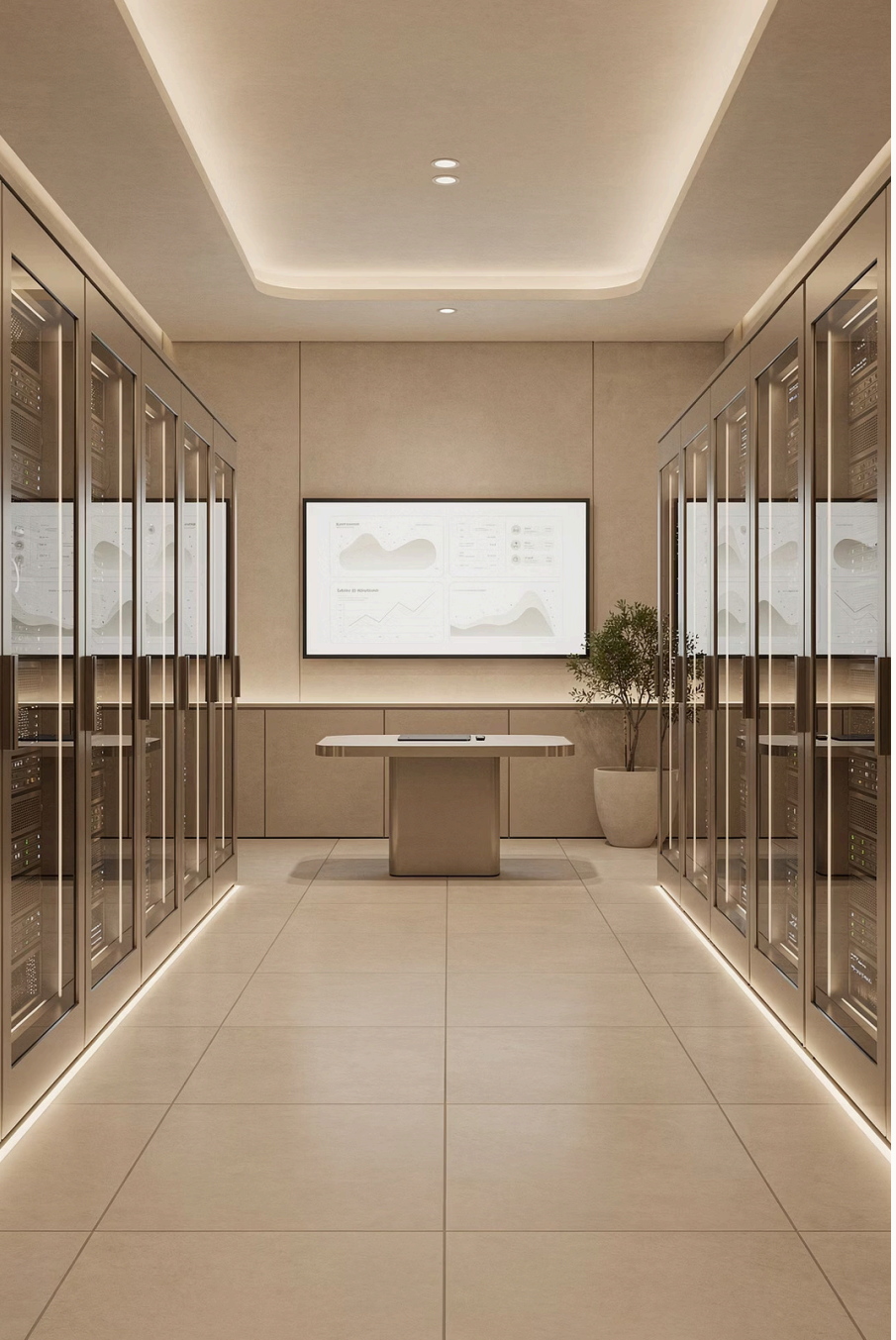


Monitorizare și Observabilitate în Azure

Sesiunea 10 — Azure Fundamentals | Cum știi că tot ce ai construit funcționează corect?

AZURE FUNDAMENTALS SESIUNEA 10

De ce monitorizarea nu este opțională

Imaginați-vă că lansați o aplicație web pentru un client important. Totul merge perfect în ziua lansării. O săptămână merge bine. Două săptămâni, perfect.

