# Projet Cloud



# Rapport de projet

Zeqiri Amir – CIN2B ETML, Vennes – A22 24 Périodes Maître : M. Sonney

# Table des matières

1	INTRO	DUCTION	. 4
	1.1 Pri	ÉSENTATION DU PROJET	. 4
	1.2 OB	SJECTIF	. 4
2	CHOI	( DES ACTIVITÉS	. 4
	2.1 SÉL	ECTION DES ACTIVITÉS POUR TOTALISER 15 POINTS	. 4
	2.1.1	Installer Azure PowerShell (2 points)	
	2.1.2	Créer un script qui liste toutes les VMs de la souscription (2 points)	
	2.1.3	Créer un budget pour un resource-group (2 points)	
	2.1.4	Créer une alerte de coût pour un resource-group (2 points)	
	2.1.5	Créer un script Powershell pour lister les ressources d'un resource-group	(2
	points		
	2.1.6	Créer une machine virtuelle Linux et s'y connecter en SSH (2 points)	. 4
	2.1.7	Sauvegarder le template ARM d'une ressource (2 points)	
	2.1.8	Déployer une VM Linux et y installer un serveur web et un index.html	
		nisé (5 points)	
		DURQUOI CES ACTIVITÉS	
	2.2.1	Activité 1	
	2.2.2	Activité 2	
	2.2.3	Activité 3	
	2.2.4	Activité 4	
	2.2.5	Activité 5	
	2.2.6 2.2.7	Activité 6	
	2.2.7	Activité 7 Activité 8	
3		ITÉS (19 POINTS)	
		·	
		TALLER AZURE POWERSHELL	
		ÉER UN SCRIPT QUI LISTE TOUTES LES VMS DE LA SOUSCRIPTION	
		ÉER UN BUDGET POUR UN RESOURCE-GROUP	
		ÉER UNE ALERTE DE COÛT POUR UN RESOURCE-GROUP	
		ÉER UN SCRIPT POWERSHELL POUR LISTER LES RESSOURCES D'UN RESSOURCE-GROUP	
		ÉER UNE MACHINE VIRTUELLE LINUX ET S'Y CONNECTER EN SSH	
		UVEGARDER LE TEMPLATE ARM D'UNE RESSOURCE	
		PLOYER UNE VM LINUX ET Y INSTALLER UN SERVEUR WEB ET UN INDEX.HTML	
4	SYNTH	ÈSE DES COMPÉTENCES DÉVELOPPÉES	15
	4.1 Co	DMPÉTENC ES TECHNIQUES	15
	4.1.1	Gestion des ressources Azure	15
	4.1.2	Scripts PowerShell	
	4.1.3	Création et gestion de machines virtuelles	
	4.1.4	Automatisation et templates ARM	
		DMPÉTENC ES TRANSVERSALES	
	4.2.1	La gestion du temps	
	4.2.2	La résolution de problèmes	
	4.2.3	L'autonomie	16
5	PROBI	ÈMES RENCONTRÉS	16

	5.1	Problème d'installation de logiciels sur la VM	16
6	CC	ONCLUSION	16
	6.1 6.2	RETOUR SUR L'EXPÉRIENCE	
7	AN	INEXES	17
	7.1 7.1 7.2 7.2 7.2	SCRIPTS1 ListerRessources.ps12 ListerVMs.ps1 TEMPLATES ARM1 ConnectLinuxSSH.zip2 ExportedTemplate-rg-amizeq.zip3 ServeurWebLinux.zip	17 17 17 17
8	RÉI	FÉRENCES	17
	8.1 8.2	CONVENTION DE NOMMAGE	

## 1 INTRODUCTION

## 1.1 Présentation du projet

Catalogue d'activité à réaliser dans le Cloud

## 1.2 Objectif

Découvrir les possibilités du Cloud à travers différentes activités à réaliser sur la plateforme Azure.

