lien GitHub

PASSEZ UN SYSTÈME IA DU POC AU MVP

Passage du POC au MVP – Chatbot RAG

Projet Data Engineer – KELLENI Antoine

ROADMAP DE LA MISSION

1. **Contexte & Objectifs**
2. Réalisation et limites du POC
3. Enjeux et phases du MVP
4. Architecture globale de la solution
5. Choix du cloud provider
6. Macro Backlog
7. Estimations des coûts
8. Conclusion

Contexte & objectifs

Contexte :

Besoin de recommandation d'événements culturels pertinents
Utilisation de données publiques issues d'OpenAgenda
Recherche conversationnelle plus naturelle via IA
Approche basée sur un chatbot RAG (Retrieval Augmented Generation)

Objectifs :

Valider la faisabilité d'un chatbot RAG (POC)
Transformer le POC en MVP exploitable
Concevoir une architecture scalable et modulaire
Maîtriser les coûts et les risques techniques
Préparer une mise en production cloud

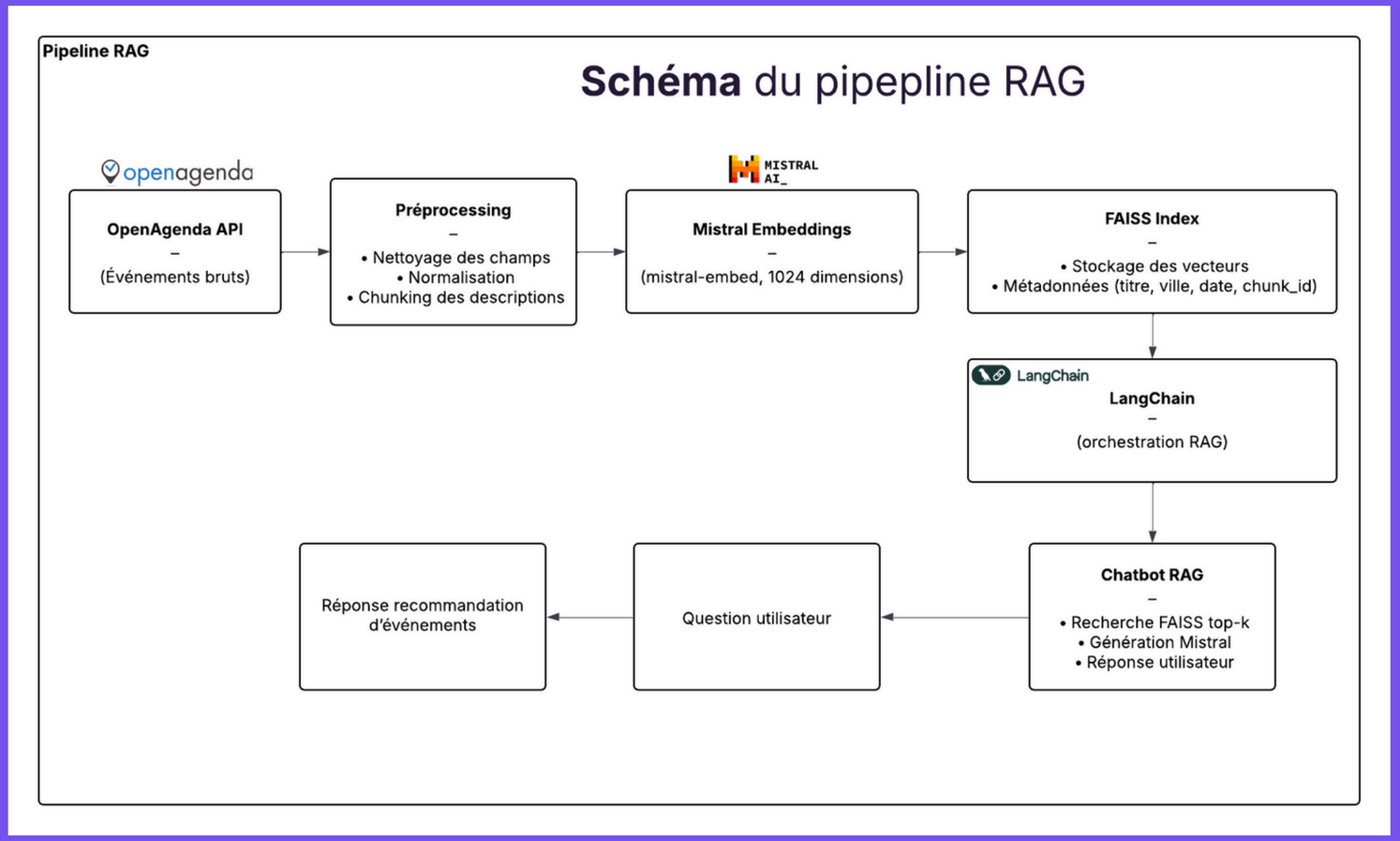
ROADMAP DE LA MISSION

1. Contexte & Objectifs
2. **Réalisation et limites du POC**
3. Enjeux et phases du MVP
4. Architecture globale de la solution
5. Choix du cloud provider
6. Macro Backlog
7. Estimations des coûts
8. Conclusion

Réalisation du POC

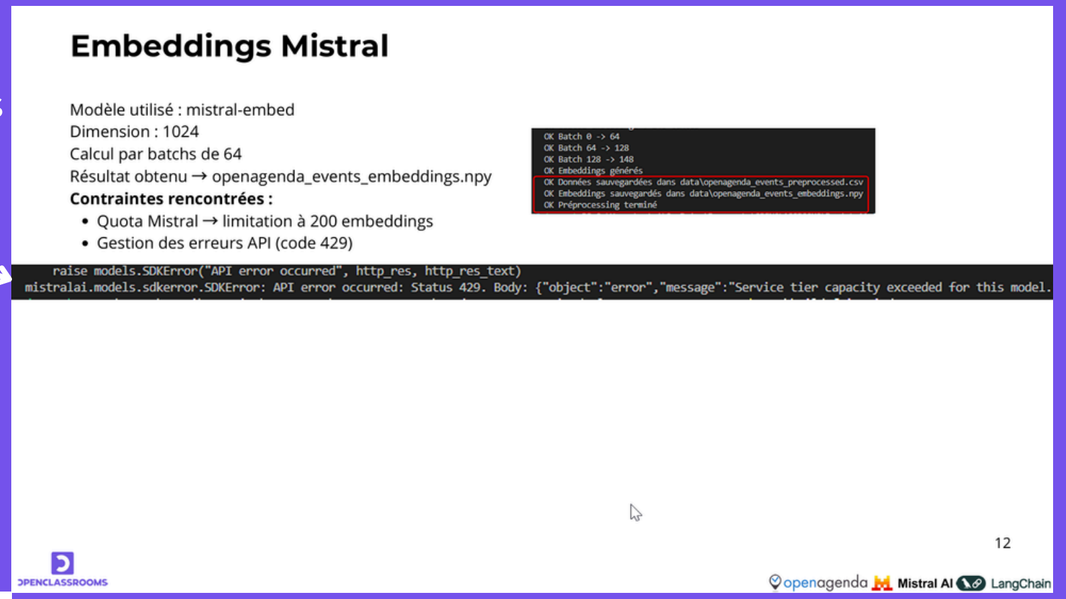
Réalistion clés effectués dans le passé :

