

Veggo Controle Center

Kubernetes



by VERMEG

Build it and drive the change !

Réalisé par : Adem BESBES

Devant le jury composé de :

Présidente: Mme. Mariem GZARA

Membre: Mme. Raghda JOUIROU

Encadrante: Mme. Sana BENZARTI

Annee Universitaire: 2024 - 2025



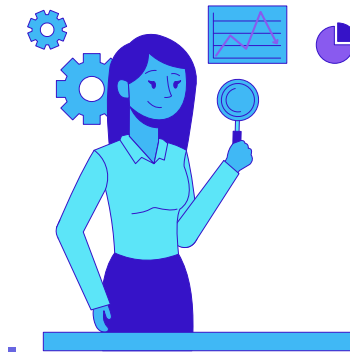
PLAN

01



Contexte et objectifs du projet

02



Analyse et spécification des besoins

03



Conception

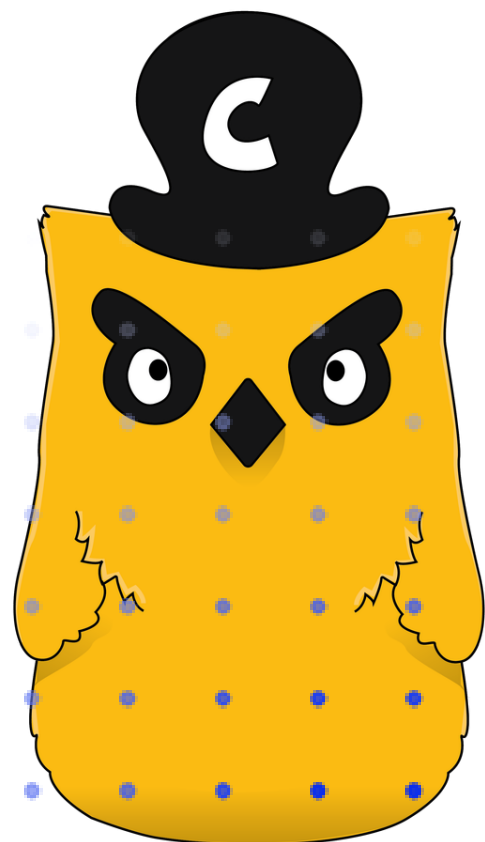
04

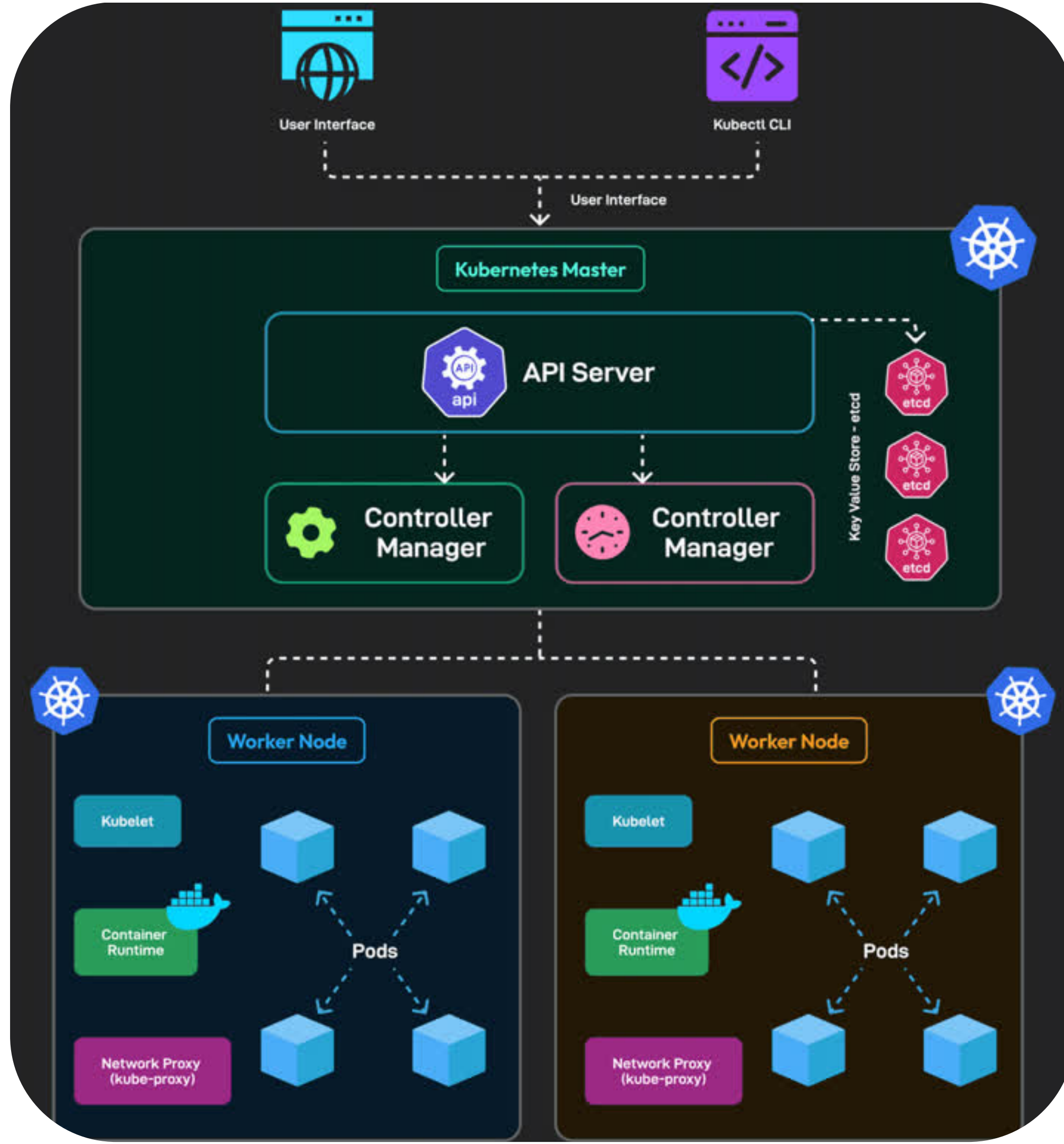


Réalisation

01