## 2 CHOIX DES ACTIVITÉS

## 2.1 Sélection des activités pour totaliser 15 points

- 2.1.1 Installer Azure PowerShell (2 points)
- 2.1.2 Créer un script qui liste toutes les VMs de la souscription (2 points)
- 2.1.3 Créer un budget pour un resource-group (2 points)
- 2.1.4 Créer une alerte de coût pour un resource-group (2 points)
- 2.1.5 Créer un script Powershell pour lister les ressources d'un resource-group (2 points)
- 2.1.6 Créer une machine virtuelle Linux et s'y connecter en SSH (2 points)
- 2.1.7 Sauvegarder le template ARM d'une ressource (2 points)
- 2.1.8 Déployer une VM Linux et y installer un serveur web et un index.html customisé (5 points)

# 2.2 Pourquoi ces activités

Les deux premières activités sont obligatoires et me permettent de débuter le projet.

### 2.2.1 Activité 1

L'étape de l'installation de Azure PowerShell est primordiale car certaines des prochaines activités se feront avec.

## 2.2.2 Activité 2

La deuxième activité me permettra de me familiariser avec la création de scripts. En commençant par un script simple, je me mets dans le bain, ce qui facilitera l'écriture de scripts plus complexes dans les activités suivantes.

#### 2.2.3 Activité 3

La troisième activité me permettra de voir comment on peut gérer l'argent sur Azure. En mettant un budget, je pourrai surveiller combien on dépense pour éviter de dépasser.

#### 2.2.4 Activité 4

La quatrième activité se joint à la troisième. L'alerte de coût va avec le budget. Elle sert à m'avertir si on dépense trop ou si on approche du budget. Comme pour le projet nous avons une limite de 5CHF, il est important de fixer des rappels.

#### 2.2.5 Activité 5

La cinquième activité m'aidera à mieux comprendre comment récupérer des informations importantes sur les ressources. En apprenant à utiliser cette commande, je me prépare pour d'autres tâches liées à la gestion des ressources dans Azure, ce qui est utile pour la suite du projet.

#### 2.2.6 Activité 6

La sixième activité me permettra de comprendre comment créer une machine virtuelle Linux et m'y connecter en SSH. Cette étape est importante car elle m'initie à l'utilisation des machines virtuelles sur Azure. En apprenant à me connecter en SSH, je pourrai interagir avec la VM et effectuer des configurations ou installations nécessaires pour d'autres activités du projet.

#### 2.2.7 Activité 7

La septième activité m'apprendra à garder une copie de la configuration d'une ressource. Comme ça, je pourrai la recréer plus tard facilement si j'en aurai besoin. Et c'est demandé dans le cahier des charges pour chaque ressource afin de prouver le travail à l'enseignant.

#### 2.2.8 Activité 8

La huitième activité est plus complexe et demande de déployer une VM Linux, d'y installer un serveur web, et d'y personnaliser un fichier index.html. Elle me permettra de mettre en pratique plusieurs compétences acquises, comme la configuration de serveurs et la personnalisation des fichiers. Cette activité est essentielle pour montrer que je sais déployer une infrastructure fonctionnelle sur Azure et créer un environnement opérationnel pour un usage concret.

# 3 ACTIVITÉS (19 POINTS)

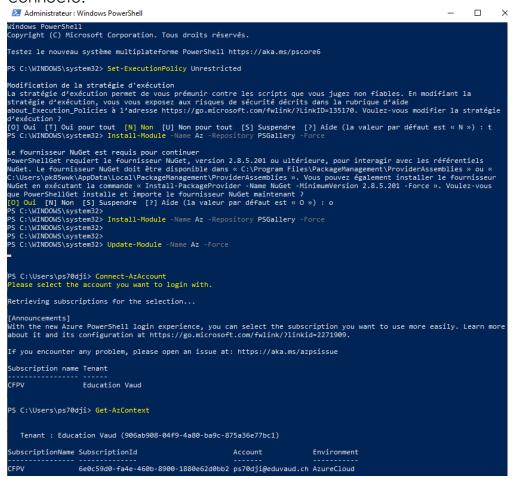
#### 3.1 Installer Azure PowerShell

Cette activité vise simplement à installer Azure PowerShell.

Pour ce faire, je me suis rendu sur <u>Microsoft Learn</u> et j'ai suivi la marche à suivre.

En premier lieu, j'ai executé cette commande Set-ExecutionPolicy Unrestricted. Ça permet de lancer tous les scripts sans restriction, mais ça réduit la sécurité.

