

Arcanio Development

Prestation fullstack.



1. BESOIN

2. SOLUTION

3. CONCEPTION

4. ARCHITECTURE

5. CLOUD

6. SÉCURITÉ

7. CHRONOLOGIE

8. MÉTHODOLOGIE

9. TARIF

10. RÉFÉRENCES

BESOIN

Midgard va créer pour ses clients un système d'analyse du risque via une plateforme en ligne.



01

Mettre à disposition des clients une application web permettant de manipuler les données.

02

Pouvoir mettre à jour l'application avec de nouveaux modules et plus de fonctionnalité.

03

Une architecture permettant de traiter différents flux ainsi que faire tourner les services.

01

Le frontend de l'application sera en HTML/JS. Il permettra de gérer l'utilisation des données et des modules par les utilisateurs.

Les modules seront la manipulation d'éléments géométriques sur fond de carte, l'affichage et l'ajout de nouveaux fichiers aux formats spécifiques.

02

Le backend sera en Laravel et aura accès aux différentes API de stockage ainsi que de calcul.

Cette technologies éprouvée permet de gérer un grand nombre d'information tout en proposant de hautes performances.

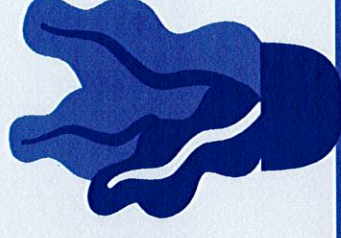
03

Une architecture data-lake pour le traitement des données et des algorithmes.

Cette approche permet de répondre au cahier des charges et de préparer Midgard à la montée en charge en suivant les dernières pratiques du cloud.

Solution

Une plateforme créée à partir des dernières technologies web, hébergée sur AWS.



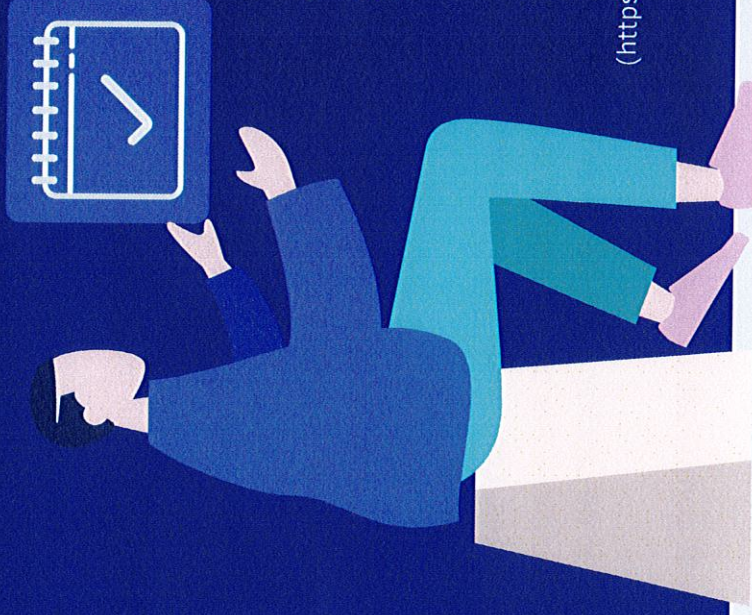
Conception

Nous reviendrons sur votre cahier des charges afin d'affiner notre maquette.

Nous sommes à même d'intégrer les différentes vues métiers (carte interactives, modèle 3D, rapport) présentes sur la maquette. Elle servira également pour le parcours utilisateur et la structure.

Les fonctions métiers du backend seront utilisées via l'API OpenDroneMap et celle de stockage. Les cartes seront rendues par un serveur WMS

Nous travaillons en respectant les meilleures pratiques web (https, responsive, caching, documentation complète, encryption de bout en bout...)



