

# RAPPORT TECHNIQUE : CHOIX DES SERVICES POUR LA PLATEFORME FREELANCE

## PROBLÈME

Actuellement, la startup gère les profils des freelances (informations personnelles, compétences, liens professionnels) de manière non structurée via e-mails et WhatsApp. Cette approche présente de nombreux inconvénients : un suivi complexe et chronophage, des risques de sécurité accrus en l'absence de gestion centralisée des données sensibles, et une difficulté manifeste à faire évoluer la gestion des profils à mesure que le nombre de freelances augmente. Il est impératif de moderniser cette infrastructure.

## OBJECTIF DU PROJET

L'objectif principal de ce projet est de concevoir et de mettre en œuvre une plateforme centralisée et robuste qui permettra de :

- Créer, mettre à jour et supprimer les profils des freelances de manière intuitive et efficace.
- Gérer les compétences et les liens professionnels de chaque freelance de manière structurée et organisée.
- Garantir la sécurité des données sensibles, assurer la scalabilité de la solution pour accompagner la croissance de la startup, et faciliter l'accès aux données pour les équipes internes.

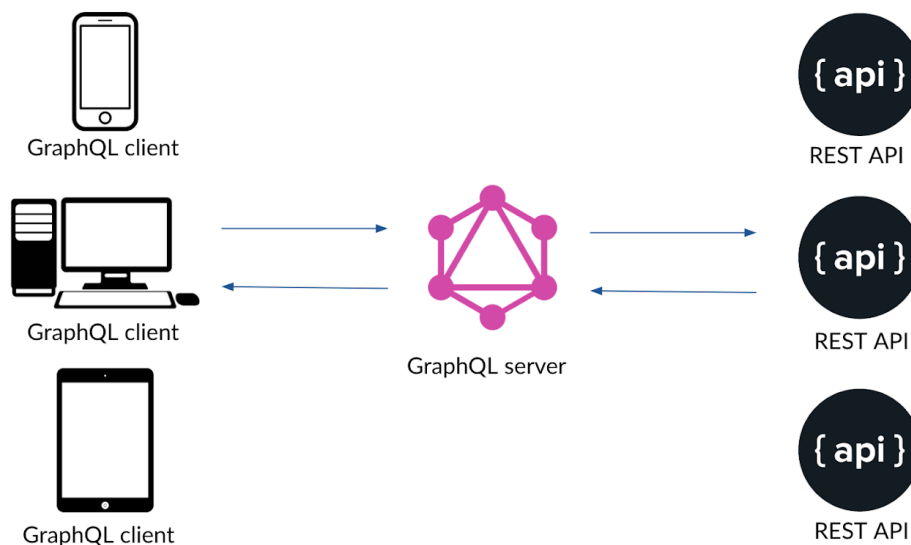
# CHOIX DES TECHNOLOGIES & SERVICES

## 1. INTERFACE API : REST OU GRAPHQL ?

✓ Choix : GraphQL

Pourquoi ?

Les profils utilisateurs de la plateforme freelance sont par nature complexes et contiennent des champs variés tels que des informations personnelles, des compétences multiples, et divers liens professionnels. GraphQL offre une flexibilité supérieure à REST ou SOAP, permettant aux clients de demander précisément les données dont ils ont besoin. Cela optimise les échanges réseau en évitant la surcharge de données (over-fetching) et les requêtes multiples (under-fetching), ce qui est particulièrement bénéfique pour les applications frontend modernes (web et mobile) nécessitant une grande efficacité.