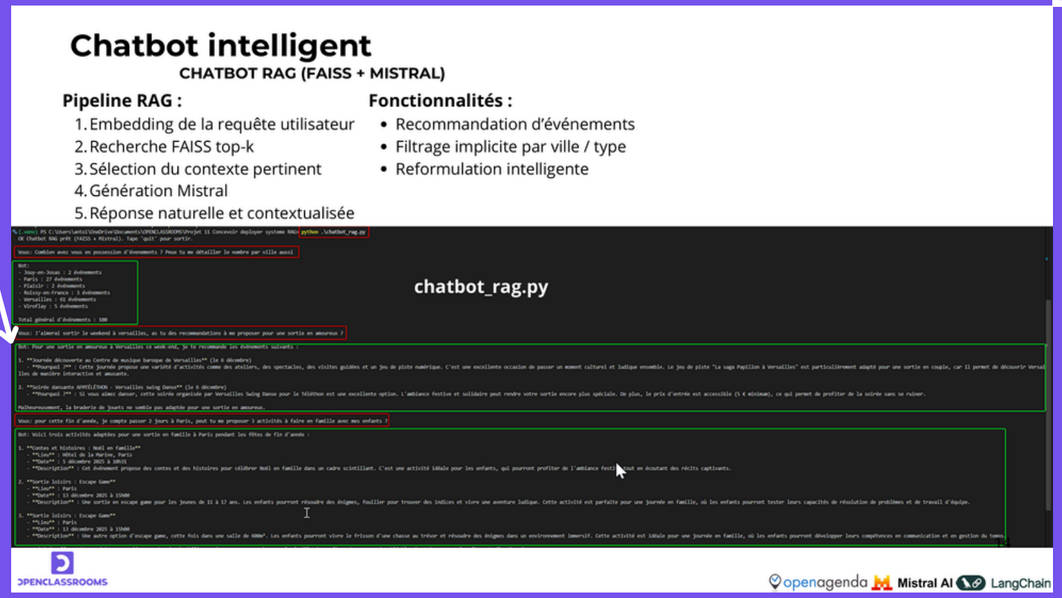
Pipeline Python local
Ingestion des données OpenAgenda
Nettoyage et normalisation
Vectorisation NLP (embeddings Mistral)
Indexation FAISS
Chatbot conversationnel fonctionnel



Limites identifiées du POC

Quotas et crédits API Mistral limitant la génération d'embeddings
Erreurs API lors de volumes plus importants
Architecture locale non scalable
Absence de déploiement cloud
Pas de monitoring ni d'observabilité
Interface utilisateur minimale (terminal)





OPENCLASSROOMS

ROADMAP DE LA MISSION

1. Contexte & Objectifs
2. Réalisation et limites du POC
3. **Enjeux et phases du MVP**
4. Architecture globale de la solution
5. Choix du cloud provider
6. Macro Backlog
7. Estimations des coûts
8. Conclusion

Enjeux du passsage au MVP

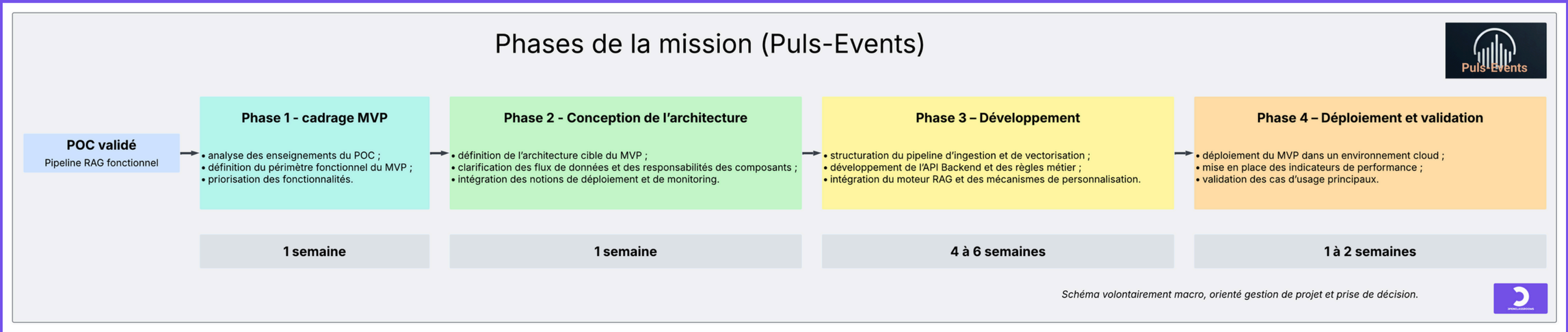
Pourquoi passer au MVP ?