Și apoi, într-o joi seara, utilizatorii încep să primească erori. Pagina se încarcă în zece secunde în loc de una. Unele cereri nu primesc niciun răspuns. Clienții trimit emailuri furioase. Telefonul sună.

Ce găsiți când vă conectați în panică

- CPU-ul este la 100% de două ore
- Baza de date are mii de conexiuni deschise
- Un proces a luat-o razna
- Utilizatorii suferă de două ore

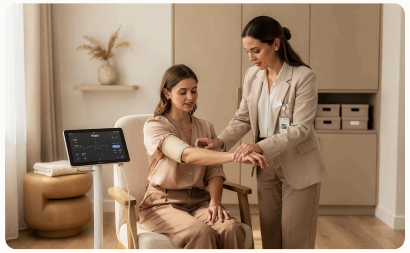
Regula nescrisă a industriei

> Dacă nu monitorizezi, nu operezi. Orice sistem în producție fără monitorizare este un sistem pe care nu îl înțelegi. Și un sistem pe care nu îl înțelegi te va surprinde inevitabil în cel mai prost moment posibil.

Monitorizarea nu previne întotdeauna problema. Dar vă anunță în momentul în care problema apare — nu după ce clienții v-au inundat cu reclamații.

Ce este monitorizarea? Modelul mental complet

Înainte să deschidem Azure Portal, hai să construim un model mental clar. Monitorizarea are trei componente principale, fiecare cu un scop distinct. Împreună formează un sistem complet de observabilitate.

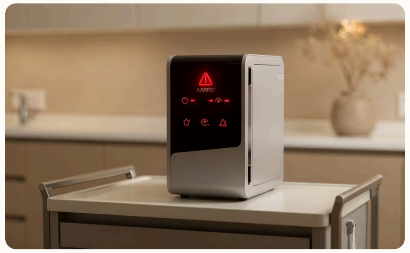


Metrici — Semnele vitale

Valorile numerice măsurate în timp real: utilizare CPU, memorie, rețea, disc. Exact cum medicul măsoară pulsul și tensiunea — te spun **ce se întâmplă acum** cu sistemul.

Log-uri — Istoricul medical

Înregistrările complete ale tot ce s-a întâmplat: cine s-a conectat, ce operațiuni au rulat, ce erori au apărut. Log-urile îți spun **de ce se întâmplă** ceea ce observi în metrici.



Alerte — Alarma automată

Echivalentul dispozitivului medical care bipăie când pulsul depășește pragul critic. Nu trebuie să stai constant cu ochii pe monitor — sistemul **te anunță automat** când trebuie să acționezi.

Mesajul cheie: Metricile îți arată **ce** se întâmplă. Log-urile îți arată **de ce** se întâmplă. Alertele te anunță **că trebuie să acționezi**. Împreună, formează fundamentul oricărui sistem observabil.

Azure Monitor — Centrul de comandă

În Azure, toate capabilitățile de monitorizare sunt centralizate în serviciul numit **Azure Monitor**. Este platforma unificată care colectează, stochează, analizează și acționează pe baza datelor din toate resursele voastre: VM-uri, web apps, baze de date, rețele, storage.

Gândiți-vă la Azure Monitor ca la centrul de comandă al unui aeroport. Controlorii de trafic aerian nu se urcă în fiecare avion să vadă cum merge. Ei au ecrane care agregă informații de la sute de avioane simultan. Văd tabloul complet și intervin când ceva iese din parametri.

Metrics

Date numerice în timp real despre comportamentul resurselor: CPU, memorie, rețea, disc.

Activity Log

Jurnalul operațiunilor: cine a creat, modificat sau șters resurse în subscripția Azure.

Log Analytics

Motorul de căutare și analiză pentru volume mari de log-uri, cu query-uri KQL.

Alerts

Sistemul de notificare automat: email, SMS, webhook sau declanșarea de acțiuni.

App Insights

Monitorizarea dedicată aplicațiilor: performanță end-to-end, erori, comportament utilizatori.

Creăm resursele pentru exercițiu

Înainte să explorăm monitorizarea, avem nevoie de ceva de monitorizat. Urmați pașii de mai jos în Azure Portal.