GraphQL for BE devs | @engfragui

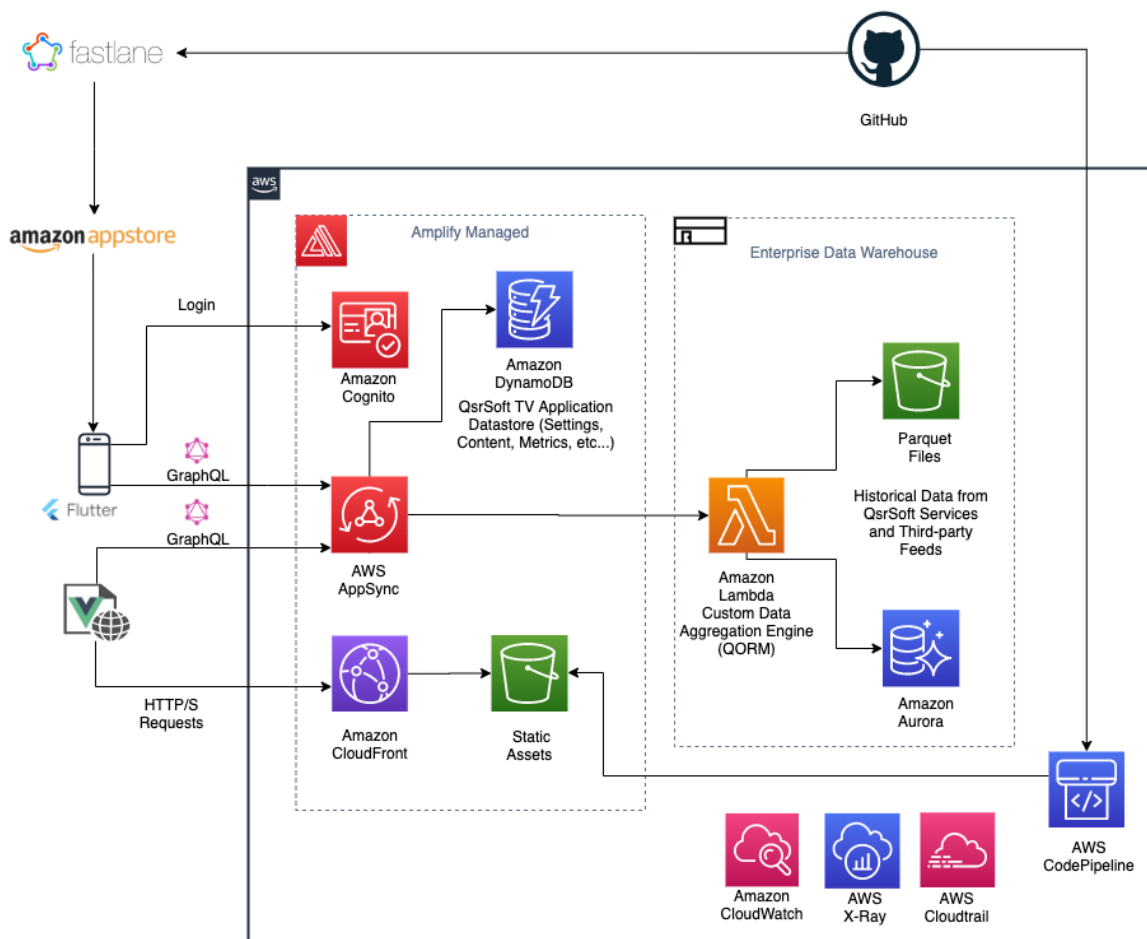
Explication sous l'image : Le frontend utilise GraphQL pour interroger uniquement les champs nécessaires (par exemple, le prénom et les compétences du freelance). Cette approche réduit la quantité de données transférées, optimisant ainsi les échanges et évitant les surcharges inutiles, ce qui améliore la performance globale de l'application.

## 2. BACKEND & API GATEWAY

✓ Choix : AWS AppSync (pour GraphQL)

Pourquoi ?

AWS AppSync est un service géré de GraphQL, ce qui signifie qu'il élimine la charge de gestion des serveurs d'API. Il s'intègre nativement et facilement avec d'autres services AWS clés comme Amazon DynamoDB (pour la base de données) ou Amazon Aurora. De plus, AppSync prend en charge des fonctionnalités essentielles telles que l'authentification, la mise en cache des données et les WebSockets pour des mises à jour en temps réel des profils, ce qui est crucial pour une expérience utilisateur dynamique.



