



## **Projet AZURE MR LENOIR**

Voici le liens vers le REPO CI/CD <https://github.com/sosoleplubo/azure>

Le liens de l'application déployé <https://purple-coast-00ccd2603.6.azurestaticapps.net/>

- Déploiement d'une application -Full-Stack- sur Microsoft Azure

## 1. Contexte et objectifs

Ce TP démontre la capacité à déployer une application -Full-Stack- (front + back + base de données) sur le cloud Azure :

- Front-end : -Vue 3- hébergé sur -Azure Static Web Apps-
- Back-end : -Node 18 / Express- exposé via -Azure App Service (Linux)-
- Base de données : -Azure SQL Database- (single DB)

### Objectifs :

1. Architecture scalable & disponible sur Internet.
2. Configuration centralisée, sécurité de l'infra.
3. Mise en place d'un pipeline CI/CD GitHub Actions.

| Resource Group           | FullStackAZURE            | West Europe    | -                       | Conteneur logique             |
|--------------------------|---------------------------|----------------|-------------------------|-------------------------------|
| App Service Plan (Linux) | FullStackBackendLinuxPlan | West Europe    | B1 (Basic)              | Hébergement du back-end       |
| Web App (Linux)          | backend-miniapp-sf        | West Europe    | Node 18-Its             | API Express                   |
| Static Web App           | FullStackSWA-Sofiane      | West Europe    | Free                    | Front Vue 3                   |
| SQL Server               | fullstack-sqlserver-sf    | France Central | Gen5                    | Moteur SQL                    |
| SQL Database             | fullstackdb               | France Central | Basic (DTU 5)           | Persistance                   |
| SQL Database             | AllowAllIPs               |                | 0.0.0.0-255.255.255.255 | Accès initial (à restreindre) |

## Configuration des services

---

### Web App (Linux)

- Runtime :- `NODE|18-lts`
- Start-up command :- `node server.js`
- Variables d'environnement (App Settings):
  - DB\_SERVER : fullstack-sqlserver-sf.database.windows.net
  - DB\_NAME : fullstackdb
  - DB\_USER : sofiane@fullstack-sqlserver-sf
  - DB\_PASSWORD : Stocké dans App Settings (à migrer vers Key Vault)
  - PORT8080

az webapp log tail

---

### Static Web App

Branch : main - repo sosoleplubo/azure  
Domain par défaut : [purple-coast-00ccd2603.6.azurestaticapps.net](https://purple-coast-00ccd2603.6.azurestaticapps.net)  
Headers personnalisés (CORS) : gérés côté back-end.

---

### Azure SQL

Authentification : SQL Login (sofiane / Abelina@1)

Schéma dbo.Users :

```
CREATE TABLE dbo.Users (
    Id      INT IDENTITY PRIMARY KEY,
    Nom    NVARCHAR(50),
    Email  NVARCHAR(100)
);

INSERT INTO dbo.Users (Nom,Email) VALUES
('Alice','alice@example.com'),
('Bob','bob@example.com');
```

---

### Sécurité et bonnes pratiques

#### Principe du moindre privilège :

Configuration via App Settings – pas de secrets dans le code.  
TLS activé (HTTPS only) sur le back-end et le front.  
CORS : Access-Control-Allow-Origin restreint au domaine du front.  
Scalabilité :

CI/CD :

Front : Action Azure/static-web-apps-deploy@v1  
Back : Déploiement Local Git (git push azure main:master).

Future enhancements :

Migrer les secrets vers Azure Key Vault & @Microsoft.KeyVault() références.  
Restreindre le Firewall SQL à l'IP sortante de l'App Service.