Industrialiser le pipeline existant
Séparer clairement les responsabilités
Améliorer la maintenabilité
Anticiper la montée en charge
Intégrer déploiement, monitoring et maîtrise des coûts

Transition vers le MVP

Analyse critique des limites du POC
Définition d’un périmètre fonctionnel réaliste
Priorisation des fonctionnalités (Must / Nice)
Choix d’une architecture cible orientée production

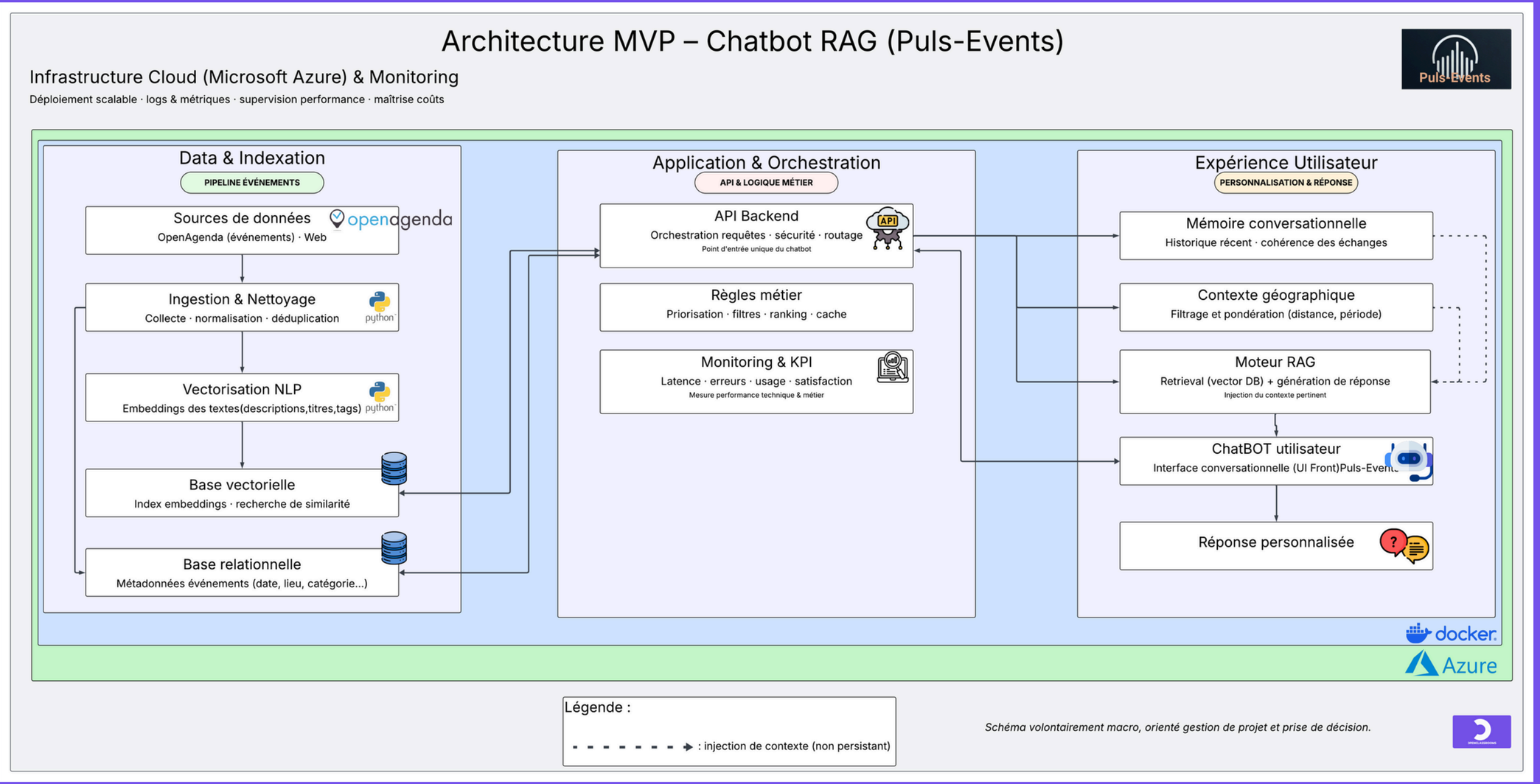
Les différentes phases du MVP



ROADMAP DE LA MISSION

1. Contexte & Objectifs
2. Réalisation et limites du POC
3. Enjeux et phases du MVP
4. **Architecture globale de la solution**
5. Choix du cloud provider
6. Macro Backlog
7. Estimations des coûts
8. Conclusion