1

Creați Resource Group

Numiți-l `rg-s10-[prenumele vostru]`. Selectați regiunea cea mai apropiată de voi.

2

Creați o mașină virtuală

Windows sau Linux — alegeți ce preferați. Folosiți cel mai mic size din **seria B** (ex. B1s sau B2s).

3

Configurați accesul

În Network Security Group, activați **RDP** (Windows) sau **SSH** (Linux) cu sursa setată la `My IP Address`.

4

Așteptați deployul

Cât timp rulează deploymentul, continuăm cu contextul teoretic. Verificați notificarea din portal când resursa este gata.

Cost reminder: Seria B este cea mai economică. Nu uitați să ștergeți resursele la finalul sesiunii pentru a evita costuri inutile.

Metrici: Semnele vitale ale serverului tău

Când VM-ul este gata: navigați la el în Azure Portal → secțiunea **Monitoring** → **Metrics**. Adăugați pe rând: `Percentage CPU`, `Network In`, și `Disk Read Bytes`.

Cum interpretăm CPU Percentage

0–5% → Server inactiv

Poate fi normal (serverul așteaptă cereri) sau o problemă (aplicația a căzut și nu mai procesează nimic).

50–70% → Încărcare normală

Serverul procesează activ. Semnul unui sistem sănătos sub trafic real.

100% susținut → Problemă critică

Serverul nu mai poate procesa cereri noi. Aplicațiile răspund lent sau deloc. Utilizatorii suferă.

Întrebarea capcană

Dacă CPU-ul serverului este constant la **2%** — este bine sau rău?

Intuitiv pare bine. Dar gândiți-vă mai adânc:

- Plătiți pentru 4 vCPU și folosiți 2% → risipă de bani, dimensionare greșită
- O aplicație care ar trebui să proceseze date stă la 2% → poate a căzut și nu mai funcționează

Concluzie: O metrică nu este bună sau rea în mod absolut. Este bună sau rea în contextul a ceea ce ar trebui să facă sistemul. Monitorizarea eficientă înseamnă să cunoști **baseline-ul normal** și să detectezi devierile.

Granularitate temporală și Time Ranges

Un aspect esențial pe care îl veți folosi constant în practică este **granularitatea temporală** a metricilor. În interfața Metrics puteți schimba intervalul de timp și frecvența valorilor.

1

Investigarea unui incident

Zoom pe intervalul afectat (ex. 2:00–4:00 AM) cu granularitate de **1 minut**. Vedeți exact momentul în care a explodat CPU-ul, când a reacționat rețeaua, când s-a umplut discul.

2

Analiza tendințelor

Intervale mari (săptămâni, luni) cu granularitate de ore sau zile. Identificați creșterea treptată a utilizării discului sau tendința traficului săptămână de săptămână.

3

Decizia de scaling

Comparați perioadele de vârf vs. perioade normale. Dacă peak-ul zilnic atinge constant 80% CPU, este momentul să discutați despre **scale up** sau **scale out**.

Diferența cheie: Granularitatea mare (1 minut) → pentru diagnostic rapid. Granularitatea mică (1 oră/zi) → pentru planificare și tendințe pe termen lung. Știind când să folosești fiecare te transformă dintr-un operator reactiv într-un arhitect proactiv.

Log-uri: Camera de supraveghere a infrastructurii

Navigați la VM → **Monitoring** → **Activity Log**. Veți vedea o listă cronologică a tuturor operațiunilor care au afectat resursa: când a fost creată, cine a creat-o, când a pornit, dacă cineva a modificat configurația sau regulile de firewall.

Gândiți-vă la Activity Log ca la o cameră de supraveghere la intrarea în firmă: înregistrează cine intră, cine iese, la ce oră, cu ce acreditări. Nu ce face persoana în birou — ci intrările și ieșirile.

Scenariul 1: Securitate

Dacă cineva a șters accidental sau intenționat o resursă importantă, Activity Log-ul vă spune exact **cine** a făcut-o și **când**. Dacă o configurație a fost modificată și a cauzat o problemă, Activity Log-ul arată **ce** a fost schimbat și de cine.

Este prima linie de răspuns în orice investigație de securitate.