Ensuite j'ai installé le module 'Az' et pour finir cette activité, je me suis connecté.



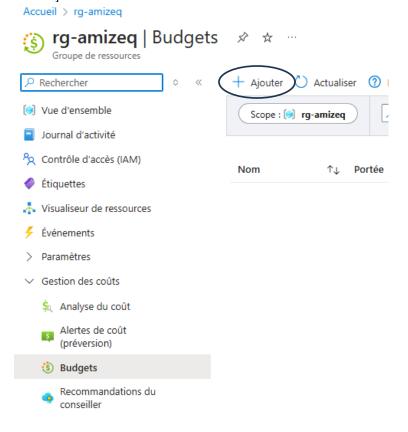
# 3.2 Créer un script qui liste toutes les VMs de la souscription

Pour faire ce script, j'ai récupéré toutes les machines virtuelles que j'ai stocké dans une variable. Puis j'ai listé toutes les machines stockées en affichant leur nom.

Pour lancer le script, je me positionne dans le dossier ou se trouver le script et je l'exécute en écrivant le nom du script avec l'extension .ps1 Le résultat montre bien toutes les VMs de la souscription.

## 3.3 Créer un budget pour un resource-group

Cette activité a pour but de créer un budget pour mon groupe de ressource nommée 'rg-amizeq'. Pour ce faire, je me suis rendu sur le groupe de ressource, dans 'gestion des coûts' puis dans 'Budgets' et j'ai appuyé sur 'Ajouter'.



Me voici sur la création d'un budget.

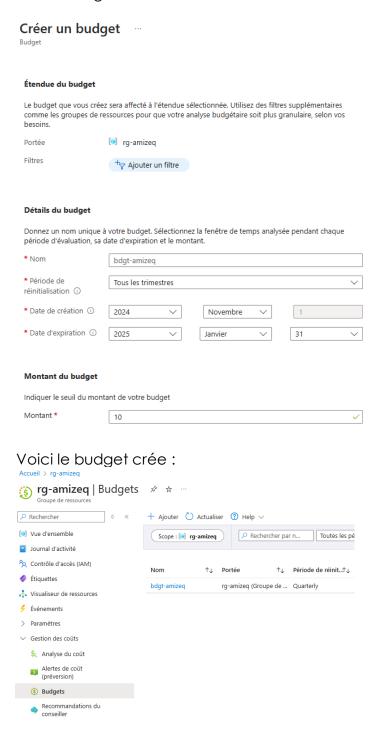
Je dois choisir un nom qui correspond à ce que je crée, comme demandé dans le cahier des charges.

Pour la période de réinitialisation, j'ai mis tous les trimestres car le projet dure un trimestre.

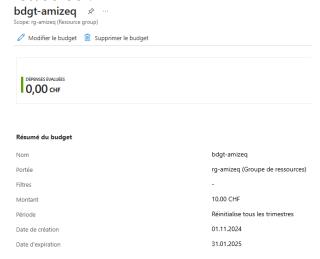
Comme le projet a débuté en novembre, j'ai choisi ce mois-ci pour la date de création.

Concernant la date d'expiration, j'ai choisi le 31 janvier car le projet se termine en ce mois.

Nous avons un budget de 5CHF pour les activités à faire, comme je ne sais pas si je peux respecter ce budget, j'ai mis 10CHF pour le seuil du montant de ce budget.



Et voici les dépenses évaluées avec le résumé de mon budget pour ma ressource :

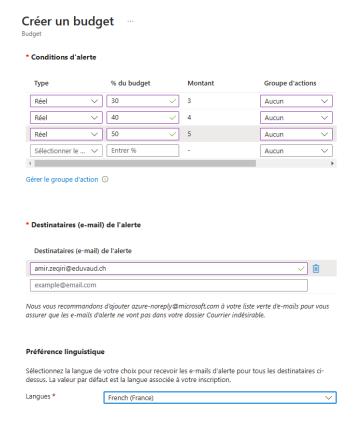


## 3.4 Créer une alerte de coût pour un resource-group

Cette activité se joint à la précédente, elle consiste à créer une alerte de coût pour le budget d'un groupe de ressource. Pour ce faire, je me suis rendu sur le groupe de ressource, dans 'gestion des coûts' puis dans 'Budgets', dans mon budget 'bdgt-amizeq', j'ai cliqué sur 'Modifier le budget' et pour finir je me suis rendu dans le point 2 'Définir des alertes'

Me voici sur la création d'alertes.