Architecture globale de la solution

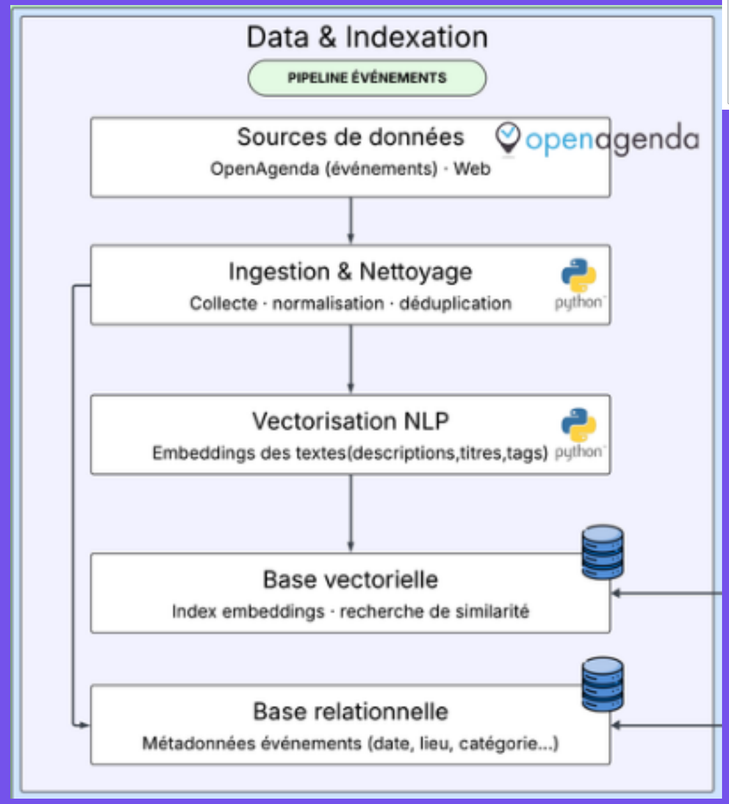


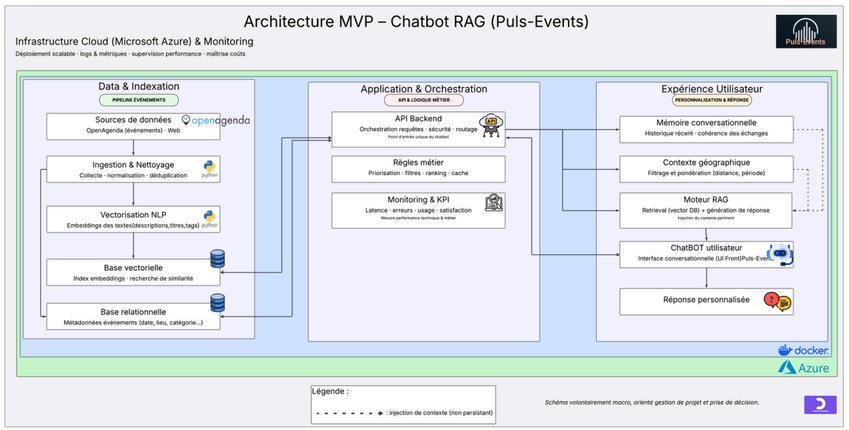
Séparation en trois couches :
Data & Indexation
Application & Orchestration
Expérience Utilisateur

API Backend comme point d'entrée unique
Intégration d'un moteur RAG
Architecture modulaire et évolutive