Scenariul 2: Compliance și Auditare

Organizațiile din domeniile financiar, medical și guvernamental au cerințe stricte de auditare. Trebuie să poată demonstra:

- Cine a avut acces la ce resurse
- Ce acțiuni a efectuat fiecare utilizator
- Când au avut loc modificările de configurație
- Cine a aprobat schimbările critice

Activity Log-ul este răspunsul Azure la aceste cerințe de compliance.

Log Analytics și Kusto Query Language (KQL)

Activity Log-ul arată operațiunile la nivelul resursei. Dar pentru a vedea ce se întâmplă *în interiorul* ei — log-urile sistemului de operare, ale aplicațiilor, ale proceselor interne — aveți nevoie de **Log Analytics**.

Ce este Log Analytics

Un serviciu Azure care colectează, stochează și permite căutarea în volume mari de log-uri din multiple surse simultan: VM-uri, web apps, baze de date, firewall-uri, aproape orice serviciu Azure.

Pentru a trimite log-urile unui VM către Log Analytics, instalați **Azure Monitor Agent** pe mașina respectivă. Agentul colectează log-urile din OS și le trimite centralizat.

> **Notă practică:** Astăzi nu configurăm Log Analytics complet — implică pași suplimentari și costuri. Dar conceptul este esențial pentru practică reală.

Exemplu de query KQL

Odată ce log-urile sunt în Log Analytics, le interogați cu **KQL — Kusto Query Language**: un limbaj declarativ similar ca filozofie cu SQL, optimizat pentru analiză de log-uri în timp real pe volume mari.

Toate erorile din ultimele 24 de ore, ordonate cronologic descrescător:

```
Event
| where EventLevelName == "Error"
| where TimeGenerated > ago(24h)
| order by TimeGenerated desc
```

KQL este o abilitate valoroasă pe care o veți dezvolta pe măsură ce avansați în cariera de cloud. Există resurse dedicate și o documentație excelentă de la Microsoft.

Creăm prima alertă — pas cu pas

Acum vine partea practică cea mai importantă: crearea unei **alerte automate**. O alertă în Azure Monitor este o regulă care spune: *dacă o metrică depășește un prag, notifică-mă.*

1

Condiția

Ce metrică monitorizăm, ce prag o declanșează, pe ce interval de timp se evaluează. Ex: CPU > 5% timp de 5 minute.

2

Grupul de acțiuni

Ce se întâmplă când condiția este îndeplinită: email, SMS, webhook, Logic App sau Azure Function.

3

Regula de alertă

Combinația dintre condiție și acțiune, cu un nume, o descriere și o **severitate** (0=Critic → 4=Verbose).

Pași în Azure Portal

1. VM → Monitoring → **Alerts** → Create → Alert rule
2. Condition: `Percentage CPU` → greater than **5%**, granularitate 1 min, evaluare 5 min
3. Actions: Create action group → Notifications → Email → adresa voastră
4. Details: Nume `Alert-CPU-High-[prenume]`, Severity: **2 — Warning**
5. Dați **Create**. Alerta este activă imediat.

De ce 5% și nu 80%?

Folosim 5% ca prag în sesiune pentru a vedea alerta declanșându-se în timp real. În **producție reală**, veți configura praguri realiste:

- CPU > 80% pentru mai mult de 10 minute
- Disc < 10% spațiu liber
- Erori HTTP 5xx > 1% din cereri
- Latență medie SQL > 1 secundă

Fiecare sistem are metricile lui critice. Rolul arhitectului este să le identifice și să configureze alertele **înainte** ca o problemă să afecteze utilizatorii.

Monitorizare la Web App și SQL Database

Un principiu important: în Azure, monitorizarea **nu este un feature separat** pe care trebuie să îl activezi. Este integrată în fiecare serviciu din momentul creării. Fiecare resursă expune automat un set de metrici relevante pentru tipul ei.

Azure Web App — Metrici critice

Requests

Numărul de cereri HTTP pe secundă. Indicatorul principal al traficului aplicației.

Response Time

Dacă normalul este 200ms și brusc ajunge la 2s — ceva s-a schimbat. Investigați imediat.

HTTP 5xx

Erori de server. O creștere bruscă înseamnă problemă internă în aplicație.