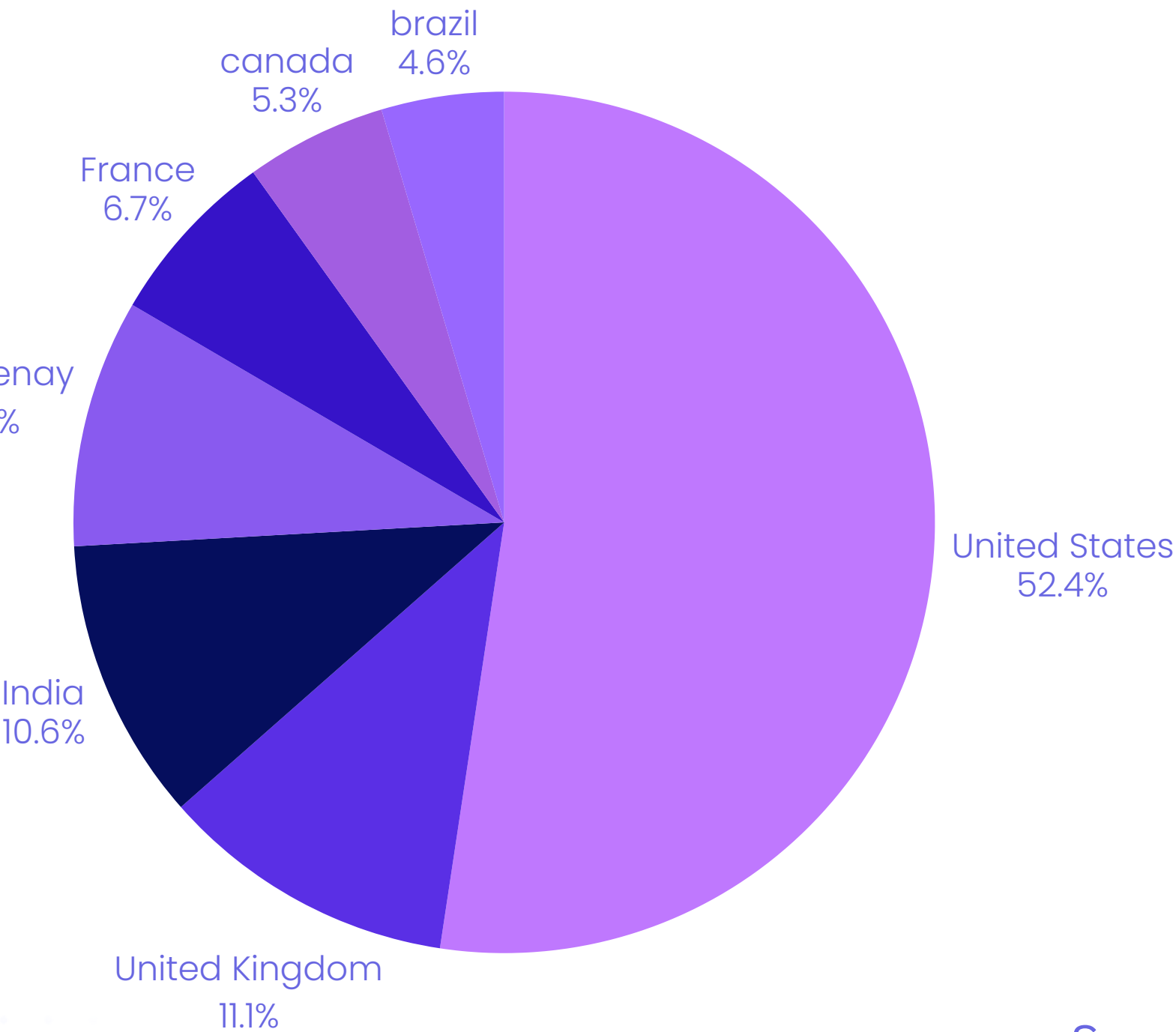
CONTEXTE ET OBJECTIFS DU PROJET





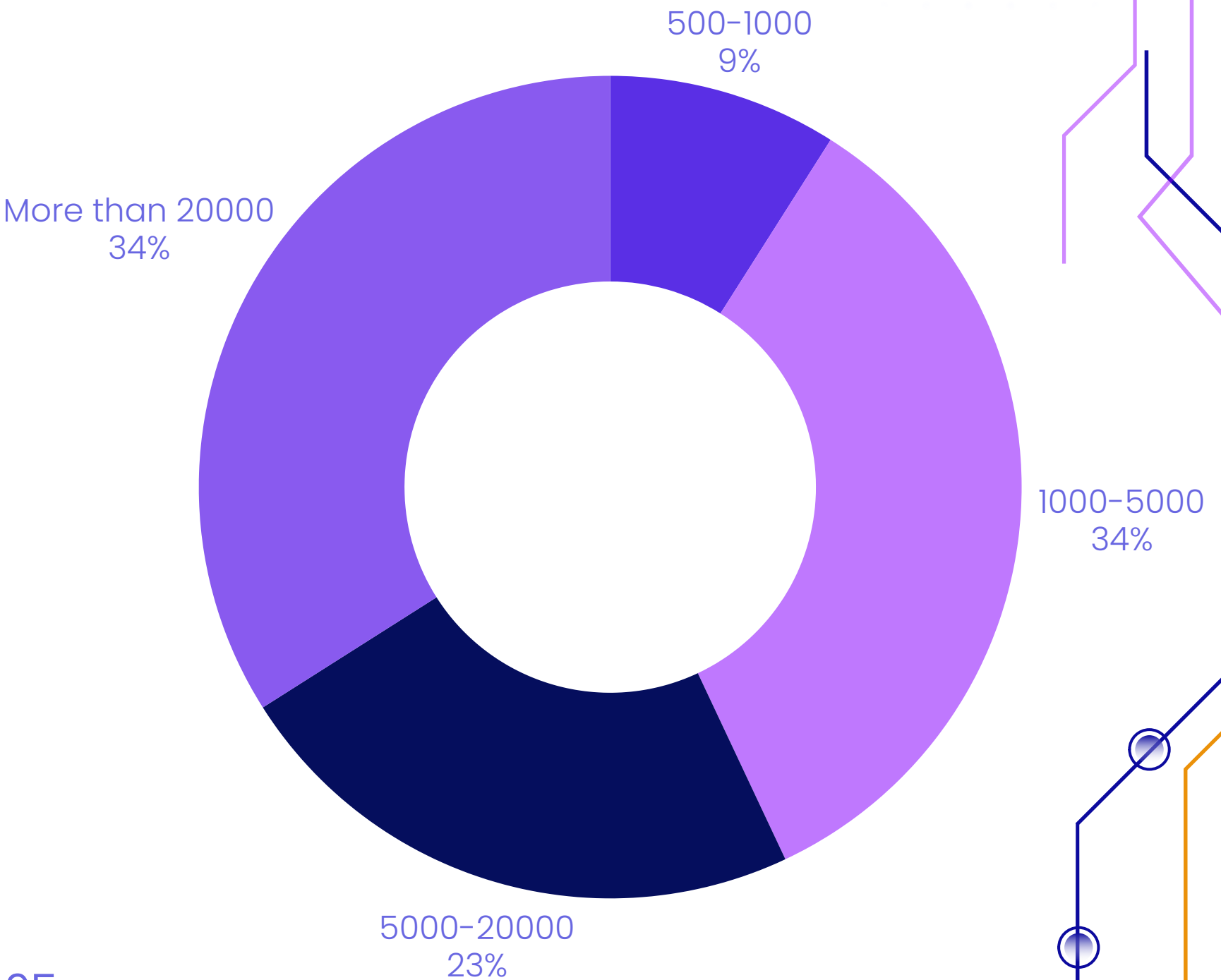
Kubernetes

KUBERNETES EST ADOPTÉ PAR 60 % DES
ENTREPRISES À L'ÉCHELLE MONDIALE



Source: CNCF
2024

TAILLE DE L'ENTREPRISE (NOMBRE D'EMPLOYÉS)



