managing unpredictable demand and steadily growing storage needs. IaaS can also simplify the planning and management of backup and recovery systems.

Platform as a Service (PaaS)

PaaS provides an environment for building, testing, and deploying software applications. The goal of PaaS is to help you create an application quickly without managing the underlying infrastructure. For example, when deploying a web application using PaaS, you don't have to install an operating system, web server, or even system updates.

PaaS is a complete development and deployment environment in the cloud, with resources that enable organizations to deliver everything from simple cloud-based apps to sophisticated cloud-enabled enterprise applications. Resources are purchased from a cloud service provider on a pay-as-you-go basis and accessed over a secure Internet connection.

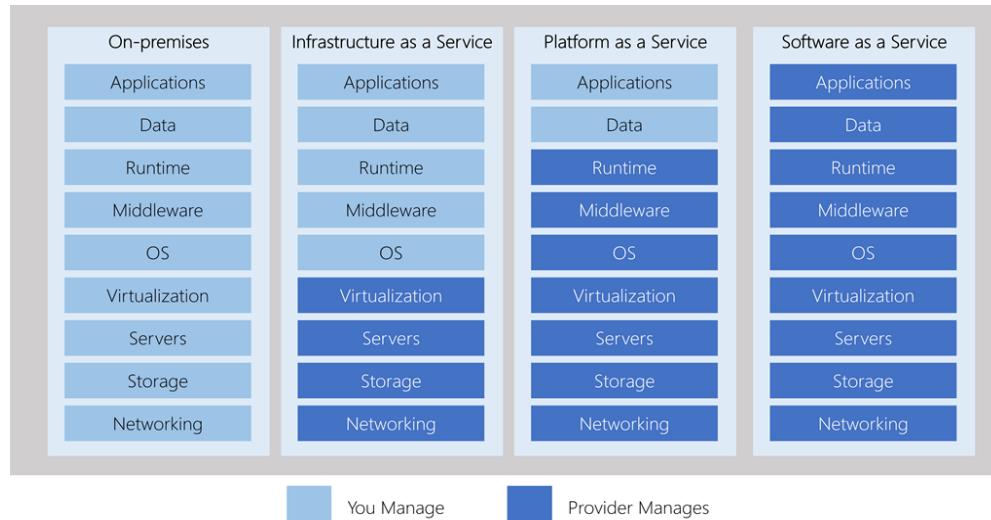
PaaS is commonly used in the following scenarios:

- **Development framework.** PaaS provides a framework that developers can build upon to develop or customize cloud-based applications. Just like Microsoft Excel macro, PaaS lets developers create applications using built-in software components. Cloud features such as scalability, high-availability, and multi-tenant capability are included, reducing the amount of coding that developers must do.

- **Analytics or business intelligence.** Tools provided as a service with PaaS allow organizations to analyze and mine their data. They can find insights and patterns, and predict outcomes to improve business decisions such as forecasting, product design, and investment returns.

Software as a Service (SaaS)

SaaS is software that is centrally hosted and managed for the end customer. It is usually based on an architecture where one version of the application is used for all customers, and licensed through a monthly or annual subscription. Office 365, Skype, and Dynamics CRM Online are perfect examples of SaaS software.



Microsoft Azure

Microsoft Azure is a public cloud computing platform—with solutions including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) that can be used for services such as analytics, virtual computing, storage, networking, and much more. It can be used to replace or supplement our on-premise servers.

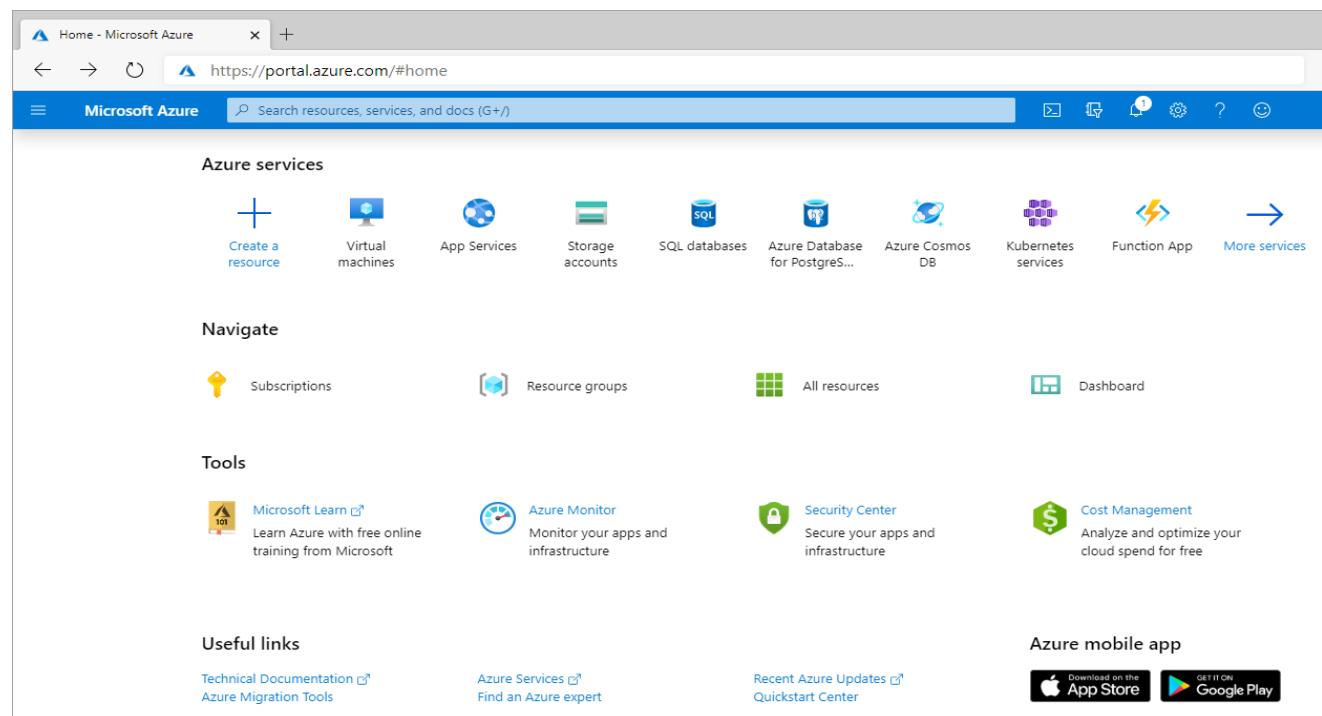
Azure is a fast, flexible, and affordable platform, and its pricing and capabilities make it the best public cloud offering on the market. Azure is a fast, flexible, and affordable platform, and its pricing and capabilities make it the best public cloud offering on the market.