HTTP 4xx

Erori de client (404 etc.). O creștere neașteptată poate indica scanning sau URL-uri rupte.

CPU Time

Util pentru detectarea memory leak-urilor sau a buclelor infinite în cod.

Azure SQL Database — Metrici critice

DTU Percentage

Constant la 90–100%? Baza de date este subprovisionată și trebuie upgrade-uită.

CPU Percentage

Utilizarea procesorului la nivelul motorului SQL. Indicator cheie de performanță.

Data IO Percentage

Utilizarea operațiunilor de I/O pe disc. Gâtul de sticlă cel mai comun în SQL.

Sessions Count

O creștere neașteptată poate indica un **connection leak** — conexiuni deschise dar niciodată închise.

Deadlocks

Blocaje între tranzacții. Indică probleme de design în logica de acces concurent la date.

Pattern comun: Fiecare serviciu are metrici specifice pentru natura lui, dar toate urmează aceeași filozofie: **utilizare de resurse, performanță, erori**. Odată ce înțelegeți structura pentru un tip de resursă, o aplicați rapid la orice alt serviciu Azure.

Observabilitate: Dincolo de monitorizare

Monitorizarea și observabilitatea sunt adesea confundate, dar sunt concepte diferite cu implicații practice distincte.

Monitorizare

Urmărești **ce știi deja că contează**. Configurezi alerte pentru CPU, memorie, disc. Știi dinainte ce vrei să urmărești și configurezi instrumentele pentru acel scop specific.

Este un demers **reactiv** — răspunzi la scenarii pe care le-ai anticipat.

Observabilitate

Sistemul tău este proiectat astfel încât să poți înțelege **orice comportament intern** prin examinarea ieșirilor externe. Poți pune întrebări la care nu te-ai gândit când l-ai construit.

De ce această cerere specifică a durat 3 secunde? De ce eroarea apare doar pentru utilizatorii din Germania?

Este un demers **proactiv și exploratoriu**.

Cei trei piloni ai observabilității

Metrici

Valori numerice în timp real. Semnalele vitale ale sistemului. Răspund la întrebarea: *Ce se întâmplă acum?*

Log-uri

Înregistrări istorice detaliate ale evenimentelor. Răspund la întrebarea: *De ce s-a întâmplat?*

Trace-uri (Distributed Tracing)

Parcursul complet al unei cereri prin toate componentele unui sistem distribuit. Răspund la: *Unde s-a pierdut timpul?*

Un distributed trace înregistrează fiecare hop al unei cereri: load balancer → API gateway → autentificare → inventar → plăți → email → baza de date. Vedeți cât timp a durat fiecare serviciu și unde a apărut eroarea. Azure oferă **Application Insights** pentru aceasta, cu distributed tracing și end-to-end transaction monitoring.

Application Insights — Monitorizarea aplicațiilor

Dacă Azure Monitor este centrul de comandă al infrastructurii, **Application Insights** este instrumentul dedicat monitorizării aplicațiilor în sine — nu a serverelor pe care rulează, ci a codului, a cererilor și a experienței utilizatorilor.

Ce face Application Insights concret

- Urmărește fiecare cerere HTTP: URL, metodă, cod răspuns, durată
- Detectează automat dependențele externe: SQL, API-uri, blob, cozi
- Înregistrează toate excepțiile și erorile negestionate din cod
- Urmărește page view-uri și sesiuni ale utilizatorilor
- Detectează anomalii automat cu algoritmi de Machine Learning

Smart Detection

Un feature excepțional de util: Application Insights analizează **pattern-urile normale** ale aplicației voastre și vă alertează automat când detectează o abatere:

- Creștere neașteptată a ratei de eșec
- Degradare a performanței față de baseline
- Anomalie în distribuția timpilor de răspuns

Nu trebuie să configurați manual praguri pentru toate scenariile posibile. **Sistemul învață ce este normal și vă alertează când normalul se schimbă.**

Application Map

Application Insights introduce **Application Map** — o vizualizare grafică a tuturor componentelor aplicației și a dependențelor dintre ele. Vedeți vizual care componente comunică cu care, câte cereri trec prin fiecare, care au rate de eroare mai mari, care introduc mai multă latență. Este o hartă vie a arhitecturii voastre, actualizată în timp real. Integrarea cu un Azure Web App se face în câteva clickuri din portal, fără modificări în codul sursă. SDK-uri disponibile pentru: .NET, Java, Node.js, Python, PHP.