J'ai fixé des alertes à 30, 40 et 50 pourcents du budget ce qui signifie que à 3, 4 et 5 CHF je vais recevoir des alertes sur mon mail personnel dans la langue française. J'ai fixé ces alertes sur le budget réel et non de prévision.



Alertes budgétaires							
Conditions d'alerte	Туре	$\uparrow_{\downarrow}$	% du budget	$\uparrow_{\downarrow}$	Montant	$\uparrow_{\downarrow}$	Groupe d'actions
	Actual		30%		3 CHF		Aucun
	Actual		40%		4 CHF		Aucun
	Actual		50%		5 CHF		Aucun
Destinataires (e-mail) de l'alerte	amir.zeqiri@eduvaud.ch						
Préférence linguistique	French (France)						

# 3.5 Créer un script Powershell pour lister les ressources d'un ressource-group

Pour faire ce script, j'ai stocké le nom de mon groupe de ressource dans une variable. Puis j'ai récupéré toutes les informations de mon groupe de ressource en utilisant la variable que j'ai créé, ce qui affiche les ressources de mon groupe.

Pour lancer le script, je me positionne dans le dossier ou se trouver le script et je l'exécute en écrivant le nom du script avec l'extension .ps1 Le résultat montre bien toutes les ressources de mon groupe de ressource.

## 3.6 Créer une machine virtuelle Linux et s'y connecter en SSH

Pour faire cette activité, je suis allé dans la création de machines virtuelles dans le portail Azure. J'ai créé ma VM Linux Ubuntu et voici la configuration.

De base	
Abonnement	CFPV
Groupe de ressources	rg-amizeq
Nom de la machine virtuelle	vm-amizeq-01
Région	Switzerland North
Options de disponibilité	Zone de disponibilité
Options de zone	Zone autosélectionnée
Zone de disponibilité	1
Type de sécurité	Standard
Image	Ubuntu Minimal 22.04 LTS - Génération2
Architecture de machine virtuelle	х64
Taille	Standard B1s (1 processeur virtuel, 1 Gio de mémoire)
Activer la mise en veille prolongée	Non
Type d'authentification	Mot de passe
Nom d'utilisateur	amizeq
Ports d'entrée publics	SSH
Spot Azure	Non

#### Disques

Taille du disque du système d'exploitation Image par défaut Type de disque de système d'exploitation HDD Standard LRS

Utiliser des disques managés Oui
Supprimer le disque de système
d'exploitation avec la machine virtuelle

Disque de système d'exploitation

éphémère

Non

#### Mise en réseau

Réseau virtuel (nouveau) vm-amizeq-01-vnet
Sous-réseau (nouveau) default (10.5.0.0/24)
Adresse IP publique (nouveau) vmamizeq01ip814

Mise en réseau accélérée Désactivé
Placer cette machine virtuelle derrière une solution d'équilibrage de charge

existante?

Supprimer l'adresse IP publique et la carte réseau lors de la suppression de la

machine virtuelle

Désactivé

#### Administration

Microsoft Defender pour le cloud De base (gratuit)
Identité managée affectée par le système Désactivé
Connectez-vous avec Microsoft Entra ID Désactivé
Arrêt automatique Désactivé
Activer hotpatch Désactivé

Options d'orchestration de patch Valeur par défaut de l'image

#### Supervision

Alertes Désactivé
Diagnostics de démarrage Activé
Activer le diagnostic du système d'exploitation invité

a exploitation invite

Activer le monitoring de l'intégrité des

applications

Désactivé

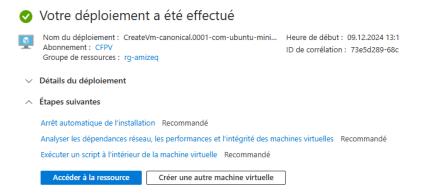
#### Paramètres avancés

Extensions Aucun
Applications de machine virtuelle Aucun
Cloud init Non
Données utilisateur Non
Type de contrôleur de disque SCSI
Groupe de placement de proximité Aucun
Groupe de réservations de capacité Aucun