Context et

Analyse et

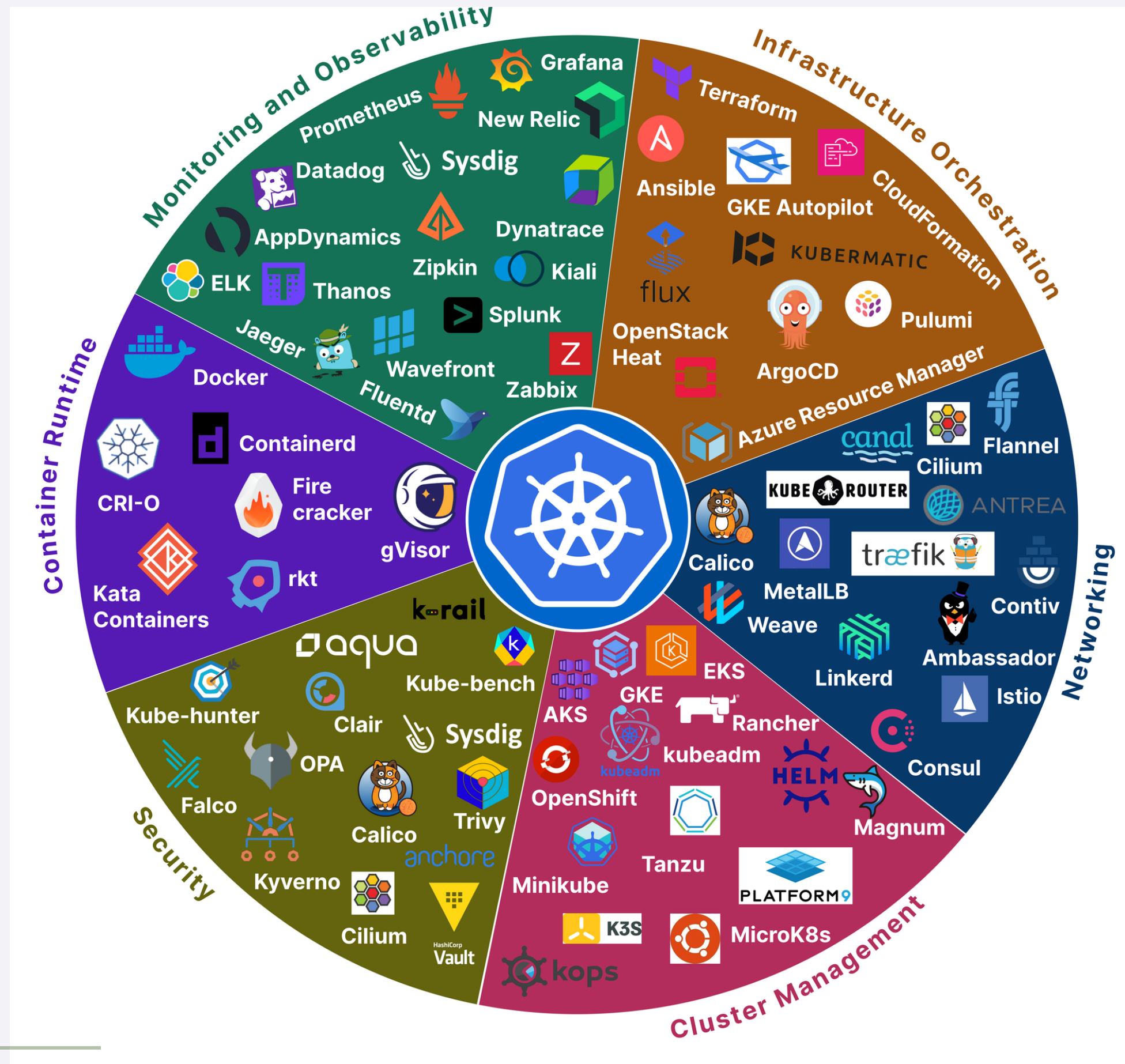
Conception

Réalisation

Objectif Projet

Specification des Besoins

ROUE DES OUTILS DE L'ÉCOSYSTÈME KUBERNETES



CERTIFICATION KUBERNETES



Context et

Analyse et

Conception

Réalisation

Objectif Projet

Specification des Besoins

OBJECTIF DU PROJET

Développer une application web intuitive destinée à optimiser l'organisation et la gestion de l'environnement Kubernetes au sein des entreprises.

Cette solution vise à simplifier la configuration, le déploiement et le suivi des applications conteneurisées (basées sur Docker), à travers une interface utilisateur centralisée.

Elle permettra une gestion unifiée, efficace et automatisée des ressources au sein d'un cluster Kubernetes, tout en améliorant la productivité des équipes DevOps et en réduisant la complexité opérationnelle.