Explication : AWS AppSync agit comme le serveur GraphQL principal de la plateforme. Il se connecte directement à DynamoDB pour le stockage des profils et peut déclencher des fonctions AWS Lambda pour exécuter la logique métier complexe, comme la vérification des données ou des

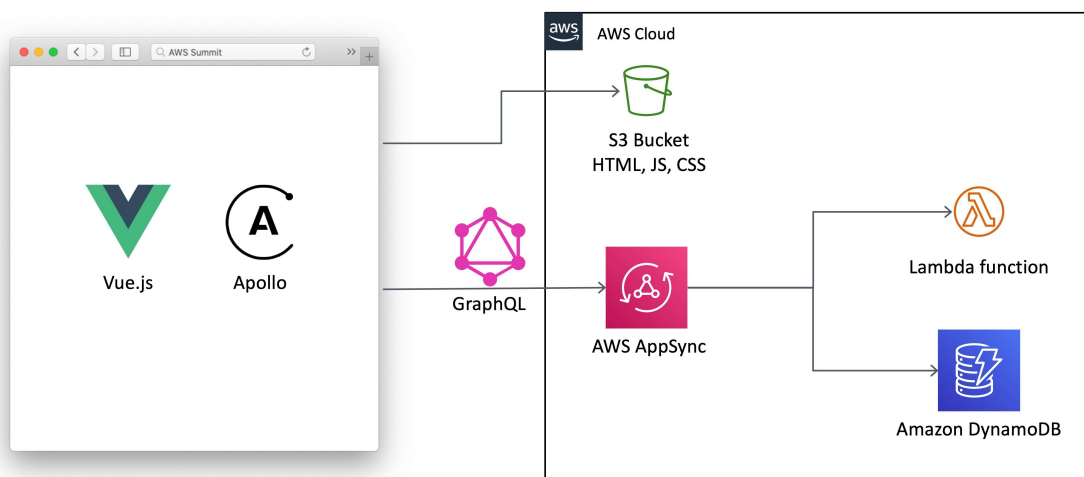
traitements asynchrones, assurant ainsi une architecture performante et découplée.

### 3. BASE DE DONNÉES

✓ Choix : Amazon DynamoDB

Pourquoi ?

Amazon DynamoDB est une base de données NoSQL entièrement gérée, ce qui la rend parfaitement adaptée à la gestion de données semi-structurées et évolutives comme les profils de freelances (qui peuvent inclure des listes de compétences, des liens variables, etc.). Sa haute scalabilité permet de gérer un volume croissant de données et de requêtes sans nécessiter de gestion de serveurs. Son intégration native avec AWS AppSync simplifie grandement l'architecture et le développement. En cas de besoin d'une structure très relationnelle, Amazon Aurora (compatible PostgreSQL) pourrait être une alternative.



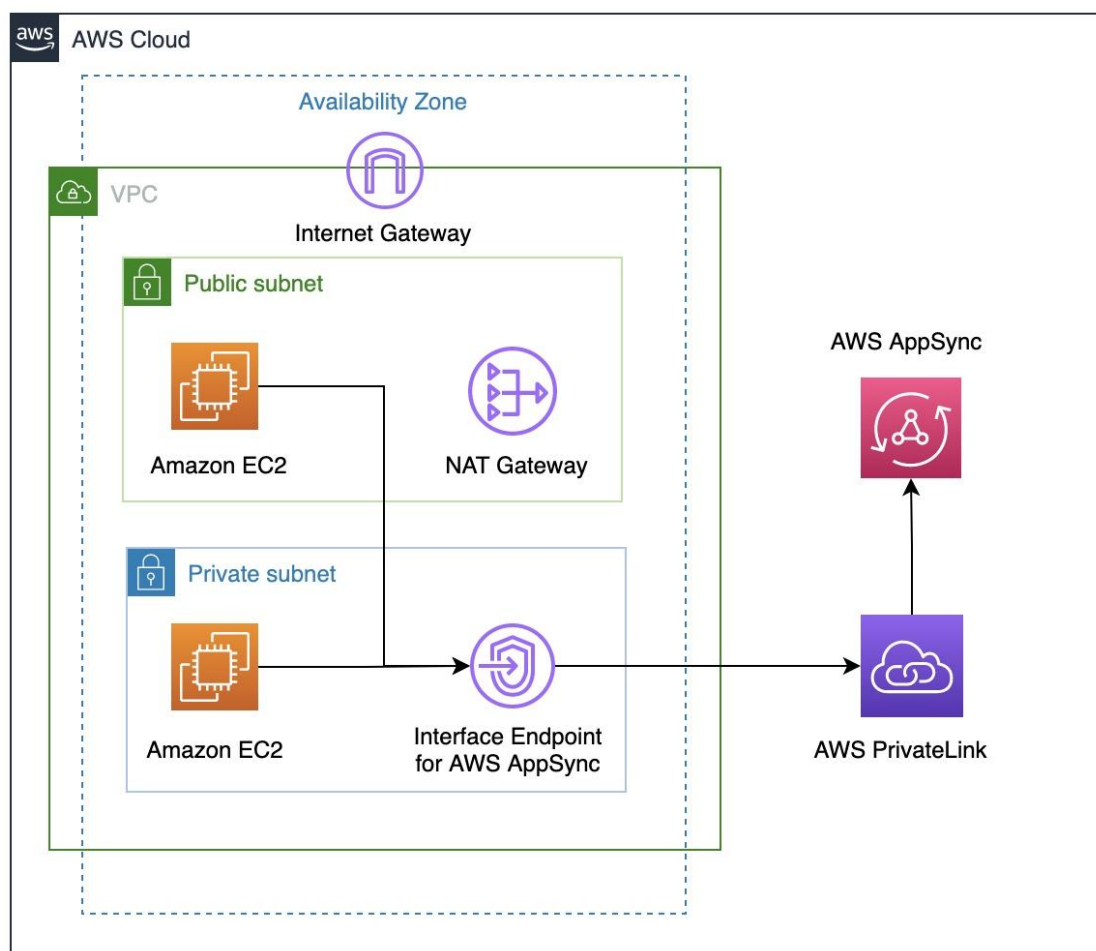
**Explication :** DynamoDB permet de stocker les profils des freelances sous forme de documents JSON, une structure flexible qui s'adapte idéalement à l'évolution des attributs (nouveaux types de liens, compétences additionnelles). Cette flexibilité est cruciale pour une plateforme en constante évolution.

