

# Dossier d'Architecture Technique

## **Table des matières :**

1. Contexte
  - 1.1. Objectif
  - 1.2. Acteurs
  - 1.3. Fonctionnalités principales
2. Architecture
  - 2.1. Choix de l'architecture
  - 2.2. Rôle des tiers
  - 2.3. Flux d'informations
  - 2.4. Déploiement
3. Pourquoi cette architecture
  - 3.1. Besoins fonctionnels
4. Risques
  - 4.1. Perspectives

# **1. Contexte**

## **1.1. Objectif**

Ce document détaille l'architecture technique d'une application Web, Daily Posting, qui a pour but d'être un fil d'actualités.

## **1.2. Acteurs**

Cette plateforme est ouverte à tout public.  
On peut accéder au site avec tout navigateur Web en suivant le lien suivant :  
<https://desolate-meadow-43082.herokuapp.com/>

## **1.3. Fonctionnalités principales**

Elle permet à l'utilisateur de poster des messages que tout utilisateur pourra consulter sur un navigateur Web.

## **2. Architecture**

### **2.1. Choix de l'architecture**

Il s'agit de mettre en place un serveur Web qui puisse gérer la création d'utilisateurs et ainsi leurs publications sur le fil d'actualités.

### **2.2. Rôle des tiers**

Pour cela, l'application a besoin d'un serveur pour traiter les données que le site reçoit et affiche, d'une base de données pour stocker toutes ces informations et d'un cloud où l'application pourra être déployée.

### **2.3. Flux d'informations**

Pour éviter le débordement d'informations, on choisit de donner aux messages une durée de vie de 1 jour.

### **2.4. Déploiement**

Suivant les recommandations du professeur encadrant et de collègues de travail, l'application Web est déployée avec Dokku et hébergée sur un serveur distant.

### **3. Pourquoi cette architecture**

L'ensemble des services que propose Heroku permet de faciliter le développement de notre application notamment en la rendant plus fiable. N'ayant pas trop d'expérience en la matière, je suivais plusieurs tutoriels en ligne et finit par choisir de faire une application de type MEVN, c'est à dire en utilisant MongoDB pour stocker les données, Express et Node.js pour le serveur et Vue pour l'affichage sur le site.

### **4. Risques**

Pour l'instant l'application ne restreint pas les utilisateurs au niveau de la publication de messages ainsi qu'à la suppression de ceux-ci.