Context et

Analyse et

Conception

Réalisation

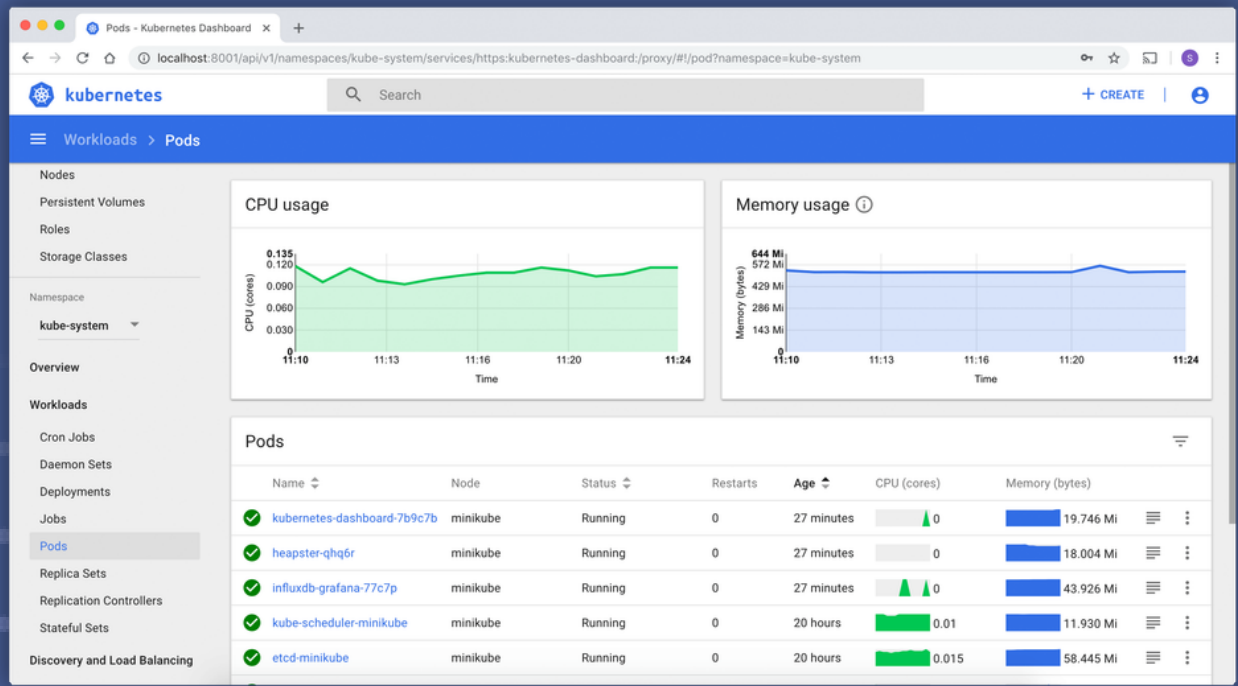
Objectif Projet

Specification des Besoins

Etude de l'existant

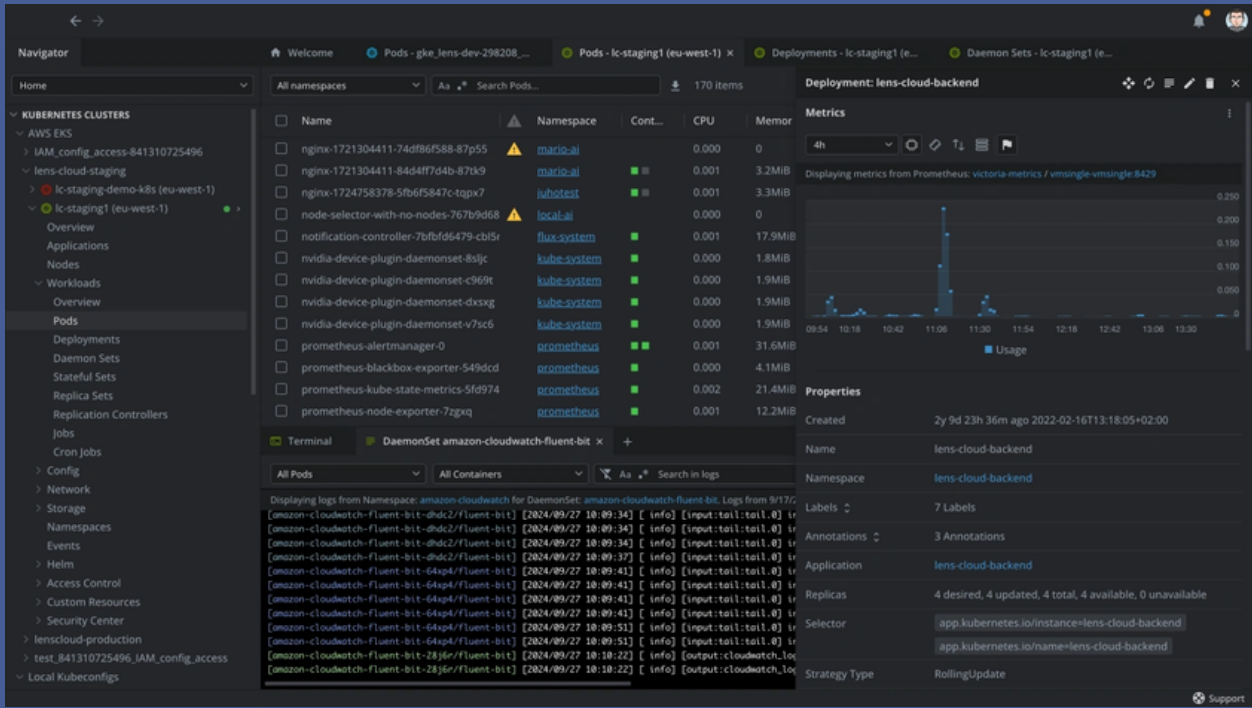
KUBERNETES DASHBOARD

Interface web officielle fournie par la CNCF, simple mais limitée en fonctionnalités avancées de gestion multi-utilisateurs ou de monitoring étendu.



LENS

Application de bureau complète qui propose une visualisation avancée des ressources Kubernetes, principalement destinée aux développeurs. Elle offre une gestion centralisée des équipes à l'échelle d'un seul cluster.



SOLUTION PROPOSÉE

Le projet introduit la notion de Teamspace, un concept innovant visant à segmenter logiquement un cluster Kubernetes en plusieurs espaces de travail indépendants, dédiés à différentes équipes.