Costuri și bune practici în monitorizare

Un subiect pe care mulți îl ignoră: **monitorizarea costă bani**. Metricile standard ale resurselor Azure sunt gratuite. Dar Log Analytics are costuri bazate pe cantitatea de date ingestionate și pe perioada de retenție. La scară mare, costul monitorizării poate fi semnificativ.

Ingestie selectivă

Nu trimiteți toate log-urile posibile către Log Analytics. Selectați categoriile relevante: erori, avertismente, evenimente de securitate. Log-urile de debug și verbose aparțin development-ului, nu producției.

Retenție adecvată

Configurați cât timp păstrați datele. Datele mai vechi de 90 de zile pot fi arhivate în Azure Storage la un cost considerabil mai mic decât în Log Analytics.

Sampling în App Insights

Dacă aplicația primește mii de cereri pe secundă, un eșantion de 10–20% oferă o imagine statistică precisă la un cost mult mai mic decât înregistrarea 100% a telemetriei.

> Monitorizarea nu este un cost. Este o investiție în vizibilitate și reziliență. Costul investigării unui incident major fără date de monitorizare — orele de muncă, downtime-ul, impactul asupra clienților — depășește de obicei cu mult costul lunar al unui setup decent de monitorizare.

Scenarii reale de monitorizare

Monitorizarea nu elimină problemele. Dar vă permite să le detectați rapid, să înțelegeți cauza și să interveniți înainte ca impactul să devină major. Iată trei scenarii care se repetă în orice companie care operează sisteme în cloud.

Scenariul 1: Degradarea performanței

Situație: Vineri după-amiază. O alertă: Response Time a depășit 2s față de normalul de 200ms. **Investigație:** Application Insights arată că degradarea a început la 15:20. Dependency map: query-urile SQL au crescut de la 20ms la 150ms. Metrici SQL: DTU Percentage la 98% de o oră. **Cauza:** Traficul a crescut organic și baza de date nu mai face față. **Soluție:** Scale up al planului SQL Database. **Fără monitorizare:** Aflați de la utilizatori că e lentă, fără să știți de ce.

Scenariul 2: Incident de securitate

Situație: Luni dimineață, trafic neobișnuit de rețea pe un server. **Investigație:** Log Analytics → evenimente de autentificare eșuată din ultimele 24h → mii de încercări dintr-un singur IP extern între 1:00–3:00 AM. **Cauza:** Atac brute force asupra parolei serverului. **Acțiune imediată:** Blocați IP-ul în NSG, activați autentificarea cu doi factori. **Fără Activity Log:** Nu ați fi știut că atacul a avut loc.

Scenariul 3: Capacitate disc

Situație: Alertă la 8:00 AM: discul VM-ului de procesare batch este la 95% capacitate. **Investigație:** Log-urile arată că un job nocturn a generat mult mai multe fișiere temporare decât de obicei din cauza unui volum mai mare de date procesate. **Soluție:** Extindeți discul din portal în câteva clickuri fără restart, și configurați o politică de curățare automată. **Fără alertă:** Discul s-ar fi umplut, procesele s-ar fi oprit, ore de procesare pierdute.

Cleanup — Ștergerea resurselor

La finalul fiecărei sesiuni, ștergem resursele create pentru a evita costuri inutile. Astăzi avem o operație specială înainte de cleanup.

01

Ștergeți regula de alertă

Navigați la **Azure Monitor** (din bara de căutare principală) sau la VM → Alerts. Găsiți regula `Alert-CPU-High-[prenume]` și ștergeți-o explicit. Bună practică: verificați întotdeauna alertele înainte de a șterge resurse.

02

Ștergeți Resource Group-ul

Navigați la **Resource Groups** → selectați `rg-s10-[prenumele vostru]` → Delete resource group. Confirmați cu numele resource group-ului.

03

Verificați că nu au rămas resurse

Mergeți la **All Resources** și filtrați după sesiunea de astăzi. Asigurați-vă că nu a rămas nicio resursă activă și facturabilă.

