# Déveloррement et Déрloiement ԁu Journаl ԁe Grаtituԁe

Introԁuсtion   
  
Ce рrojet сonsiste à ԁéveloррer une аррliсаtion web nommée Journаl ԁe Grаtituԁe рermettаnt аux utilisаteurs ԁ’аjouter, ԁe moԁifier et ԁe suррrimer ԁes entrées exрrimаnt leur grаtituԁe. L'аррliсаtion аnаlyse égаlement le sentiment ԁes entrées grâсe аu serviсe Azure Cognitive Serviсes et stoсke les ԁonnées ԁаns un сomрte Azure Storаge. Enfin, l'аррliсаtion а été ԁéрloyée sur un Aрр Serviсe Azure en utilisаnt Doсker.

**Étарes Réаlisées  
  
1. Configurаtion ԁes Serviсes Azure**

* Créаtion ԁes ressourсes Azure:
  + **Resource Group** : GratitudeAppRG.
  + **App Service** : gratitude-journal-app.
  + **Azure Storage** : gratitudestorage (table utilisée : GratitudeEntries).
  + **Service Azure Language** : gratitude-language-service pour l’analyse des sentiments.

**Capture d'écran** :   
  
Une image contenant texte, Police, ligne, nombre

Description générée automatiquement

**2. Développement de l'Application**

* **Technologies Utilisées** :
  + Backend : Flask.
  + Services Azure : Azure Storage et Azure Cognitive Services (Text Analytics).
  + Frontend : HTML, CSS

**Principales Fonctionnalités** :

* **Gestion des utilisateurs** :
  + Connexion sécurisée pour les utilisateurs standard et administrateurs, permettant des accès différenciés.
* **Gestion des entrées** :
  + Ajout, modification et suppression des entrées dans un journal de gratitude interactif.
* **Analyse des sentiments** :
  + Analyse automatique des entrées grâce à un service d’analyse de sentiment, détectant l’humeur (positive, neutre, négative) de l’utilisateur.
  + Génération de messages de motivation personnalisés adaptés à l’humeur détectée, pour encourager ou remonter le moral des utilisateurs.
* **Statistiques personnalisées** :
  + Affichage des tendances émotionnelles des utilisateurs (ex : Nombre d’entrées positives, neutres ou négatives) sur une période donnée.
* **Accessibilité et design** :
  + Interface utilisateur intuitive et moderne, avec un mode clair et sombre pour s’adapter aux préférences des utilisateurs.
* **Stockage sécurisé** :
  + Sauvegarde des données dans une base de données Cloud, assurant leur accessibilité et leur confidentialité.

**Captures d'écran** :

* Page de connexion (mode Sombre/Clair) :  
  Une image contenant texte, capture d’écran, Police, conception

  Description générée automatiquement

Une image contenant texte, capture d’écran, conception

Description générée automatiquement

* Interface principale avec les entrées :  
  Une image contenant texte, capture d’écran, Système d’exploitation, logiciel

  Description générée automatiquement
* Modification d'une entrée :  
  Une image contenant texte, capture d’écran, Police, Système d’exploitation

  Description générée automatiquement

**3. Tests Locaux**

* Exécution locale de l'application via Flask pour valider les fonctionnalités avant le déploiement.
* Test des connexions utilisateur, ajout/modification/suppression d'entrées, et analyse des sentiments.

**Capture d'écran** : **ADMIN / Utilisateur standard**  
  
Une image contenant texte, logiciel, Icône d’ordinateur, Page web

Description générée automatiquement

Une image contenant texte, logiciel, Icône d’ordinateur, Page web

Description générée automatiquement

**4. Création et Configuration de l'Image Docker**

**Fichier Dockerfile :**

# Utiliser une image Python officielle

FROM python:3.9-slim

# Définir le répertoire de travail

WORKDIR /app

# Copier les fichiers nécessaires

COPY requirements.txt .

COPY . .

# Installer les dépendances

RUN pip install -r requirements.txt

# Exposer le port pour Flask

EXPOSE 5000

# Commande pour démarrer l'application

CMD ["python", "app.py"]  
  
  
**Commandes Docker** :

docker build -t flask-gratitude-app .

docker tag flask-gratitude-app nassim4881/flask-gratitude-app:latest

docker login

docker push nassim4881/flask-gratitude-app:latest  
  
**Capture d'écran** :   
Une image contenant texte, logiciel, nombre, diagramme

Description générée automatiquement

**5. Déploiement sur Azure avec Docker**

* Déploiement de l'image Docker sur l'App Service :
  + Configuration du conteneur avec l'image Docker : nassim4881/flask-gratitude-app:latest.
* Résultat : L'application est accessible à l'adresse suivante : [**https://gratitude-journal-app-a9eha6cqhjawd2ct.northeurope-01.azurewebsites.net**](https://gratitude-journal-app-a9eha6cqhjawd2ct.northeurope-01.azurewebsites.net)
* Une image contenant texte, capture d’écran, Police

  Description générée automatiquement