1er couche : Data & Indexation

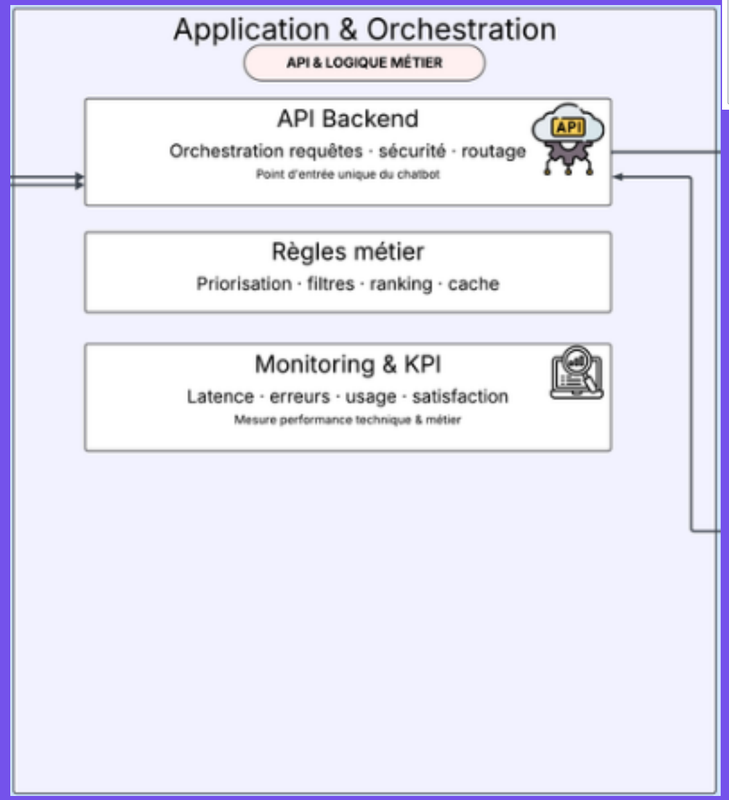
Sources : OpenAgenda (événements)
Ingestion et nettoyage automatisés
Vectorisation NLP
Base relationnelle :
    métadonnées (date, lieu, catégorie…)
Base vectorielle :
    recherche sémantique par similarité

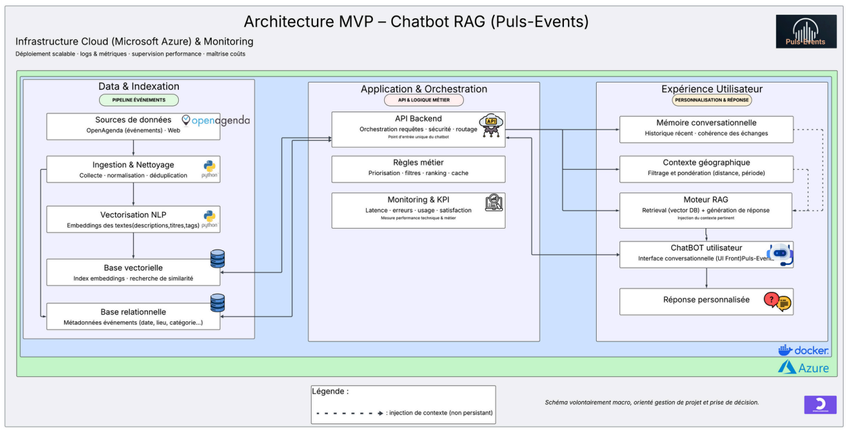




2e couche : Application & Orchestration

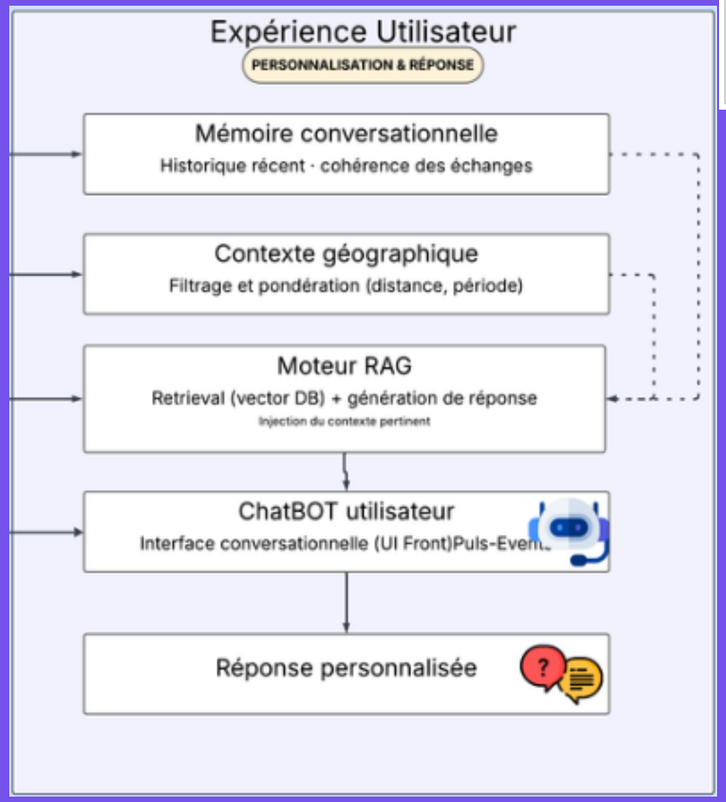
- API Backend :
  - orchestration des requêtes
  - sécurité et routage
- Règles métier :
  - filtres, priorisation, ranking, cache
- Monitoring & KPI :
  - latence, erreurs, usage, satisfaction