## 4. FONCTIONS DE TRAITEMENT (LOGIQUE MÉTIER)

✓ Choix : AWS Lambda

Pourquoi ?

AWS Lambda est un service de calcul sans serveur qui exécute du code en réponse à des événements, éliminant ainsi la nécessité de provisionner ou de gérer des serveurs. Cela garantit des coûts faibles, car vous ne payez que pour le temps de calcul consommé. Lambda est parfait pour exécuter la logique métier spécifique de la plateforme, comme la validation des données d'un profil, l'envoi de notifications (email, Slack), ou des traitements asynchrones après une mise à jour de profil, s'intégrant de manière transparente avec AppSync et DynamoDB.



Explication : Les fonctions AWS Lambda encapsulent la logique métier. Par exemple, lorsqu'un freelance met à jour son profil, AppSync peut déclencher

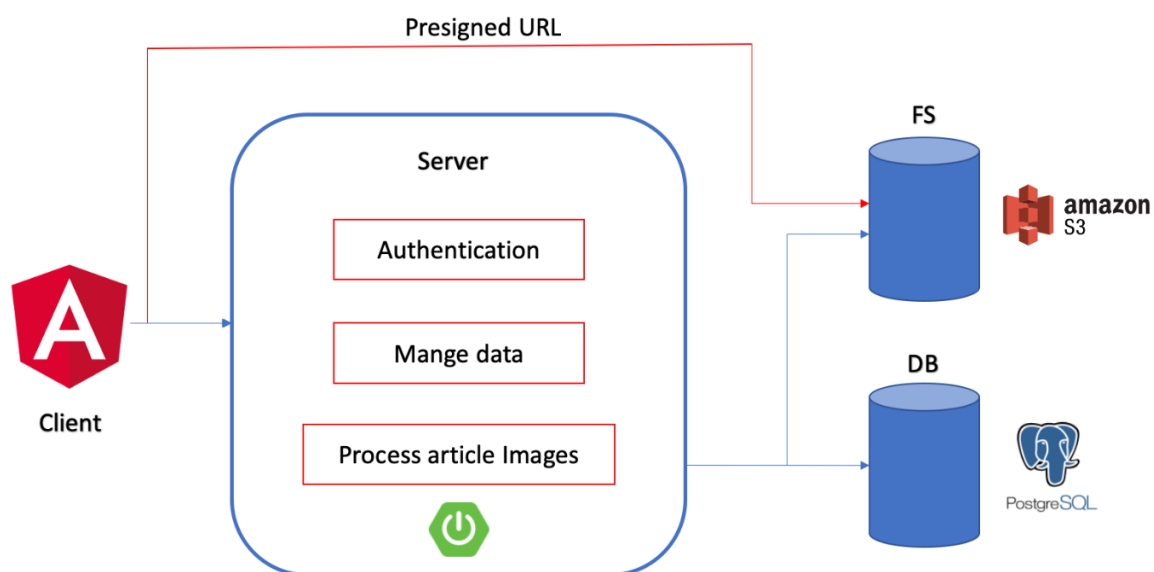
une fonction Lambda pour valider le format de l'email, vérifier l'unicité d'un pseudo ou envoyer une notification automatique à l'équipe de gestion via Slack ou email.