Azure CLI

Azure CLI is a cross-platform command-line program that connects to Azure and executes administrative commands on Azure resources. Cross-platform means that it can be run on Windows, Linux, or macOS. For example, to create a VM, you would open a command prompt window, sign in to Azure using the command az login.

Azure Portal

The Azure portal is a public website that you can access with any web browser. Once you sign in with your Azure account, you can create, manage, and monitor any available Azure services. You can identify a service you're looking for, get links for help on a topic, and deploy, manage, and delete resources. It also guides you through complex administrative tasks using wizards and tooltips. The dashboard view provides high-level details about your Azure environment. You can customize the dashboard by moving and resizing tiles and displaying services you're interested in. The portal doesn't provide any way to automate repetitive tasks. For example, to set up multiple VMs, you would need to create them one at a time by completing the wizard for each VM. This process makes the portal approach time-consuming and error-prone for complex tasks.

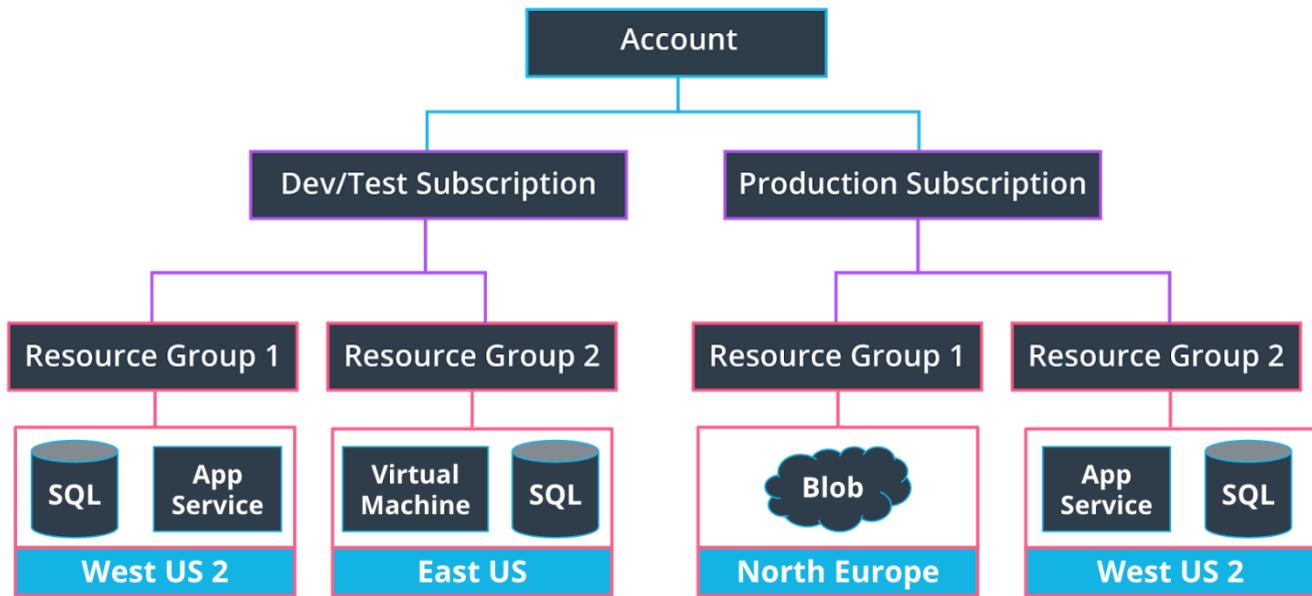


Azure Subscription and Resource Group

Azure uses a hierarchical system to keep resources organized and to manage expenses easily. This starts at the top with your **Azure Account**. The next level down is the **Subscription** level. In this project I have used only one subscription, but it's likely in our day-to-day Azure work that we may be working with more than one. For instance, there may be one subscription for development and testing, and another for production systems.

After the subscription level is where we'll find **Resource Groups**. These help to organize resources we use, such as Virtual Machines and App Services (as well as storage options and other resources) in order to make resource management easier. We may have a resource group for a specific project, or because resource groups are tied to a **Region**, we may have resource groups containing similar resources in multiple locations across the world. A region contains at least one data center but could have multiple data centers that are close by and networked together through a low-latency network. There are over 60 regions available worldwide and available in 140 countries, such as East US and Japan West.

When choosing a region, it's important to consider what we are trying to achieve. For development and testing purposes, we likely want a region close to ourselves, for production purposes we often want resources to be close to our user.



Some points to keep in mind while selecting a region:

- Service availability - Some services may not be available in a particular region.
- Performance - Latency determines network service performance; are you creating resources for yourself or your end user?
- Cost - Costs of services vary by region. If latency isn't an issue, you might want to deploy your services in the cheapest region.

Azure Compute Services

Azure compute is an on-demand computing service for running cloud-based applications. It provides computing resources such as disks, processors, memory, networking, and operating systems. The resources are available on-demand and can typically be made available in minutes or even seconds. You pay only for the resources you use, and only for as long as you're using them.

Azure supports a wide range of computing solutions for development and testing, running applications, and extending your datacenter. The service supports Linux, Windows Server, SQL Server, Oracle, IBM, and SAP. Azure also has many services that can run virtual machines (VMs). Each service provides different options depending on your requirements. Some of the most prominent services are:

- Azure Virtual Machines
- Azure App Service

Azure Virtual Machines

Virtual machines are software emulations of physical computers. They include a virtual processor, memory, storage, and networking resources. VMs host an operating system, and we can install and run software just like a physical computer. When using a remote desktop client, we can use and control the VM as if we were sitting in front of it.

With Azure Virtual Machines, we can create and use VMs in the cloud. Virtual Machines provides infrastructure as a service (IaaS) and can be used in different ways. When we need total control over an operating system and environment, VMs are an ideal choice. Just like a physical computer, we can customize all the software running on the VM. This ability is helpful when we're running custom software or custom hosting configurations.

With Azure Virtual Machines, we can create and use VMs in the cloud. VMs provide infrastructure as a service (IaaS) in the form of a virtualized server and can be used in many ways. Just like a physical computer, we can customize all of the software running on the VM.