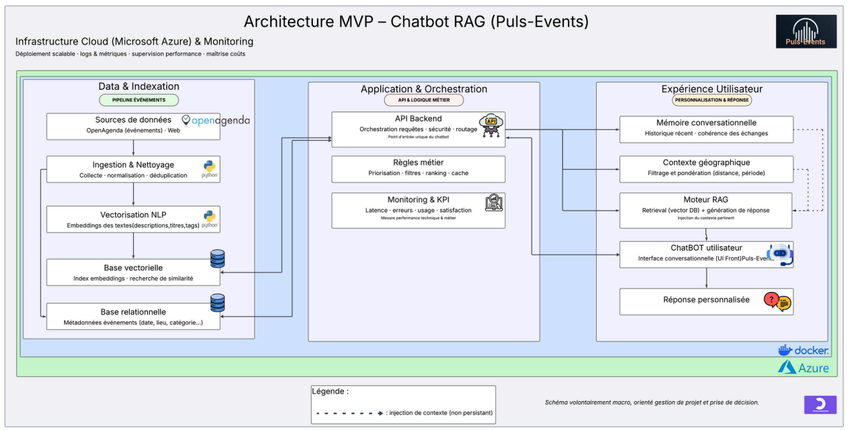




3eme couche : Expérience Utilisateur

Interface chatbot conversationnelle
Mémoire conversationnelle (court terme)
Contexte géographique injecté
Moteur RAG :
    retrieval depuis la base vectorielle
    génération de réponses contextualisées
Réponse personnalisée à l'utilisateur





ROADMAP DE LA MISSION

1. Contexte & Objectifs
2. Réalisation et limites du POC
3. Enjeux et phases du MVP
4. Architecture globale de la solution
5. **Choix du cloud provider**
6. Macro Backlog
7. Estimations des coûts
8. Conclusion

Choix du Cloud Provider

- Veille menée sur :
  - AWS
  - GCP
  - Microsoft Azure

- Choix retenu : Microsoft Azure

Justifications

- Déploiement rapide
- Scalabilité progressive
- Monitoring intégré
- Modèle pay-as-you-go
- Adapté à une démarche MVP

ROADMAP DE LA MISSION

1. Contexte & Objectifs
2. Réalisation et limites du POC
3. Enjeux et phases du MVP
4. Architecture globale de la solution
5. Choix du cloud provider
6. **Macro Backlog**
7. Estimations des coûts
8. Conclusion

Macro Backlog

Fonctionnalité	Description	Priorité	Complexité	Estimation délai	Risques identifiés	Mitigation
Ingestion OpenAgenda	Collecte automatisée des événements via API OpenAgenda	Must-Have	Moyenne	3-4 jours	Volume important	Pagination, filtres
Nettoyage & normalisation	Nettoyage texte, normalisation champs, déduplication	Must-Have	Moyenne	3 jours	Données hétérogènes	Règles de nettoyage
Stockage relationnel	Base relationnelle pour métadonnées (date, lieu, catégorie…)	Must-Have	Moyenne	2-3 jours	Modèle mal adapté	Schéma simple & évolutif
Chunking des descriptions	Découpage des textes longs avec overlap	Must-Have	Faible	1-2 jours	Perte de contexte	Overlap contrôlé
Génération embeddings	Embeddings NLP avec Mistral	Must-Have	Moyenne	2-3 jours	Quota API	Limitation volume
Base vectorielle	Index FAISS pour recherche sémantique	Must-Have	Moyenne	2 jours	Performance	Index simple
API Backend	Point d'entrée unique du chatbot	Must-Have	Élevée	5-7 jours	Couplage fort	Architecture modulaire
Moteur RAG	Retrieval + génération de réponse	Must-Have	Élevée	5 jours	Hallucinations	Contexte contrôlé
Chatbot utilisateur	Interface conversationnelle (CLI / UI simple)	Must-Have	Faible	2 jours	UX limitée	MVP assumé
Règles métier	Filtres date / ville / catégorie / ranking	Nice-to-Have	Moyenne	3 jours	Complexité logique	Règles simples
Contexte géographique	Pondération par distance	Nice-to-Have	Moyenne	2 jours	Données imprécises	Géoloc simplifiée
Monitoring & KPI	Logs, erreurs, latence	Nice-to-Have	Faible	1-2 jours	Manque visibilité	Logs centralisés