Cette approche permet :

- Une organisation plus structurée du cluster
- Une collaboration sécurisée entre les équipes
- Une isolation claire des environnements de travail
- Le maintien d'une architecture centralisée et cohérente à l'échelle d'un seul cluster

Chaque Teamspace dispose de :

- Ses propres ressources (namespaces, quotas, services, etc.)
- Des droits d'accès personnalisés
- Une configuration adaptée aux besoins de l'équipe

Context et

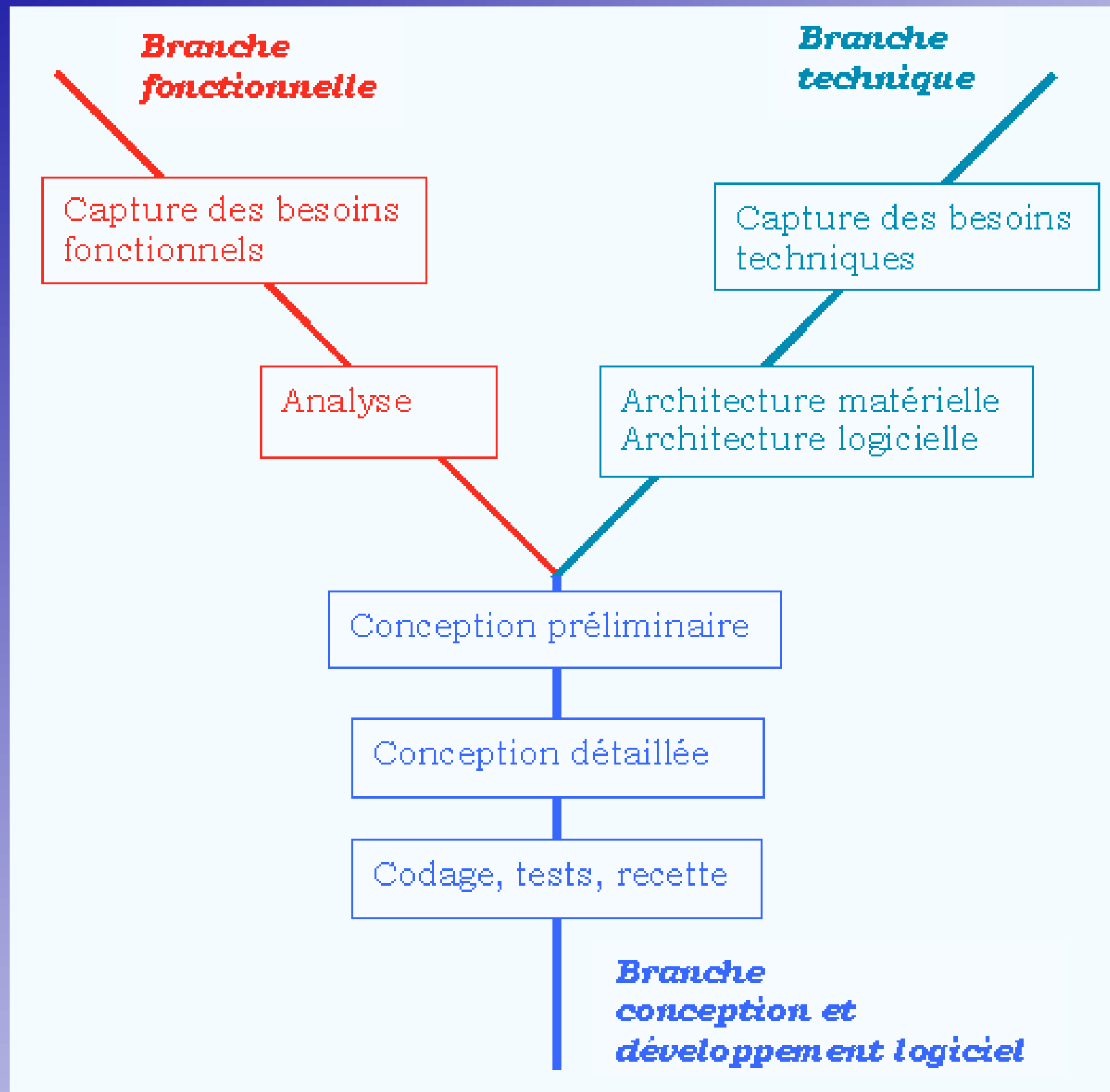
Analyse et

Conception

Réalisation

Objectif Projet

Specification des Besoins

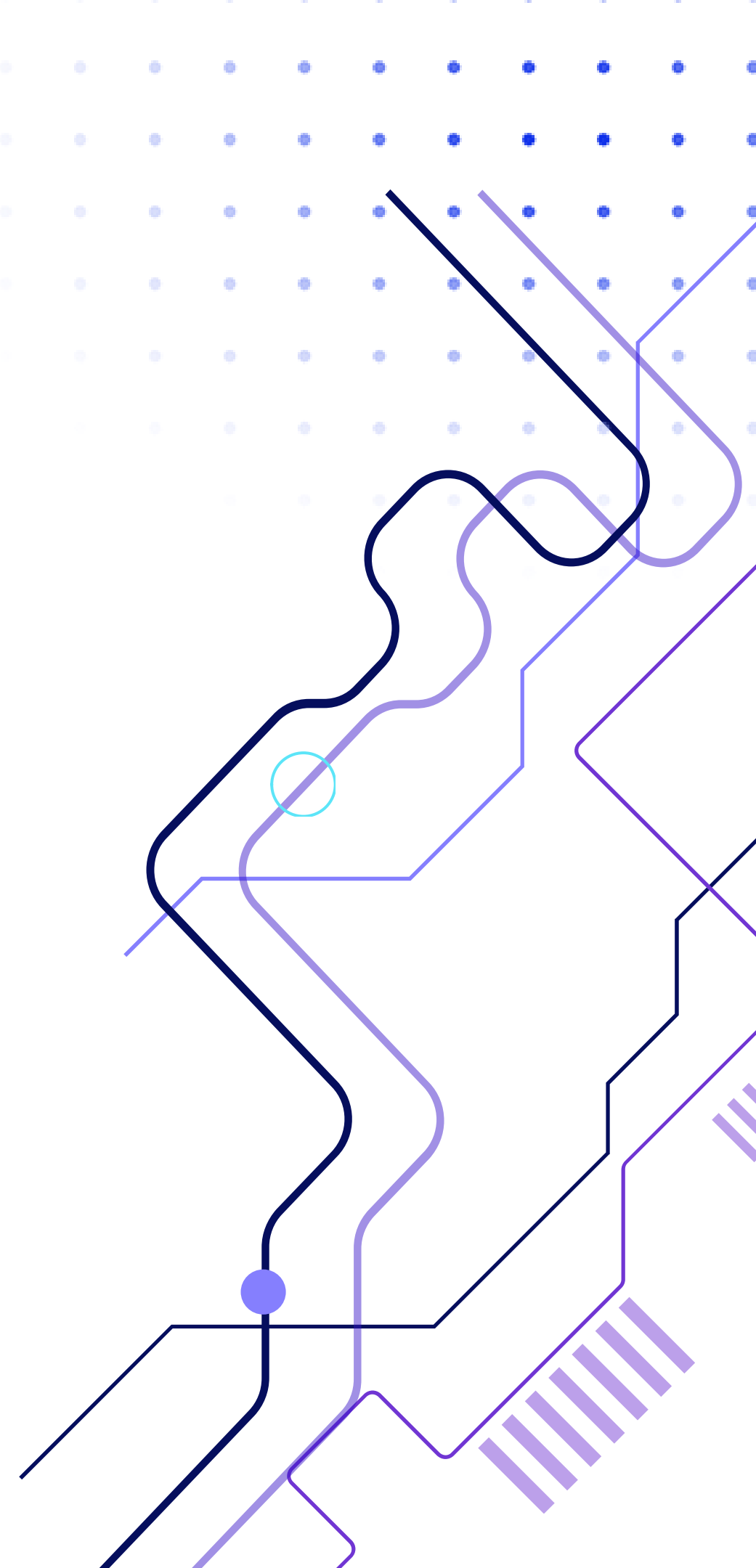


MÉTHODOLOGIE DU TRAVAIL