VMs are an ideal choice when we need:

- Total control over the operating system (OS).
- The ability to run custom software.
- To use custom hosting configurations.

An Azure VM gives us the flexibility of virtualization without having to buy and maintain the physical hardware that runs the VM. We will still need to configure, update, and maintain the software that runs on the VM.

We can create and provision a VM in minutes when you select a preconfigured VM image. Selecting an image is one of the most important decisions we'll make when you create a VM. An image is a template used to create a VM. These templates already include an OS and often other software, like development tools or web hosting environments.

Benefits of VMs are:

- VMs allow us full access and control of the VM.
- Lower up-front cost compared to purchasing and maintaining hardware.
- Support of both Linux and Windows VMs.
- Multiple types to choose from, such as compute or memory-optimized VMs, along with varying amounts of CPU, RAM and storage.
- VMs allow for the installation of custom images and are an excellent choice for migrating from an on-premises server to the cloud.
- Multiple VMs can be grouped to provide high availability, scalability, and redundancy. There are two options when it comes to scaling—Virtual Machine Scale Sets and Load Balancers. These will be covered in a different course.

Limitations of VMs are:

- They are more expensive
- They can be more time consuming for the developer than other compute options

Examples of when to use VMs

- 1. During testing and development.** VMs provide a quick and easy way to create different OS and application configurations. Test and development personnel can then easily delete the VMs when they no longer need them.
- 2. When running applications in the cloud.** The ability to run certain applications in the public cloud as opposed to creating a traditional infrastructure to run them can provide substantial economic benefits. For example, an application might need to handle fluctuations in demand. Shutting down VMs when we don't need them or quickly starting them up to meet a sudden increase in demand means we pay only for the resources we use.
- 3. When extending your datacenter to the cloud.** An organization can extend the capabilities of its own on-premises network by creating a virtual network in Azure and adding VMs to that virtual network. Applications like SharePoint can then run on an Azure VM instead of running locally. This arrangement makes it easier or less expensive to deploy than in an on-premises environment.
- 4. During disaster recovery.** As with running certain types of applications in the cloud and extending an on-premises network to the cloud, we can get significant cost savings by using an IaaS-based approach to disaster recovery. If a primary datacenter fails, you can create VMs running on Azure to run your critical applications and then shut them down when the primary datacenter becomes operational again.

Azure App Services

With Azure App Service, we can quickly build, deploy, and scale enterprise-grade web, mobile, and API apps running on any platform. We can meet rigorous performance, scalability, security, and compliance requirements while using a fully managed platform to perform infrastructure maintenance. App Service is a platform as a service (PaaS) offering. This also enables us to build and host web apps, background jobs, mobile back-ends, and RESTful APIs in the programming language of our choice without managing infrastructure. It offers automatic scaling and high availability. App Service supports Windows and Linux and enables automated deployments from GitHub, Azure DevOps, or any Git repo to support a continuous deployment model. This platform as a service (PaaS) environment allows us to focus on the website and API logic while Azure handles the infrastructure to run and scale our web applications.

Types of app services

- Web apps
- API apps
- Web Jobs
- Mobile apps

App Service handles most of the infrastructure decisions you deal with in hosting web-accessible apps. Some of those decisions are:

- Deployment and management are integrated into the platform.
- secured Endpoints.
- Scalability of sites quickly to handle high traffic loads.
- The built-in load balancing and traffic manager

Benefits of using an App Service :

- Support of multiple languages, such as .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python
- High availability, auto-scaling and support of both Linux and Windows environments.
- Continuous deployment model using GitHub, Azure DevOps, or any Git repo.
- Vertical or Horizontal scaling. Vertical scaling increases or decreases resources allocated to our App Service, such as the amount of vCPUs or RAM, by changing the App Service pricing tier. Horizontal scaling increases or decreases the number of Virtual Machine instances our App Service is running.
- You can set the amount of hardware allocated to host your application, and cost varies based on the plan you choose. There are three different tiers - Dev/Test, Production, and Isolated. We'll be using the free option within Dev/Test for the exercises in this course.

Limitations of an App Service :

- You have limited access to the host server, so you are unable to control the underlying OS or install software on the server.
- You're always paying for the service plan, even if your services or application isn't running.
- There are hardware limitations, such as a maximum of 14GB of memory and 4 vCPU cores per instance
- While they support multiple languages, as noted in the benefits above, they are limited to just using those languages (as of when this course was built).

Each of these has their own use cases, although sometimes there is still some ambiguity on when to use each. Virtual Machines are usually better when you need control of the underlying operating system or are using custom software to support our needs whereas an app service is typically better for lightweight applications and services, especially when we don't have the need for high performance compute services. Additionally, we'll need to take into consideration the hardware limitations of App Services, as noted above.

**I prefer to deploy my further mentioned project in app service
The reason due to which I chose to deploy my application as
mentioned later in App Service is as follows:**

Overview :-

We can deploy our application with either means whether its an app or an virtual machine. Here, App service is a part of Platform as a Service or (PaaS) which means I am provided with the whole building and now I have to develop the app only and it is also specialized for HTTP-based service for hosting web application whereas in Virtual Machine I have access to the Infrastructure as a Service which means (IaaS), which means here I will be responsible for the Operating system, the server I'll use to deploy the app into here I will also be responsible for the ram required, computation power required and many other factors. Both the platforms are good for deploying the application but here I have chosen the app service as for this app I do not require to control much of the computation power or the storage or anything else as this is a small scale application. Whereas If I was to develop a large scale application which might grow over a period of time I must have chosen the Virtual Machine. In choosing the App service we will only have to pay for the resources as we use them so it is more cost benifitil as compared to the virtual machine. Also Deploying Virtual

Machine is a time consuming process whereas deploying an app service is not that time consuming so it is a better option.

Cost Analysis:-

Virtual Machines cost three times the price of the app service and in app service i will be paying for recourses as i use them but in virtual machine ill be paying for some recourse even while i am not using them

Scalability

we will have to integrate the load balancer and auto scaling functions in the Virtual machoines but in app service these functions such as load balancer and auto scaling are pre integrated

Availability and workflow

In availability and workflow options virtuial machines would be better as they are available for aprox 99.95 percent of time and have better workflow as mentioned in the first para but as it is a small scale application so we can ignore these factors.

Changes in deployment which might have changed my decision

If this would have been a large scale application then i would have prefered virtual machine as it provides better availability and workflow in this case we can surely ignore the cost and time factors as good thing take time to develop.

