

# 1. Objectifs

* Garantir la continuité de service en cas d’incident (panne serveur, attaque, erreur humaine, sinistre).
* Minimiser la perte de données avec un RPO (Recovery Point Objective) acceptable pour l’entreprise.
* Réduire le temps de remise en service avec un RTO (Recovery Time Objective) réaliste.
* Respecter la règle du **3-2-1** en matière de sauvegarde.

# 2. Périmètre des données à sauvegarder

* **Base de données principale** : utilisateurs, produits, commandes, paiements.
* **Fichiers statiques** : images produits, modèles 3D (RA).
* **Configurations serveur et API** (reverse proxy, fichiers d’environnement, scripts).
* **Logs applicatifs** utiles au diagnostic.

# 3. Stratégie de sauvegarde

## 3.1 Fréquence

* **Sauvegarde incrémentale** toutes les **6 heures** (base de données et fichiers critiques).
* **Sauvegarde complète** chaque nuit à 2h00.
* **Archivage hebdomadaire** (dimanche) conservé pour 3 mois.

## 3.2 Règle 3-2-1

* **3 copies**:
  + Données de production (serveur principal)
  + Sauvegarde locale (NAS ou serveur de sauvegarde dédié)
  + Sauvegarde distante (Cloud sécurisé, ex : AWS S3 / Azure Blob / Google Cloud Storage)
* **2 supports différents** :
  + Disque dur
  + Stockage objet cloud
* **1 copie hors site** :
  + Dans un datacenter cloud situé en Europe (RGPD compliant).

## 3.3 Méthodes techniques

* Base de données:
  + Utilisation d’outils natifs (ex. pg\_dump / pg\_basebackup si PostgreSQL).
  + Compression + chiffrement AES-256 des dumps.
* Fichiers statiques :
  + Synchronisation avec versionnement (rsync + snapshots).
  + Automatisation via scripts + planification (cron ou orchestrateur type Ansible).

# 4. Sécurité des sauvegardes

* Chiffrement systématique des sauvegardes (AES-256).
* Authentification forte (MFA) pour accéder aux sauvegardes distantes.
* Rétention conforme aux contraintes légales (ex. données clients protégées par RGPD).

# 5. Test et validation des sauvegardes

* **Vérification automatique** après chaque sauvegarde (checksum, intégrité).
* **Test de restauration complet** effectué :
* Mensuellement en préproduction.
* Trimestriellement en production simulée.

# 6. Gouvernance et responsabilités

* **Administrateur système** : configuration et supervision des sauvegardes.
* **CTO / Responsable technique** : validation et audit des tests de restauration.
* **DPO (si applicable)** : contrôle du respect RGPD.

# 7. Indicateurs de suivi (KPI)

* Taux de succès des sauvegardes (%)
* Temps moyen de restauration (RTO mesuré)
* Âge moyen de la dernière sauvegarde exploitable (RPO)

# 8. Plan de Redéploiement

**1. Préparation**

* **Resource Group** : créer *MaisonEnPlace-Prod*.
* **Réseau** : provisionner un **Azure Virtual Network (VNet)** + sous-réseaux (app, DB, monitoring).
* **Sécurité** : configurer **Network Security Groups (NSG)** → autoriser 80/443, restreindre 5432 aux services applicatifs.

**2. Base de données**

* Créer **Azure Database for PostgreSQL – Flexible Server** (Europe, haute dispo activée).
* Configurer sauvegardes automatiques (rétention 7–35 jours).
* Importer le backup

**3. Fichiers statiques**

* Créer un **Azure Storage Account** (zone Europe).
* Activer **Blob Storage** pour stocker images et modèles 3D.
* Restaurer depuis backup

**4. Application**

* Créer un **Azure Container Registry (ACR)** pour stocker l’image Docker.
* Créer un **App Service Plan** (Linux, Standard ou Premium pour scaling).
* Déployer via Docker
* Configurer variables d’environnement : *DATABASE\_URL, STORAGE\_URL, SECRET\_KEY, etc.*

**5. Reverse Proxy et TLS**

* Activer **Managed Certificates** sur App Service pour HTTPS.
* Configurer **Application Gateway** (load balancing global).

**6. Monitoring et sauvegardes**

* Activer **Application Insights** pour logs et métriques.
* DB : sauvegardes natives Azure (restauration point-in-time).
* Fichiers statiques : versioning activé sur Blob Storage.
* App : snapshots automatiques d’App Service si activés.

**7. Automatisation**

* Créer **GitHub Actions** : build → push image vers ACR → déploiement App Service.
* Créer **Runbooks dans Azure Automation** pour tester régulièrement la restauration.

**8. Tests et bascule**

* Déployer d’abord en **préproduction** (resource group séparé).
* Importer un backup récent et exécuter des **smoke tests** (login, catalogue, paiement sandbox).
* Basculer le DNS vers App Service une fois validé.

**9. Gouvernance et responsabilités**

* **Administrateur Azure** : supervision RG, réseau, NSG.
* **Développeurs** : pipeline CI/CD, déploiement.
* **CTO/DPO** : conformité RGPD et audits réguliers.

**10. Rollback**

* **Application** : redeployer image précédente depuis ACR.
* **BDD** : utiliser restauration point-in-time (Azure PostgreSQL).
* **DNS** : repointer vers ancienne version si nécessaire.