ROADMAP DE LA MISSION

1. Contexte & Objectifs
2. Réalisation et limites du POC
3. Enjeux et phases du MVP
4. Architecture globale de la solution
5. Choix du cloud provider
6. Macro Backlog
7. **Estimations des coûts**
8. Conclusion

Estimation des coûts

BUILD

Poste de coût	Description	Nature du coût	Complexité	Charge estimée
Cadrage & conception	Analyse POC, définition MVP, architecture cible	Ponctuel	Moyenne	Moyenne
Pipeline data	Ingestion OpenAgenda, nettoyage, normalisation	Ponctuel	Moyenne	Moyenne
Base relationnelle	Modélisation et mise en place des métadonnées	Ponctuel	Moyenne	Faible à moyenne
Vectorisation NLP	Génération des embeddings et indexation	Ponctuel	Moyenne	Moyenne
Base vectorielle	Mise en place FAISS et tests de performance	Ponctuel	Moyenne	Faible
API Backend	Orchestration, sécurité, règles métier	Ponctuel	Élevée	Élevée
Moteur RAG	Retrieval + génération contrôlée	Ponctuel	Élevée	Élevée
Interface chatbot	Interface utilisateur simple (CLI / UI web)	Ponctuel	Faible	Faible
Tests & validation	Tests fonctionnels et cas d'usage	Ponctuel	Faible	Faible
Documentation	Documentation technique & projet	Ponctuel	Faible	Faible

OPEX

Poste de coût	Description	Type	Impact selon charge utilisateur
Hébergement cloud	Exécution API, RAG, UI (conteneurs)	Récurrent	Moyen
Stockage relationnel	Métadonnées événements	Récurrent	Faible
Stockage vectoriel	Index embeddings FAISS	Récurrent	Faible à moyen
Appels NLP / LLM	Embeddings + génération RAG	Variable	**Élevé**
Réseau	Appels API, flux entrants/sortants	Récurrent	Faible
Monitoring & logs	Collecte métriques et erreurs	Récurrent	Faible
Maintenance	Correctifs et ajustements	Récurrent	Moyen

Optimisation budgetaire

Levier	Description	Effet
Limitation du périmètre MVP	Fonctionnalités essentielles uniquement	Réduction BUILD
Réduction volume embeddings	Vectorisation ciblée	Réduction OPEX
Cache des réponses	Réutilisation résultats fréquents	Réduction appels LLM
Batch processing	Traitements groupés	Réduction coûts NLP
Montée en charge progressive	Ajustement à l'usage réel	Maîtrise OPEX
Architecture modulaire	Remplacement facile des briques	Pérennité

ROADMAP DE LA MISSION

1. Contexte & Objectifs
2. Réalisation et limites du POC
3. Enjeux et phases du MVP
4. Architecture globale de la solution
5. Choix du cloud provider
6. Macro Backlog
7. Estimations des coûts
8. **Conclusion**

Conclusion

Bilan

Limites techniques identifiées et traitées
MVP structuré et orienté production
Architecture scalable et maintenable
Maîtrise des coûts et des risques

Perspectives

Amélioration de l’expérience utilisateur
Optimisation des coûts LLM
Montée en charge progressive
Déploiement à plus grande échelle

Conclusion

Projet mené de bout en bout : POC → MVP

MERCI