Azure Storage

Azure Storage accounts

An Azure storage account can store data objects you create, such as blobs and files. This storage account provides a unique namespace in Azure for your data, and every item that you store has an address that includes your unique account name. It's worth noting the name of the storage account can contain only lowercase letters and numbers.

In this project, I have mostly utilize *General-purpose v2* storage accounts, which provide support and the latest features for Azure Storage services such as blobs, Data Lake Gen2, Files, Disks, Queues, and Tables. This type of storage account is recommended for most scenarios when using Azure Storage services.

Storage accounts can contain multiple blob containers within them, such as "images" and "movies" containers, to organize different data files. From there, each container can have many blobs inside of them (the files themselves).

Benefits of using azure for storage

Some of the benefits of using Azure for storage are:

- Automated backup and recovery
- An option to replicate data at multiple data centers worldwide to help prevent outages from unplanned events, such as hardware failure.
- Data analytics support
- Data encryption for added security

- Support for the storage of multiple data types. Azure is designed to hold 3 main types data—relational data, non-relational data or NoSQL data, and unstructured data such as images.
- Scale up or scale out when demand is high and scale back when demand is low.
- Eliminate the expense of having to purchase, install, configure, and maintain on premises hardware.

Azure Storage Options

Among many storage options, some of the Azure Storage options are:

- Azure SQL Server and SQL Database
- Azure Blob Storage
- Azure Cosmos DB
- Disk Storage
- Azure Data Lake Storage
- HPC Cache

Azure SQL Databases

Azure SQL databases are used for structured, relational data. In Azure, we also need to first create a related SQL server to hold the database, although we can create multiple SQL databases under a single SQL server. Azure SQL databases do not have a free tier option like we saw with App Services earlier; however, when you sign up for a new free Azure account, you get 12 months of up to 250GB of Azure SQL database storage. The lowest tier is currently ~\$5 / month otherwise for a small amount of storage.

Azure Blob Storage

A **Binary Large Object** or **Blob** is a data type that can store unstructured (binary) data and is used to store things like images or videos in a database. Blobs are not the go-to choice when it comes to structured data that is queried frequently, such as user profile information. They have higher latency than memory and local disk and don't have the indexing features that make databases efficient at running queries. Blobs are commonly used in combination with databases to store non-queryable data, such as a profile picture for a user's profile. Each user record in the database would include the URL of the blob containing the user's photo.

Blob storage life cycle

These are set based on how frequently data will be accessed, with hot accessed the most frequently, with relatively lower latency for requests, but higher costs.

Blob storage offers a rule-based policy you can use to transition our data between these tiers to optimize performance and cost. This is shown again in the below graphic - while our blobs may start in a Hot container, they could be moved to a Cool container if they have not been modified in the past 30 days. From there, if they have not been modified in 90 days, they could be transitioned to Archive storage; you could then perhaps delete them after a year of non-usage.



Azure Cosmos DB

Azure Cosmos DB is a fully managed NoSQL database for modern app development. Single-digit millisecond response times, and automatic and instant scalability, guarantee speed at any scale. Business continuity is assured with SLA-backed availability and enterprise-grade security. App development is faster and more productive thanks to turnkey multi region data distribution anywhere in the world, open source APIs and SDKs for popular languages. As a fully managed service, Azure Cosmos DB takes database administration off your hands with automatic management, updates and patching. It also handles capacity management with cost-effective serverless and automatic scaling options that respond to application needs to match capacity with demand.

Azure Disk Storage

VM uses disks as a place to store an operating system, applications, and data in Azure. All virtual machines have at least two disks- a Windows operating system disk and a temporary disk. Both the operating system disk and the image are virtual hard disks (VHDs) stored in an Azure storage account. The VHDs used in Azure is .vhf files stored as page blobs in a standard or premium storage account in Azure. Virtual machines can also have one or more data disks that are also stored as VHDs.

Azure Data Lake Storage

Microsoft Azure Data Lake Storage (ADLS) is a fully managed, elastic, scalable and secure file system that supports HDFS semantics and works with the Apache Hadoop ecosystem. It provides industry-standard reliability, enterprise-grade security and unlimited storage that is suitable for storing a large variety of data. It is built for running large-scale analytics systems that require large computing capacity to process and analyze large amounts of data. Data stored in ADLS can easily be analyzed using Hadoop frameworks like MapReduce and Hive.

HPC Cache

Azure HPC Cache speeds access to your data for high-performance computing (HPC) tasks. By caching files in Azure, Azure HPC Cache brings the scalability of cloud computing to your existing workflow. This service can be used even for workflows where your data is stored across WAN links, such as in your local datacenter network-attached storage (NAS) environment.

Azure HPC Cache is easy to launch and monitor from the Azure portal. Existing NFS storage or new Blob containers can become part of its aggregated namespace, which makes client access simple even if you change the back-end storage target.

Security and monitoring

Azure security responsibilities

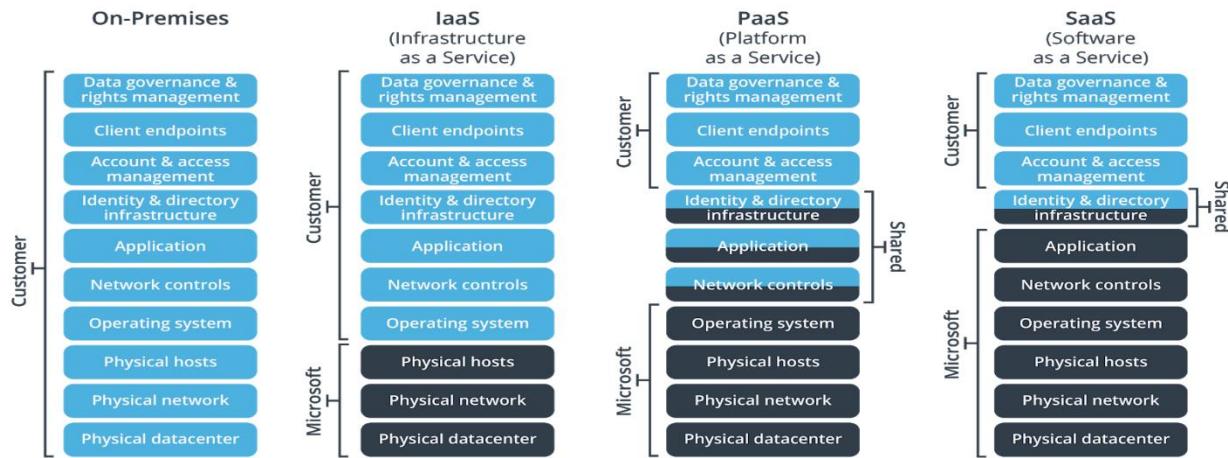
The security responsibilities between cloud developers and the cloud provider:-

Security in information systems, also known as "cybersecurity," is created, produced, and distributed by people. The systems and software they use are merely tools. No one can design an information system to be inherently secure. Security is a process that either enables that system to operate in a secure fashion, or to be redeveloped or replaced with another system that can.