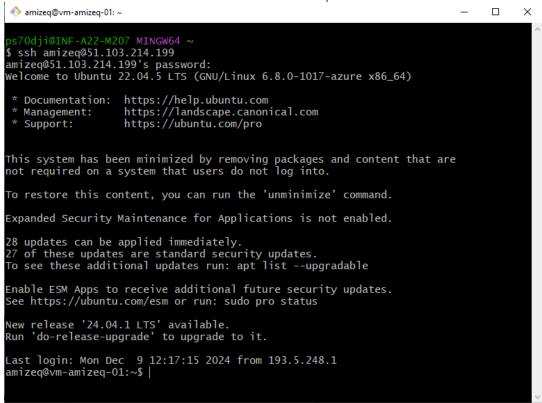
Il est important de sélectionner le disque HDD et non SSD afin d'économiser les couts.

Également important d'activer le port SSH.

Voici la ressource crée:

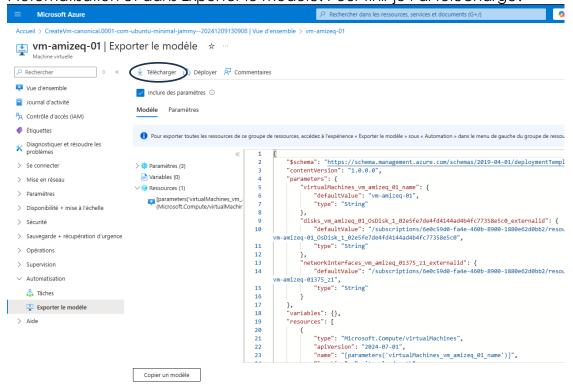


Pour se connecter à la machine, je récupère l'adresse IP publique et entre cette commande dans Git Bash : ssh amizeq@51.103.214.199



## 3.7 Sauvegarder le template ARM d'une ressource

Pour faire cette activité, j'ai utilisé la machine virtuelle Linux que je viens de créer. Je me suis rendu dans la ressource sur le portail Azure, dans Automatisation et dans Exporter le modèle. Pour finir je l'ai téléchargé.



Voici le dossier compressé contenant le template ARM :

ConnectLinuxSSH.zip

# 3.8 Déployer une VM Linux et y installer un serveur web et un index.html

Pour faire cette activité, j'ai utilisé la machine virtuelle Linux que j'ai créé précédemment, je me suis connecté en SSH avec cette commande : ssh amizeq@51.103.214.199. Pour commencer, j'ai installé apache2 grâce à cette commande : sudo apt install apache2.

```
Last login: Mon Dec 9 12:18:42 2024 from 193.5.248.1
amizeq@vm-amizeq-01:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

#### Voici l'installation :

```
Running kernel seems to be up-to-date.
No services need to be restarted.
 lo containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host. amizeq@vm-amizeq-01:~
```

Avec la commande systematl status apache, je vérifie que le service apache2 est bien installé et surtout, je vérifie qu'il est bien lancé: active (running).

```
Dec 09\ 12:28:29 vm-amizeq-01 systemd[1]: Starting The Apache HTTP Server... Dec 09\ 12:28:30 vm-amizeq-01 systemd[1]: Started The Apache HTTP Server. amizeq@vm-amizeq-01:\sim$ |
```

L'installation de nano est importante. Avec nano je vais pouvoir modifier le fichier index.html depuis le terminal. Voici la commande: sudo apt-get install nano.

```
amizeq@vm-amizeq-01:~
amizeq@vm-amizeq-01:~\$ sudo apt-get install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nano is already the newest version (6.2-lubuntu0.1).
0 upgraded, 0 newly installed, 0 to remove and 20 not upgraded.
amizeq@vm-amizeq-01:~\$ sudo nano /var/www/html/index.html
```

Avec la commande sudo nano /var/www/html/index.html j'accède au fichier index.html que je dois modifier. Je le personnalise et j'enregistre les modifications en appuyant sur Ctrl + o.