## 5. STOCKAGE DE FICHIERS (CV, IMAGES DE PROFIL)

✓ Choix : Amazon S3

Pourquoi ?

Amazon S3 (Simple Storage Service) est un service de stockage d'objets hautement sécurisé, durable et évolutif. Il est idéal pour stocker des fichiers statiques volumineux comme les CV des freelances, leurs portfolios ou leurs images de profil. Son intégration est simple avec les autres services AWS, et il permet de restreindre l'accès aux fichiers via des URLs signées pré-générées, assurant que seuls les utilisateurs autorisés peuvent télécharger ou visualiser des documents sensibles.



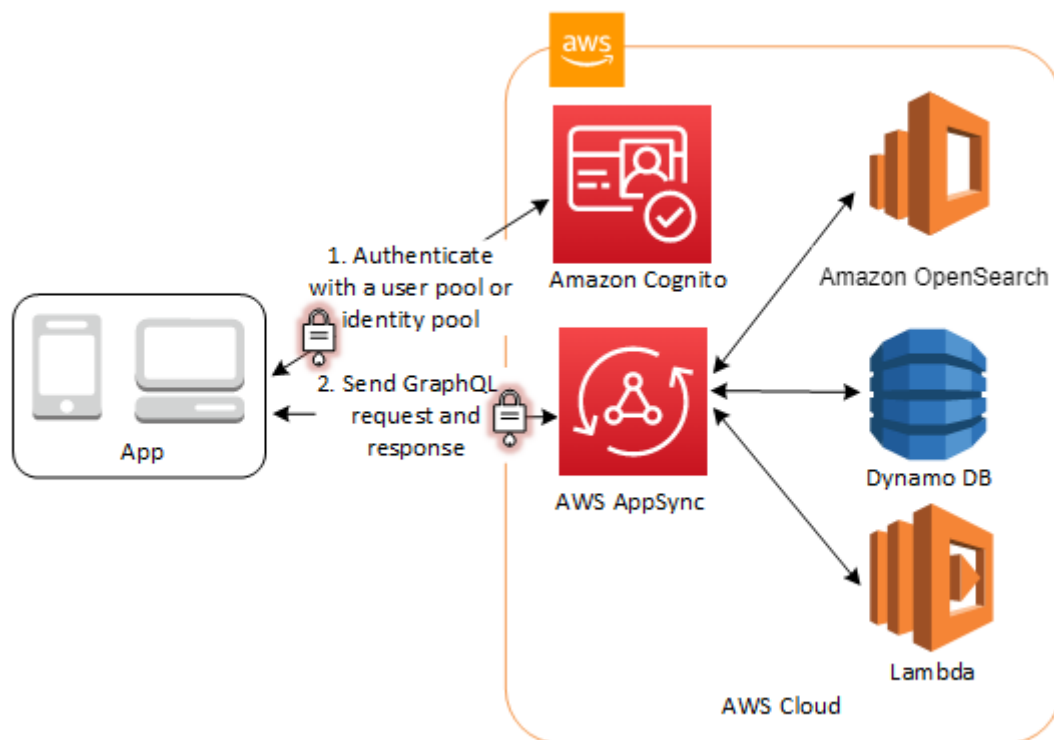
Explication : Les fichiers tels que les images de profil ou les CV des freelances sont stockés de manière sécurisée sur S3. Une fois téléchargés, les liens vers ces fichiers sont stockés dans DynamoDB, permettant à l'application de les récupérer et de les afficher facilement dans l'interface utilisateur.