Cloud services are an extension of the data center. They enable the functionality that an organization provides to its customers to be served from a multiplicity of locations. They allow that functionality to be distributed and yet remain cohesive and interoperative. And most importantly to some businesses, they also distribute the risks involved with managing and securing highly distributed services to professionals outside the organization. Companies such as Amazon and Microsoft employ legions of world-class network-security engineers. Organizations that use public cloud platforms are the beneficiaries of their expertise.

Security responsibilities are also different among different technology stacks weather its on-premises, IaaS, PaaS or SaaS . For the cloud developer, on-premises still puts all security responsibilities onto them. With IaaS, the security of physical hardware shifts to the cloud provider. With PaaS, the operating system security shifts to be the responsibility of the cloud provider, while network, application and identity security are a shared responsibility. With SaaS, the network

and application shift responsibility to the cloud provider, but identity stays as a shared responsibility.



Security responsibilities that always stay with the cloud developer:

- Account and access management (E.g. do you give the appropriate people the right access, and revoke access when appropriate?)
- Client endpoints (E.g. do you secure the appropriate endpoints to confidential information?)
- Data Governance and Rights Management (E.g. do you prevent employees from emailing confidential documents to third parties?)

Security options in azure

There are a ton of services and topics to consider in Azure, each with their own respective use cases. Some of them are listed below

- *Azure Active Directory*: Provides single sign-on (SSO) and multi-factor authentication (MFA) capabilities, such as Sign in with Microsoft
- *App Configuration*: Stores application settings in one secure location
- *Key Vault API*: Stores application keys and secrets in one secure location
- *Managed Identities*: A part of Azure Active Directory; This helps streamline providing an app or app user access to other Azure resources
- *Shared Access Signatures*: Give external parties certain limited access (determined by you) to different Azure resources
- *Role-Based Access Controls (RBAC)*: Help internally manage who has access to what resources, and what they can do to said resources
- *Azure Monitor*: Provides a wide range of monitoring services such as log analytics, metrics, alerts, and much more
- *Application Insights*: Part of Azure Monitor; This helps monitor performance and other key metrics

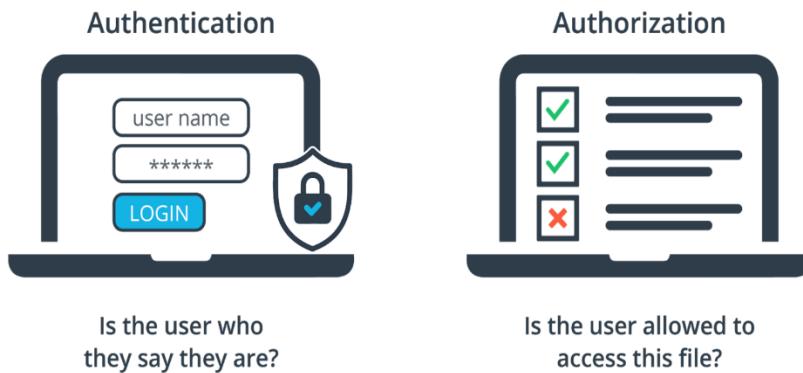
Azure Active directory

Azure Active Directory is Microsoft's solution for single sign-on (SSO) and multi-factor authentication (MFA). We'll be using it in combination with the Microsoft Authentication Library (MSAL) to use "Sign in with Microsoft" buttons in an app, although it can be used more broadly for identity management purposes within an organization. The "tenant" in Azure AD is usually equivalent to an organization.

As you get started with Azure AD, there may be a limit to your access if you are using an enterprise account .

OAuth2 with MSAL

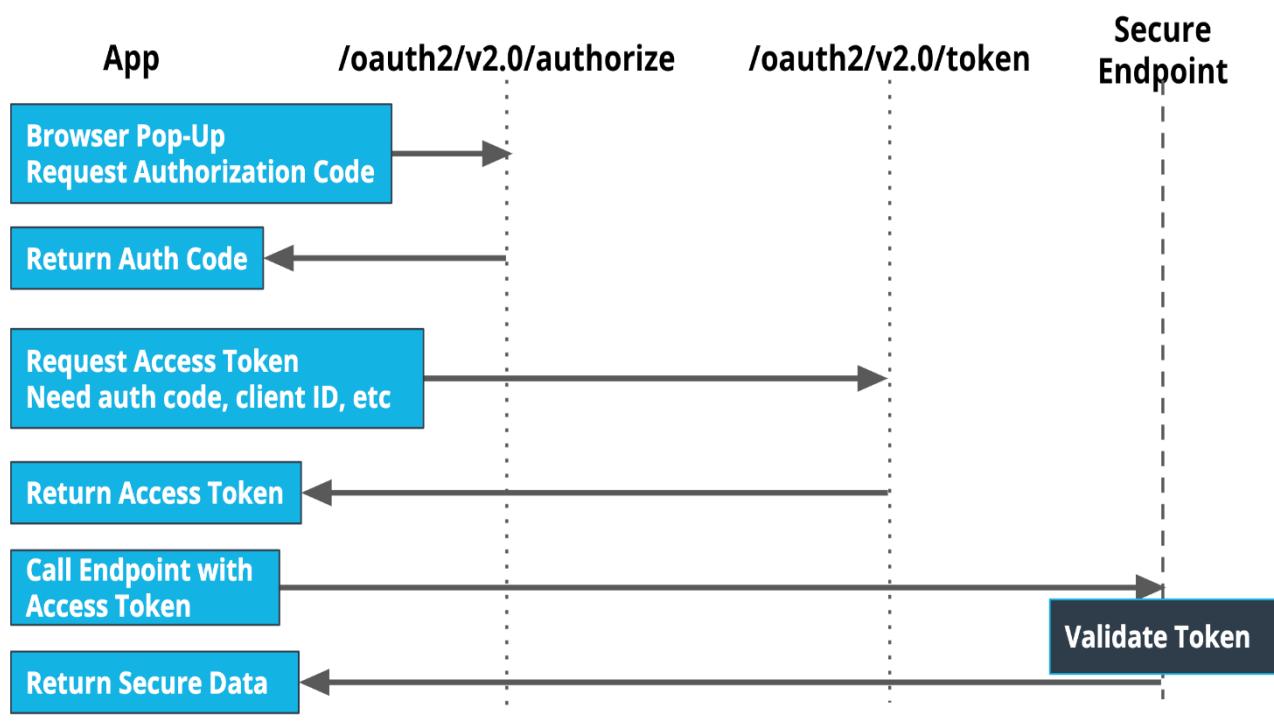
Authentication is the process of checking if the user is who they say they are, and **authorization** is the process of checking if the user is allowed to access data and what they can do with the data.