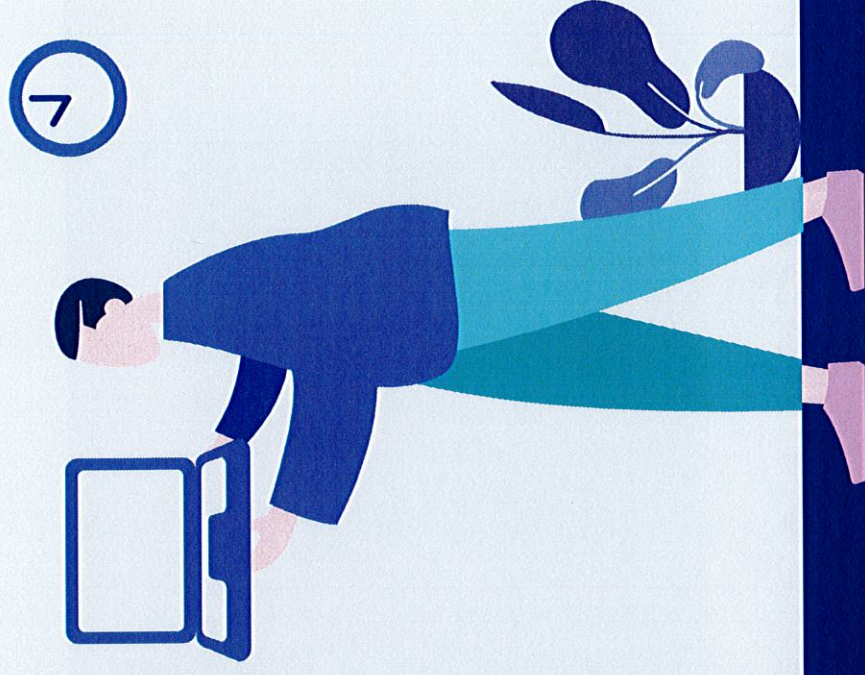
L'architecture

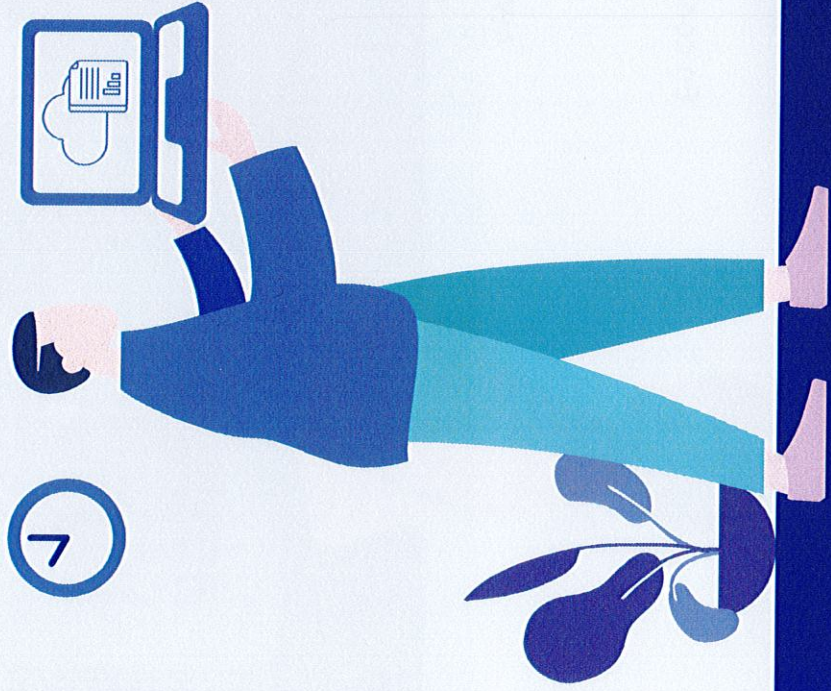
Elle est composée des parties suivantes:

Un lac de stockage afin de gérer les fichiers particuliers. Par son principe il uniformise les données, comme un entrepôt avec des étagères modulaires.

Son accès et relations (vol, missions, clients) seront gérées par une API Kubernetes en Laravel.

La solution Kubernetes sépare l'application en container afin de pouvoir isoler les problèmes, minimiser les coûts et simplifier le management.





Le cloud

L'application web sert d'interface avec les différents composants de l'architecture.

Elle est accompagnée d'un serveur WMS afin de réaliser sur les cartes les opérations demandées par le site web.

Un ensemble d'unité serverless gère les tâches répétitives (tri, tag, processus léger...) sous forme de containers.

L'autre groupe de serveur est au contraire de hautes performances sur demande pour les gros calculs et l'entraînement des algorithmes.

La sécurité

En conséquence de la type de clientèle, d'importantes mesures sont prises.

Le principe des droits minimums est en place et tous les accès restreints ainsi qu'enregistrés.

AWS propose des outils adéquats et agnostiques afin de les assurer.



Chaque échange entre les serveurs et avec internet sont encrypté par la dernière technologies adaptée (HTTPS, AES-256, 3DES).

Ces services couvrent aussi les fichiers, les décryptant automatiquement pour une entité autorisée par l'API de stockage.