## 6. AUTHENTIFICATION & AUTORISATION

✓ Choix : Amazon Cognito

Pourquoi ?

Amazon Cognito est un service qui gère l'inscription, la connexion et la gestion des sessions pour les utilisateurs de votre application. Il s'intègre nativement et facilement avec AWS AppSync pour l'authentification des requêtes API. Cognito permet également d'intégrer des fournisseurs d'identité externes tels que Google, Facebook, ou des systèmes d'entreprise via SAML/OIDC, offrant une flexibilité pour la gestion des identités des freelances et des administrateurs de la plateforme.



**Explication :** Amazon Cognito s'occupe de tout le cycle de vie de l'utilisateur, de l'inscription des nouveaux freelances à la gestion de leurs sessions. Il émet des jetons d'authentification qui sont ensuite utilisés par AppSync pour autoriser l'accès sécurisé aux données et aux opérations de la plateforme.

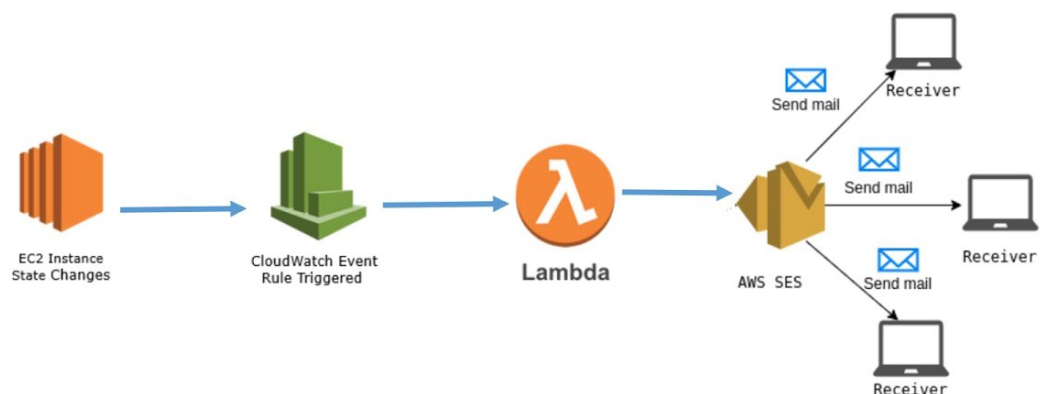
## 7. SUIVI & NOTIFICATIONS

✓ Choix : Amazon CloudWatch + Amazon SES

Pourquoi ?

Amazon CloudWatch est un service de surveillance et d'observabilité qui fournit des données et des aperçus actionnables pour surveiller les applications et les services AWS. Il est essentiel pour suivre les erreurs, les latences et les métriques de performance de la plateforme. Amazon SES (Simple Email Service) est un service d'envoi d'emails hautement évolutif et économique, parfait pour envoyer des emails de confirmation (inscription, mise à jour de profil), des alertes ou des notifications aux freelances et aux administrateurs.

---



**Explication :** Chaque opération significative (création ou mise à jour de profil) peut générer des logs dans CloudWatch pour le suivi et le débogage. De plus, des événements spécifiques peuvent déclencher des fonctions Lambda qui, à leur tour, envoient des notifications par email via Amazon SES, informant les utilisateurs ou les administrateurs des changements importants.





## CONCLUSION

L'architecture proposée s'appuie sur les services AWS gérés, offrant une solution sans serveur, scalable, sécurisée et économique pour la plateforme de gestion des profils freelances. Chaque composant a été choisi pour sa capacité à répondre aux exigences spécifiques du projet, tout en assurant une intégration harmonieuse et une maintenance simplifiée.

Composant	Service AWS Choisi	Raison principale
API	AppSync (GraphQL)	Flexibilité des requêtes, efficacité réseau
Base de données	DynamoDB	NoSQL, haute scalabilité, performance
Authentification	Cognito	Gestion utilisateurs, sécurité intégrée
Fichiers	S3	Stockage d'objets sécurisé, durable, évolutif
Logique métier	Lambda	Sans serveur, coûts faibles, événementiel
Emails	SES	Notifications par email fiables et économiques
Monitoring	CloudWatch	Suivi des performances et des logs, alertes