Authentication vs Authorization

OAuth 2.0 is the industry-standard protocol for authorization. Instead of creating apps that each maintain their username and password information, apps can delegate that responsibility to a centralized identity provider.

Azure AD is the centralized identity provider in the cloud that we'll use to implement OAuth 2.0, or more specifically "Sign in with Microsoft", in our case. The Microsoft Authentication Library (MSAL) is a Python library we'll use to implement "Sign in with Microsoft" functionality in an app. Active Directory Authentication Library (ADAL) is an older library that does not support personal Microsoft accounts.



A simplified workflow for OAuth 2.0 with MSAL

Benefits of Logging

Implementing logging in your applications can help with:

- Troubleshooting problems or preventing potential new ones
- Improving application performance or maintainability
- Automating operations that would otherwise require manual intervention
- Also Azure gives developers the ability to monitor metrics, Use App-based logging, Send logs to storage and Create alerts.

Logging Considerations in a Flask App

When building a Flask application, print statements won't show up like they usually would in a regular Python application. You'll need to use the built-in logger Flask has to log events. You're able to set the minimum severity level of the events you want to capture; for example, you could set the logger to only capture events at a warning level or above.

In my project I have used Azure Active Directory ,Oauth2.0 with MSAL and Azure monitoring

Introduction to the project

About the Project

In this project, I was given a simple Content Management System (CMS) for articles, where a user can log in, view published articles, and publish new articles.

An article consists of a title, author, and body of text (to be stored in an Azure SQL Server) along with an image (to be stored in Azure Blob Storage).

The CMS includes the following components:

- A webapp using Python with the Flask framework.
- A SQL database that contains a user table and an article table for the webapp to query.
- A Blob Storage container where images are stored.

Here, I was in charge of creating and managing the resources necessary to fully deploy both the necessary storage and compute resources for the app to Azure.

As part of the project, I have also analyzed which resource best fits for app deployment between a VM or App Service.

Lastly, I have added in an authentication option to Sign in with Microsoft, as well as logging for successful and unsuccessful logins as shown in further screenshots.

Screenshots of Parts of my project

- 1) Resource Group
- 2) App service
- 3) App Service Plan
- 4) SQL Server and Database
- 5) Storage Account for Image Storage
- 6) Working Site
- 7) Working Site logs
- 8) Invalid Login
- 9) Invalid Attempt Logs
- 10) Post Edit
- 11) Create New Post
- 12) Valid Login Logs
- 13) Blob Storage
- 14) SQL Database being Updated
- 15) URI Redirects

Screenshots of my project

1) Resource Group

The screenshot shows the Microsoft Azure portal interface for managing a Resource Group named 'CMS'. The left sidebar navigation bar includes Home, CMS (Resource group), Overview, Activity log, Access control (IAM), Tags, Events, Settings (Deployments, Security, Policies, Properties, Locks), Cost Management (Cost analysis, Cost alerts (preview), Budgets, Advisor recommendations), and Help & Support. The main content area displays a table of resources with columns for Name, Type, and Location. The table shows six resources: 'ASP-CMS-b74f' (App Service plan, West US 2), 'CMS (cms-adi/CMS)' (SQL database, West US 2), 'cms-adi' (SQL server, West US 2), 'cmsadi' (App Service, West US 2), 'cmsadi' (Storage account, West US 2), and 'login123' (Storage account, West US 2). A search bar at the top is set to 'Search resources, services, and docs (G+ /)'. The top navigation bar shows several open tabs related to Azure and other Microsoft services.

Name	Type	Location
ASP-CMS-b74f	App Service plan	West US 2
CMS (cms-adi/CMS)	SQL database	West US 2
cms-adi	SQL server	West US 2
cmsadi	App Service	West US 2
cmsadi	Storage account	West US 2
login123	Storage account	West US 2

2)App Service Deployed

The screenshot shows the Microsoft Azure portal interface for a deployment named "Microsoft.Web-WebApp-Portal-232ab425-8b3f". The main panel displays a green checkmark indicating "Your deployment is complete". Deployment details are listed, including the deployment name, subscription (Visual Studio Enterprise Subscription), correlation ID, and resource group (CMS). A "Go to resource" button is present. The left sidebar shows navigation options like Overview, Inputs, Outputs, and Template. The right sidebar includes links for Security Center, Free Microsoft tutorials, and Work with an expert.

3)Deployed app service plan

The screenshot shows the Microsoft Azure portal interface for a deployment named "Microsoft.Web-ASP-Portal-86db4094-8f16". The main panel displays a green checkmark indicating "Your deployment is complete". Deployment details are listed, including the deployment name, subscription (Visual Studio Enterprise Subscription), correlation ID, and resource group (CMS). A "Go to resource" button is present. The left sidebar shows navigation options like Overview, Inputs, Outputs, and Template. The right sidebar includes links for Security Center, Free Microsoft tutorials, and Work with an expert.