Notă: Ștergând Resource Group-ul, toate resursele din el sunt șterse automat, inclusiv regulile de alertă asociate. Dar verificarea explicită a alertelor este o bună practică profesională.

Tema pentru acasă

Tema pentru sesiunea de astăzi are două puncte principale și un bonus opțional pentru cei curioși.

Punctul 1 — Explicați conceptele

Explicați diferența dintre o **metrică, un log și o alertă**. Folosiți o analogie proprie — *diferită* de analogia medicală sau cea a avionului din sesiunea de astăzi. Imaginați-vă că explicați unui manager non-tehnic de ce are nevoie de monitorizare pentru aplicația companiei.

Punctul 2 — Impact de business

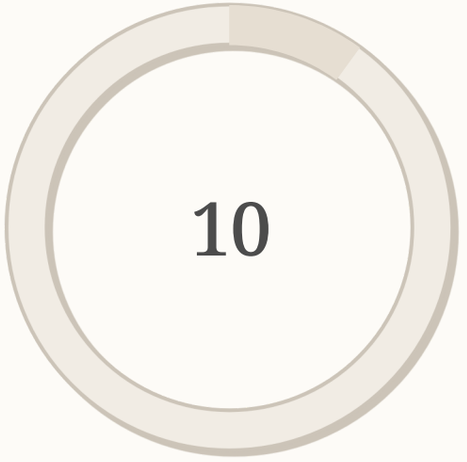
De ce este monitorizarea **critică** pentru o aplicație live, în producție? Dați cel puțin **două scenarii concrete** în care lipsa monitorizării ar putea cauza un impact real de business: pierderi financiare, clienți nemulțumiți, probleme de securitate.

Bonus — Azure Service Health

Cercetați ce este **Azure Service Health**. Este diferit de monitorizarea resurselor voastre — este monitorizarea infrastructurii Microsoft în sine. Cum ați folosi Service Health în contextul unui incident major la nivel de regiune Azure? Ce acțiuni ar trebui să luați?

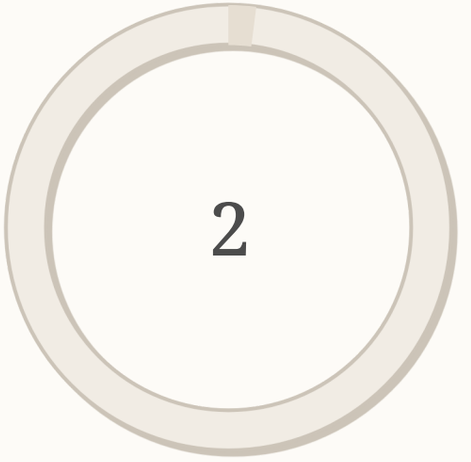
Privire spre înainte — S11 și S12

Opriți-vă un moment și priviți panorama completă a ceea ce ați construit în ultimele zece sesiuni. Ați pornit de la zero — de la ce este un IP, ce este un port. Azi, tabloul arată altfel.



Sesiuni parcurse

De la concepte de rețea la monitorizare și observabilitate



Sesiuni rămase

Securitate & Best Practices, apoi Mini-Proiect Integrativ Final

Sesiunea 11 — Securitate și Best Practices

Mergem mai în profunzime pe securitate: identitate, acces, criptare, Azure Security Center, RBAC. Veți înțelege cum se construiește o **postură de securitate coerentă** — nu o colecție de setări izolate.

Sesiunea 12 — Proiect Final și Cariera în Cloud

Un **mini-proiect integrativ** care pune împreună tot ce ați învățat. Un recap complet al cursului. O conversație despre carieră: certificări recomandate, cum arată o zi de muncă pentru un cloud architect, oportunități pe piață.

> Nu știți doar să folosiți servicii Azure. Știți să gândiți arhitectural. Știți să puneți întrebările corecte: de ce folosim asta și nu altceva, ce se întâmplă dacă ceva se strică, cât costă, cine are acces, cum știu că funcționează. Aceasta este mentalitatea unui profesionist cloud. Vă mulțumesc pentru participare, curiozitate și persistență. Ne vedem la sesiunea unsprezece.