Benefits of using Cloud Services:

Availability, scalability, reliability, predictability, security and governance, and manageability.

High Availability (HA)

Ability of a system to remain operations to users during planned or unplanned outages. It is important to remember that it is near impossible to have 100% availability.

Planned outages include operating system security patches, application updates, hardware replacement, migrating to a new hosting provider.

Unplanned outages include hardware failure, network disruptions, power outages, natural disasters, cyber attacks, software bugs, poor scaling/architecture design.

Methods to mitigate planned outages:

1. Gradual deployment strategy (do not deploy to all the servers at once). Start with handful servers and gradually grow deployment.

2. Testing and monitoring deployment (errors (files missing, corrupted) are increasing over the time).

3. Easy rollback plan (have a plan to go back if there are major issues). MS Azure has tools to make rolling back easy.

4. Small deployments (less features and less changes in every deployment)

5. Frequent deployments (become expert at deployments)

6. Automation (instead of manual deployments, follow standards)

Methods to mitigate unplanned outages:

1. Every single core component has redundancy (we will not have a single server, single web app, single network, single region dependency for application). This may increase the cost, but it will help to mitigate the unplanned outages.

2. Use Azure's built-in features for availability: Availability Sets, Availability Zones, Cross-Region Load Balancing/Front Door.

3. Constant health monitoring/probes (real time monitoring systems).

4. Automation (automate fall back strategies).

5. Strong security practices (for cyber attacks, ...).

6. Be geographically distributed.

7. Have a disaster recovery plan (similar to fire evacuation plan).

8. Test that disaster recovery plan (similar to fire drill).

9. Load testing (this for scaling). Identify where are the capacity limits. This helps to identify bottleneck and fix them.

Scalability

The ability of a system to accommodate increasing demand by adding or removing resources as needed.

Note: Some system cannot be easily scaled.

Scalability allows a system to adapt to changing usage patterns and handle increased traffic without requiring changes to the application code or system design (system is automatically designed to handle increasing/decreasing resources requirements)

Not every system has traffic that fluctuates based on day or day of the year. Examples: E-commerce websites have Black Friday, School registrations are busy during the enrolment times, tax systems are busy in April

Questions:

Can you expand the capacity of a system very easily by adding more servers? Or will it be a massive undertaking to do that?

Even after adding 10/100 servers and still cannot handle users, then that system is not scalable.

Types of scaling:

Vertical scaling (scaling up/scaling down): This is we have a single server, and we are adding more resources (more memory, number of CPUs) to the server. The issue is there is an upper limit to this (Azure – 96 vCPUs, 384 memory). Also, single server does not improve availability.

Horizontal scaling (scaling out/scaling in): Adding more servers to a system. Here no limits to scaling. However, this adds complexities for load balancing. Note that this will improve availability.

When discuss scalability, it has impact on system cost. Adding more resources to a system adds to cost (in Azure cost will double when number of CPU doubled).

Reducing resources can reduce cost.

Thus, having a scalable system allows for a system to be perfectly sized. In the cloud we can easily add/remove resources and this optimizes the cost by reducing wasted computing resources.

Elasticity

The ability of a system to quickly and easily scale up or down the amount of resources that a system uses in response to changing demand.

This has to involve some sort of automation and often called “autoscaling” in cloud computing. The system monitors some metric (CPU utilization) and determine how busy a system is. It adds resources when it exceeds a limit for being busy and vice versa.

This is more efficient and cost-effective use of resources. It minimizes computing “waste” – resources paid for and not used.

Note: Seld-hosted systems tend to have a large percentage of “over-provisioned” resources for anticipated future growth. Here we have the potential to have a maximum capacity higher than you could afford if you had a static provisioning of resources.

Reliability

The ability of a system to recover from failure.

Azure has several built-in services that you can use to keep an application running after failure has occurred.

There are many different failures that can occur even in a well-designed system:

- Hardware failure
- Network interruptions
- Power failures
- Large-scale regional outage

Why we need fail safe systems?

You have to trust that cloud provider is doing everything to make its platform reliable. This includes transparency during the service issues.

Azure achieves this by:

- Auto-scaling
- Avoid single points of failure (distribute code and application to multiple instances and multiple virtual machine (VM), multiple regions)
- Data backup and replication
- Keep monitoring (health probes and self-healing)

Predictability

The ability to forecast and control the performance and behavior of a system. This also includes the ability to predict the future costs.

Predictability gives you the confident that system will continue to perform at the expected level in the future.

How is it achieved?

- Autoscaling
- Load balancing
- Different instance types, sizes, pricing tiers
- Cost management tools (Azure has a pricing calculator which would help to get an estimate for the cost based on current spending rate. This will also help with budgeting.)
- There are APIs (application programming interface) for cost. (The automation system will be able to get estimates)

Security

Cloud providers (Azure, AWS, Google Cloud) are obviously massive targets for hackers, and so they rightly spend a lot of time, money, and effort on platform security.

Cloud providers go through security audits and compliance certifications.

As the security is a shared responsibility could providers help users with security by providing tools while working on the parts, they have control on.

Why is it needed?

- You want confidence that your cloud provider cannot easily be defeated by hacker and those with malicious intent.

How is it achieved?

- Follow industry standard compliance certifications.
- Microsoft Security Response Center (MSRC). A team is working in a center and monitoring system, suspicious traffic, and so on.
- Always-on DDoS (A DDoS attack is a malicious attempt to disrupt the normal traffic of a targeted server, service or network by overwhelming it with a flood of Internet traffic).
- Azure Policy (more on governance). Minimum policies to set company standards.
- Entra ID (role-based access control)
- Always up-to-date platform services (database management)

- Update management (make sure security patches are up to date)

- Encryption by default

Governance

In simple words, how your organization does business.

The process of defining, implementing, and monitoring a framework of policies that guides an organization's cloud operations.

Why is it needed?

- Your company wants to ensure its policies are followed in the cloud. This includes basic auditing and reporting as well as enforcement. (cost would be a policy)

- The company want to compliant with industry standards such as HIPAA/PCC/GDPR (legal requirements in health industry, not to disclose credit card information, ...)

How is it achieved?

- Azure Policy and Blueprint

- Management groups

- Custom rules (which rules apply to which groups)

- Soft delete (if something is deleted, it stays for some time before being delete forever).

- Azure provides/publishes guide and best practices such as Cloud Adoption Framework

Manageability

Manageability is broken into two parts:

- Management of the cloud (ability to manage applications in the cloud)
- Management in the cloud (ability to manage cloud itself)

Management of the cloud includes all the ways that cloud providers give you to manage applications. This includes:

- Templates
- Automations
- Scaling
- Monitoring and alerts
- Self-healing

Management in the cloud include the features that cloud providers give you to manage your resources. These include:

- Web portal (Azure web portal)
- Command line interface and scripts
- APIs
- PowerShell

Why is it needed?

- How easy it is to work with your applications in the cloud impacts cost, performance, security and other priorities.
- Based on the cloud vendor (due to different interfaces, different ways of managing), you may find it different.

How is it achieved (in MS Azure)?

- Azure portal, CLI, PowerShell, Cloud Shell, RESR APIs, and other programmatic methods.

- Consolidated monitoring with Azure monitoring and alerting system (certain level of traffic, and so on)

- Ability to use ARM templates, Bicep, Terraform, etc.

- Autoscaling of most types of computer resources