4) SQL Server and Sql Database deployed

The screenshot shows the Microsoft Azure portal with the URL <https://portal.azure.com/#blade/HubsExtension/DeploymentDetailsBlade/overview/id/%2fsubscriptions%2F4f1c9224-64d1-439f-a3cc-3b3f1649a81%2fresourceGroups%2FCMS%2F...>. The page title is "Microsoft.SQLDatabase.newDatabaseNewServer_71f79e72688045aea700f | Overview". The main content area displays a green checkmark indicating "Your deployment is complete". It shows deployment details for a "Subscription: Visual Studio Enterprise Subscription" and a "Resource group: CMS". A table lists the resources created, their type, status, and operation details:

Resource	Type	Status	Operation details
cms-adi/cms-adi	Microsoft.Sql/servers/databases	Created	Operation details
cms-adi	Microsoft.Sql/servers	OK	Operation details
cms-adi/ClientIp-2021-11-6_18-15-27	Microsoft.Sql/servers/firewallrules	Created	Operation details
cms-adi/Default	Microsoft.Sql/servers/connectionPolicies	OK	Operation details
cms-adi/AllowAllWindowsAzureips	Microsoft.Sql/servers/firewallrules	Created	Operation details
cms-adi	Microsoft.Sql/servers	Created	Operation details

On the right side, there are links to "Security Center", "Free Microsoft tutorials", "Work with an expert", and "Find an Azure expert".

5) Deployed Storage Account for image storage

The screenshot shows the Microsoft Azure portal with the URL <https://portal.azure.com/#/resource/subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/endpoints>. The page title is "cmsadi | Endpoints". The left sidebar shows navigation options like Overview, Activity log, Tags, and Data storage. The main content area displays the endpoints for the storage account:

- Provisioning state:** Succeeded (6/9/2021, 4:49:52 PM)
- Storage account resource ID:** [/subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi](https://subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi)
- Blob service:** Resource ID: [/subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/blobServices/default](https://subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/blobServices/default); Endpoint: <https://cmsadi.blob.core.windows.net/>
- File service:** Resource ID: [/subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/fileServices/default](https://subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/fileServices/default); Endpoint: <https://cmsadi.file.core.windows.net/>
- Queue service:** Resource ID: [/subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/queueServices/default](https://subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/queueServices/default); Endpoint: <https://cmsadi.queue.core.windows.net/>
- Table service:** Resource ID: [/subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/tableServices/default](https://subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/tableServices/default)

6) Working Site before login

The screenshot shows a web browser window with multiple tabs open. The active tab is titled 'Sign In - My Flask Application'. The page content is a 'Sign In' form for 'Article CMS'. It includes fields for 'Username' and 'Password', a 'Remember Me' checkbox, and a 'Sign In' button. Below the form, there is an 'OR' link followed by a 'Sign in with Microsoft' button.

7) Working site logs

The screenshot shows the Microsoft Azure portal with the search bar set to 'Search resources, services, and docs (G+)'. The main area displays the 'Log stream' for the 'cms-adi' App Service. The log output shows the deployment process and initial startup logs, including Docker logs and application startup messages. Key logs include:

```
Configure : 10.10.23.565313
Setting Up Routes : 10.10.24.999564
Exiting Configure : 10.10.25.048594
Hosting environment: Production
Content root path: /opt/Kudu
Now listening on: http://0.0.0.0:8181
Application started. Press Ctrl+C to shut down.

/home/LogFiles/2021_06_10_lwlsdlwk00002F_docker.log (https://cms-adi.scm.azurewebsites.net/api/vfs/LogFiles/2021_06_10_lwlsdlwk00002F_docker.log)Please use https://aka.ms/linux-diagnostics to enable logging to see container logs here.

Initiating warmup request to container cms-adi_2_eed71d40 for site cms-adi
Container cms-adi_2_eed71d40 for site cms-adi initialized successfully and is ready to serve requests.

Starting container for site
docker run -d -p 3321:8000 --name cms-adi_3_f68907a7 -e WEBSITE_SITE_NAME=cms-adi -e WEBSITE_AUTH_ENABLED=False -e WEBSITE_ROLE_INSTANCE_ID=0 -e WEBSITE_HOSTNAME=cms-adi.azurewebsites.net -e WEBSITE_INSTANCE_ID=99f6860ee93da7767f21d91709023125d4bd849c3d426402983debd70d8adfa3 appsvc/python:3.7_20210421.4
Logging is not enabled for this container.Please use https://aka.ms/linux-diagnostics to enable logging to see container logs here.

Initiating warmup request to container cms-adi_3_f68907a7 for site cms-adi
Container cms-adi_3_f68907a7 for site cms-adi initialized successfully and is ready to serve requests.

/home/LogFiles/webssh/pm2.log (https://cms-adi.scm.azurewebsites.net/api/vfs/LogFiles/webssh/pm2.log)

Ending Log Tail of existing logs ---
Starting Live Log Stream ---
```

8) Site after Invalid login

The screenshot shows a web browser with multiple tabs open. The active tab is titled "Article CMS" and displays a sign-in form. The URL in the address bar is "cms-adi.azurewebsites.net/login". The page shows an error message: "Invalid username or password". Below the message are fields for "Username" and "Password", a "Remember Me" checkbox, and a "Sign In" button. There is also an "OR" link followed by a "Sign in with Microsoft" button. The browser's toolbar at the top includes various icons and links.

9) Invalid login attempt logs in Azure portal

The screenshot shows the Microsoft Azure Log Stream interface. The left sidebar lists various service categories like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Security, Events (preview), Deployment, Quickstart, Deployment slots, Deployment Center, Settings, Configuration, Authentication, Authentication (classic), and Application Insights. The main pane displays log entries. One entry stands out as an error: "2021-06-13T07:15:24.920215875Z: [ERROR] [2021-06-13 07:15:24.919] ERROR in views: Invalid login attempt!". This log entry is preceded by several informational messages related to user authentication attempts.

10) Valid login attempt(edit post)

Article CMS

Edit Post

Title

Author

Body

Image
 No file chosen

11)Valid Login Attempt Create Post

Article CMS

Edit Post

Title

Author

Body

Image
 No file chosen

12)Valid login attempt

This screenshot shows the Microsoft Azure Log Stream interface for an App Service named 'cms-adi'. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Security, Events (preview), Deployment (Quickstart, Deployment slots, Deployment Center), Settings, Configuration, Authentication, and Application insights. The main pane displays a log stream with the following entries:

```
2021-06-13T07:14:14.155842353Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/content/bootstrap.min.css HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.155842353Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/scripts/modernizr-2.6.2.js HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.155842353Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/scripts/jquery-1.10.2.js HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.155842353Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/scripts/bootstrap.js HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.155842353Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/scripts/respond.js HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.155842353Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/images/sign-in-with-ms.svg HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.155842353Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "POST /login?next=%2F HTTP/1.1" 302 209 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.155842353Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET / HTTP/1.1" 200 3530 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
Ending Log Tail of existing logs ---
```