Plusieurs alarmes envoient des notifications (Slack, SMS, mail) selon des règles très précise, comme un pic d'activité ou un comportement inhabituel, s'appuyant sur les enregistrements.

Le prestataire cloud fournit une disponibilité de 99.99%/an et 99,99+% de durabilité. Ces taux, de loin les plus hauts, garantissent accessibilité et la pérennité de l'infrastructure.

Chronologie

Il y aura une réunion au début, en cours, et à la fin de chaque étape.

Un prototype sera utilisable à la fin de la première partie, puis une version par semaine.

Un état des lieux peut être fait à tout moment.



Architecture

- 1 mois, 02/03 - 23/03
- 3 semaines création
 - 1 semaine prise en main (tout du long)



Frontend

- 1 mois, 23/03 - 20/04
- 1 semaine UI
 - 2 semaines métier
 - 1 semaine de retours

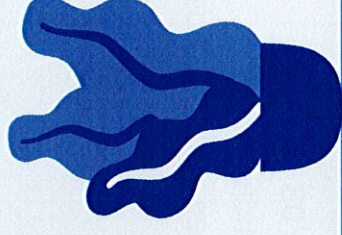


Backend

- 1 mois, 20/04 - 18/05
- 2 semaines gestion utilisateurs
 - 2 semaines fonctionnalités

Méthodologie

- Afin de garantir des échanges productifs et rapides vous serez invité sur notre serveur Slack.
En plus des réunions programmées un compte rendu des avancées, problèmes rencontrés et questions sera réalisé chaque semaine.
- Un schéma de la logique des différentes parties sera fait au début de la prestation, assurant de partir dans la bonne direction.
- Notre approche permet d'effectuer des tests durant le développement de façon automatique, réduisant le temps de correction.
En parallèle les serveurs sont capables de détecter une erreur et de se reconstruire seuls.
- Midgard sera seul propriétaire du résultat de la prestation.
Nous nous engageons également à veiller à la non-reproduction du projet.



4000€

Site web frontend

200€ * 20 j/h

4000€

Site web backend

200€ * 20 j/h

8000€

Architecture

400€ * 20 j/h

Période de garantie

3 mois

/// Bon pour accord,

Le 26 février 2020

Incubateur d'entreprises innovantes

in2i3

Maison du Parc - ZI Erbaiglo - 29600 BASTIA

Siret : 798 432 087 00010

Tarif

Le prix d'hébergement est appliqué à l'utilisation par AWS.

TVA non applicable, article 293B du CGI.



L'agence

arcanio.dev



**MAXIME
DZIERWA**

DÉVELOPPEUR FULLSTACK

+33 6 28 74 98 76

maxime.dzierwa@gmail.com



**DAMIEN
GRANDI**

DÉVELOPPEUR FULLSTACK

+33 6 84 15 15 85

damien.grandi@gmail.com

L'agence

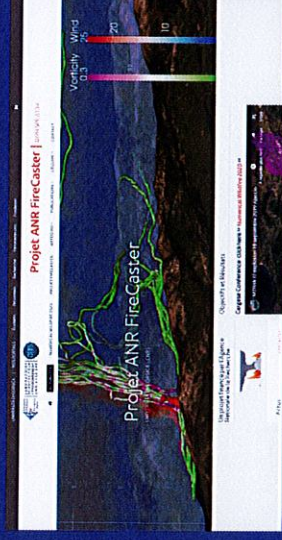
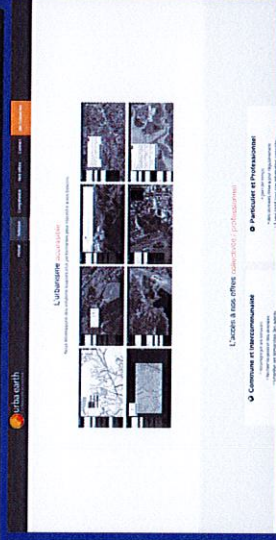
Nos références

weboom.fr

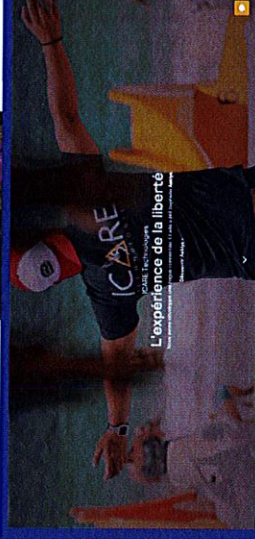


weboom.fr

urba-earth.com



firecaster.universita.corsica



icaretechnologies.com
API et infrastructure