```
amizeq@vm-amizeq-01: ~
 GNU nano 6.2
                                 /var/www/html/index.html
<head>
title>Amir Zeqiri - index.html personnalisé</title>
:/head>
Voici ma page index.html personnalisée
AmNet - Amir Zeqiri
 /html>
   Help
                 Write Out
                                Where Is
                                            ∧K Cut
                                                              Execute
                                                                             Location
                  Read File
                                Replace
                                               Paste
```

# 4 SYNTHÈSE DES COMPÉTENCES DÉVELOPPÉES

# 4.1 Compétences techniques

Pendant ce projet, j'ai développé plusieurs compétences techniques :

### 4.1.1 Gestion des ressources Azure

J'ai appris à créer et gérer des budgets et des alertes pour surveiller les coûts sur Azure.

#### 4.1.2 Scripts PowerShell

J'ai acquis des connaissances pour écrire et exécuter des scripts qui listent les ressources ou VMs d'une souscription ou d'un groupe de ressources.

#### 4.1.3 Création et gestion de machines virtuelles

J'ai su configurer des VMs Linux, les connecter en SSH et les personnaliser en installant des logiciels comme Apache et Nano.

#### 4.1.4 Automatisation et templates ARM

J'ai compris comment sauvegarder les configurations des ressources pour les réutiliser ou les partager.

## 4.2 Compétences transversales

En parallèle des aspects techniques, j'ai amélioré:

### 4.2.1 La gestion du temps

La planification des tâches pour respecter les délais m'a permis de structurer mon travail efficacement.

## 4.2.2 La résolution de problèmes

Chaque défi rencontré, comme la configuration SSH ou la gestion des budgets, m'a permis de développer des approches méthodiques.

#### 4.2.3 L'autonomie

Le projet m'a poussé à chercher des solutions par moi-même en m'appuyant sur des ressources comme la documentation Azure ou des tutoriels.

## 5 PROBLÈMES RENCONTRÉS

## 5.1 Problème d'installation de logiciels sur la VM

Certaines commandes, comme l'installation d'Apache ou de Nano, renvoyaient des erreurs dues à des permissions ou à des paquets manquants.

### 6 CONCLUSION

## 6.1 Retour sur l'expérience

Ce projet m'a permis d'explorer les fonctionnalités d'Azure et de renforcer mes compétences techniques en infrastructure Cloud. J'ai particulièrement apprécié la création des scripts PowerShell et la gestion de VMs, car ces activités combinaient logique et pratique. Malgré quelques défis, j'ai pu les surmonter et mieux comprendre les solutions Cloud proposées par Microsoft.

## 6.2 Applications futures

- Écriture de scripts pour automatiser des tâches courantes.
- Gestion des coûts et optimisation des ressources dans le Cloud.
- Déploiement d'environnements Cloud

Ces connaissances me donnent une bonne base pour explorer d'autres aspects du Cloud, comme la gestion avancée des bases de données ou l'intégration avec des solutions DevOps.

## 7 ANNEXES

# 7.1 Scripts

- 7.1.1 <u>ListerRessources.ps1</u>
  - ../Scripts/ListerRessources.ps1
- 7.1.2 <u>ListerVMs.ps1</u>
  - ../Scripts/ListerVMs.ps1

## 7.2 Templates ARM

- 7.2.1 ConnectLinuxSSH.zip
  - ../Templates ARM/ConnectLinuxSSH.zip
- 7.2.2 ExportedTemplate-rg-amizeq.zip
  - ../Templates ARM/ExportedTemplate-rg-amizeq.zip
- 7.2.3 ServeurWebLinux.zip
  - ../Templates ARM/ServeurWebLinux.zip

## 8 RÉFÉRENCES

## 8.1 Convention de nommage

https://learn.microsoft.com/fr-fr/azure/cloud-adoption-framework/ready/azure-best-practices/resource-naming

https://learn.microsoft.com/fr-fr/azure/cloud-adoptionframework/ready/azure-best-practices/resource-abbreviations#aimachine-learning

# 8.2 Activité 8 (Serveur web Linux)

https://www.youtube.com/watch?v=ihljbkEQQxY https://www.youtube.com/watch?v=cqctXbNx0Yl