This screenshot shows the Microsoft Azure Log Stream interface for an App Service named 'cms-adi'. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Security, Events (preview), Deployment (Quickstart, Deployment slots, Deployment Center), Settings, Configuration, Authentication, and Application insights. The main pane displays a log stream with the following entries:

```
2021-06-13T07:15:04 Welcome, you are now connected to log-streaming service.
Starting Log Tail -n 10 of existing logs ---
/appsvctmp/volatile/logs/runtime/061cd0b54c7a9a0560acbcdcaadccf1aa33a0b8fe729508a2707bff0bb90b49b.log
2021-06-13T07:14:13.830733514Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:13 +0000] "GET /login?next=%2F HTTP/1.1" 200 3010 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.145938590Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/content/bootstrap.min.css HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.274766877Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/scripts/modernizr-2.6.2.js HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.274766877Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/scripts/jquery-1.10.2.js HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.274766877Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/scripts/bootstrap.js HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.274766877Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:14 +0000] "GET /static/scripts/respond.js HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.274766877Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:15 +0000] "GET /static/images/sign-in-with-ms.svg HTTP/1.1" 200 0 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.274766877Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:15 +0000] "POST /login?next=%2F HTTP/1.1" 302 209 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
2021-06-13T07:14:14.274766877Z: [INFO] 172.16.0.1 - - [13/Jun/2021:07:14:15 +0000] "GET / HTTP/1.1" 200 3530 "http://cms-adi.azurewebsites.net/login?next=%2F" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.101 Safari/537.36"
Ending Log Tail of existing logs ---
```

13) Working blob storage

The screenshot shows the Microsoft Azure portal interface. The left sidebar navigation includes 'Overview', 'Activity log', 'Tags', 'Diagnose and solve problems', 'Access Control (IAM)', 'Data migration', 'Events', 'Storage Explorer (preview)', 'Containers', 'File shares', 'Queues', 'Tables', 'Networking', and 'Security + networking'. The main content area displays the 'Endpoints' section for the storage account 'cmsadi'. It lists various service endpoints with their URLs:

- Provisioning state: Succeeded
- Created: 6/9/2021, 4:49:52 PM
- Storage account resource ID: /subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi
- Blob service Resource ID: /subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/blobServices/default
- Blob service URL: https://cmsadi.blob.core.windows.net/
- File service Resource ID: /subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/fileServices/default
- File service URL: https://cmsadi.file.core.windows.net/
- Queue service Resource ID: /subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/queueServices/default
- Queue service URL: https://cmsadi.queue.core.windows.net/
- Table service Resource ID: /subscriptions/15def75a-ed17-49f0-b6c8-bb013d920207/resourceGroups/CMS/providers/Microsoft.Storage/storageAccounts/cmsadi/tableServices/default

14) SQL storage account being updated after deployment

The screenshot shows the Microsoft Azure portal interface. The left sidebar navigation includes 'Overview', 'Activity log', 'Tags', 'Diagnose and solve problems', 'Quick start', 'Query editor (preview)', 'Power Platform' (with options for Power BI, Power Apps, and Power Automate), 'Settings' (with Compute + storage, Connection strings, Properties, and Locks), and 'Power BI' (with preview). The main content area displays the 'Query editor (preview)' for the 'CMS (adityacms)' database. The interface shows a query editor with four tabs (Query 1, Query 2, Query 3, Query 4) and a results viewer. The current query is:

```
1 SELECT TOP (1000) * FROM [dbo].[POSTS]
```

The results table shows the following data:

id	title	author	body	image_path	timestamp	use
1	Lorem ipsum dolor s...	John Smith	Proin sit amet mi orn...	2021-06-09T10:40:4...	2021-06-09T10:40:4...	1
2	Matterial classify	Rohit	This is a test post	70VKKIDML8PL2W...	2021-06-10T13:06:4...	1
3	ME	Aditya Garg	He is enrolled as a c...	30A3EB2F3AKKKMV...	2021-06-10T13:57:3...	1

15) URI redirects

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes links for 'Home Page - My Flask App', 'Deploy An Article CMS to A...', 'Udacity Reviews', 'Microsoft Learn Student A...', 'cms-adi - Microsoft Azure', and several other tabs like 'B.Tech 4yr Batch201...', 'Product Hunt - The...', 'Hacker News', 'GitKraken Ambassa...', 'Twitter Developers', 'How it works | Goo...', 'Other bookmarks', and 'Reading list'. The user's name 'adityagarg.ag2000@gmail.com' and 'DEFAULT DIRECTORY' are visible in the top right.

The main content area shows the 'cms-adi' application configuration. The left sidebar has sections for Overview, Quickstart, Integration assistant, Manage (Branding, Authentication, Certificates & secrets, Token configuration, API permissions, Expose an API, App roles, Owners, Roles and administrators, Manifest), and Support + Troubleshooting. The 'Authentication' section is currently selected.

In the main pane, under 'Platform configurations' for 'Web', there is a 'Redirect URIs' section. It contains a table with one row:

URI
https://cms-adi.azurewebsites.net/getAToken

Below the table is a 'Front-channel logout URL' input field containing 'https://cms-adi.azurewebsites.net/login'.

References

Azure

<https://docs.microsoft.com/en-us/learn/modules/azure-architecture-fundamentals/>

https://docs.microsoft.com/en-us/learn/modules/align-requirements-in-azure/?WT.mc_id=udacity_learn-wwl

Azure portal

https://docs.microsoft.com/en-us/learn/modules/tour-azure-portal/?WT.mc_id=udacity_learn-wwl

Compute services

<https://docs.microsoft.com/en-us/learn/modules/azure-compute-fundamentals/>

Storage

<https://docs.microsoft.com/en-us/learn/modules/azure-database-fundamentals/>

Security responsibilities

https://docs.microsoft.com/learn/patterns/manage-security-operations/?WT.mc_id=udacity_learn-wwl

Azure Active Directory

https://docs.microsoft.com/en-us/learn/modules/secure-aad-users-with-mfa/?WT.mc_id=udacity_learn-wwl

Monitoring and Logging

https://docs.microsoft.com/azure/app-service/web-sites-monitor?WT.mc_id=udacity_learn-wwl

Skills I've built these past few weeks

- Creating and deploying virtual machines and app services
- Setting up cloud storage services, such as Azure SQL Databases and Blob Storage, and connecting them to an app
- Adding OAuth 2.0 capabilities with the Microsoft Authentication Library and Azure Active Directory to your app
- Implementing monitoring and logging solution