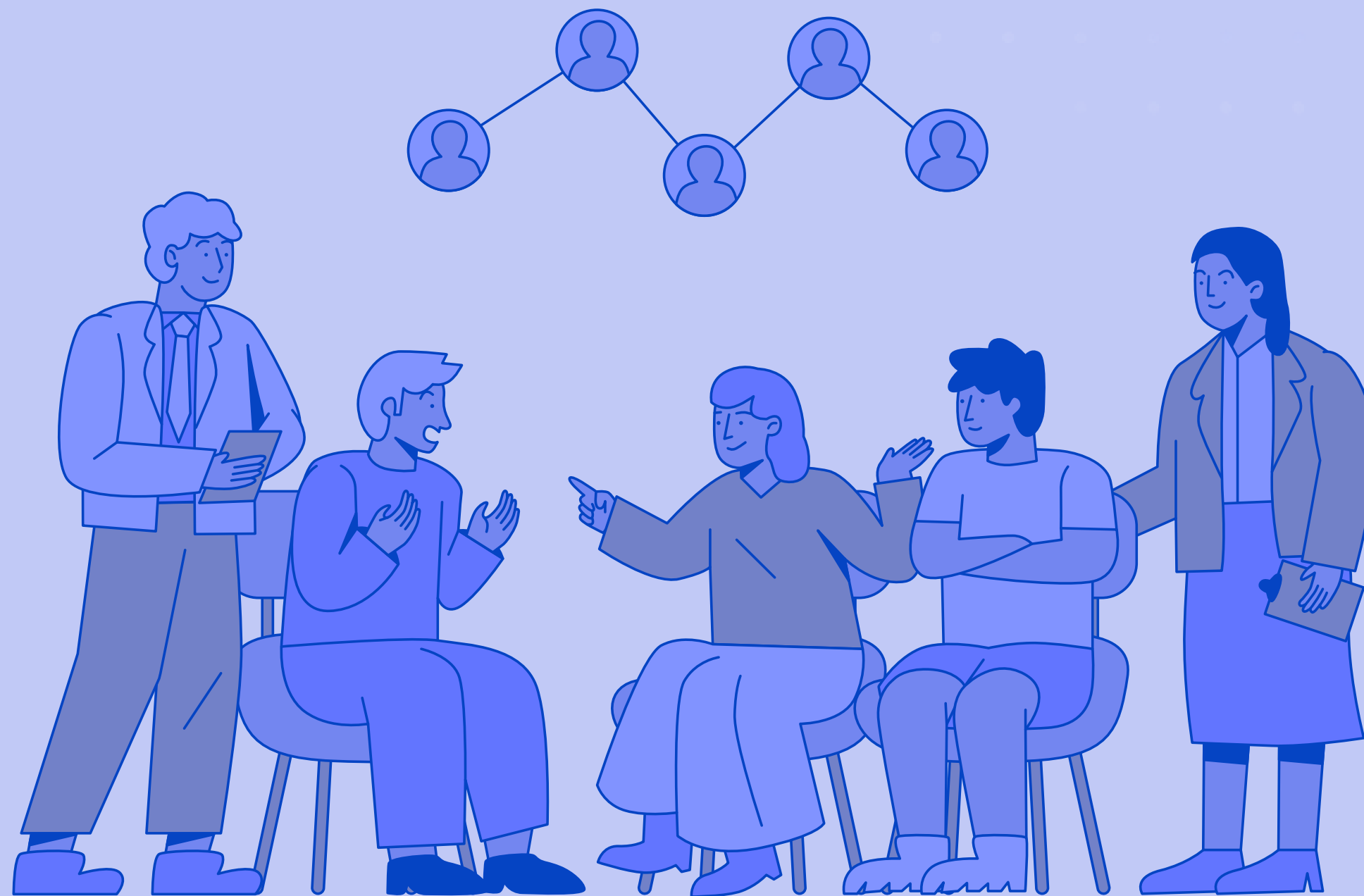
Two Tracks Unified Process

02

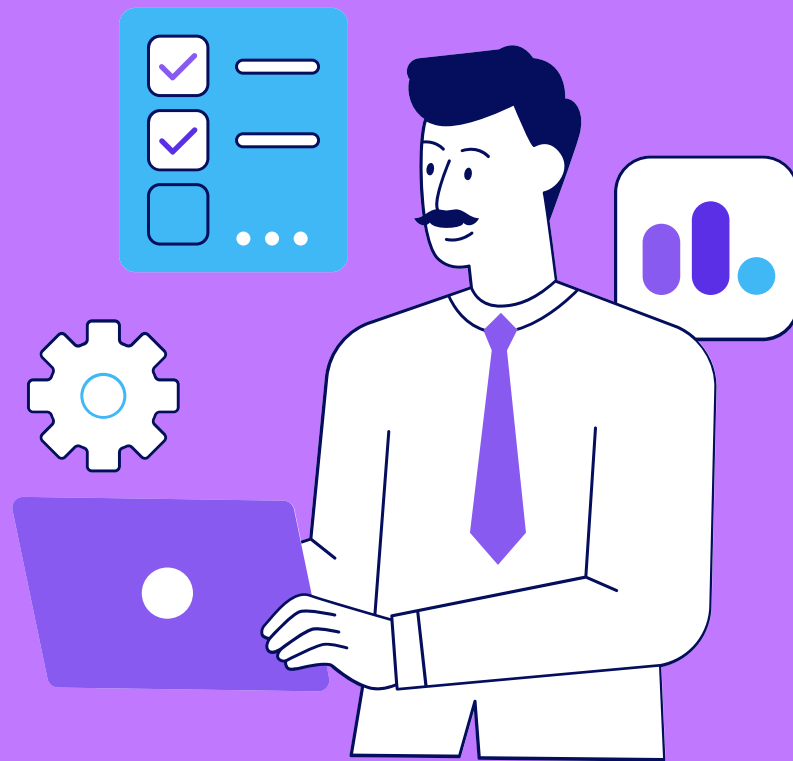
ANALYSE ET SPÉCIFICATION DES BESOINS



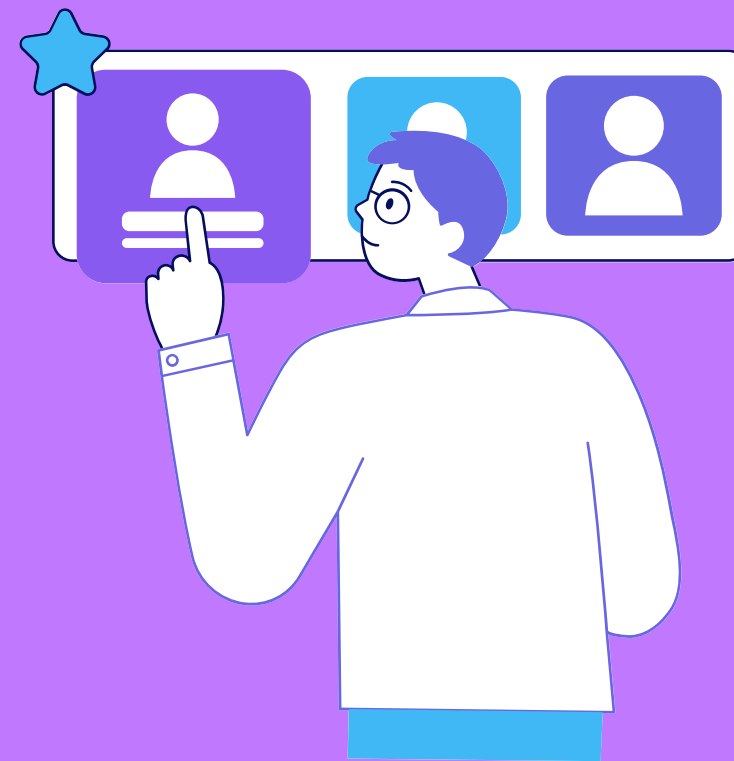
LES BESOINS FONCTIONNELS



LES ACTEURS :



Administrateur



Chef d'équipe



Développeur

Context et

Analyse et

Conception

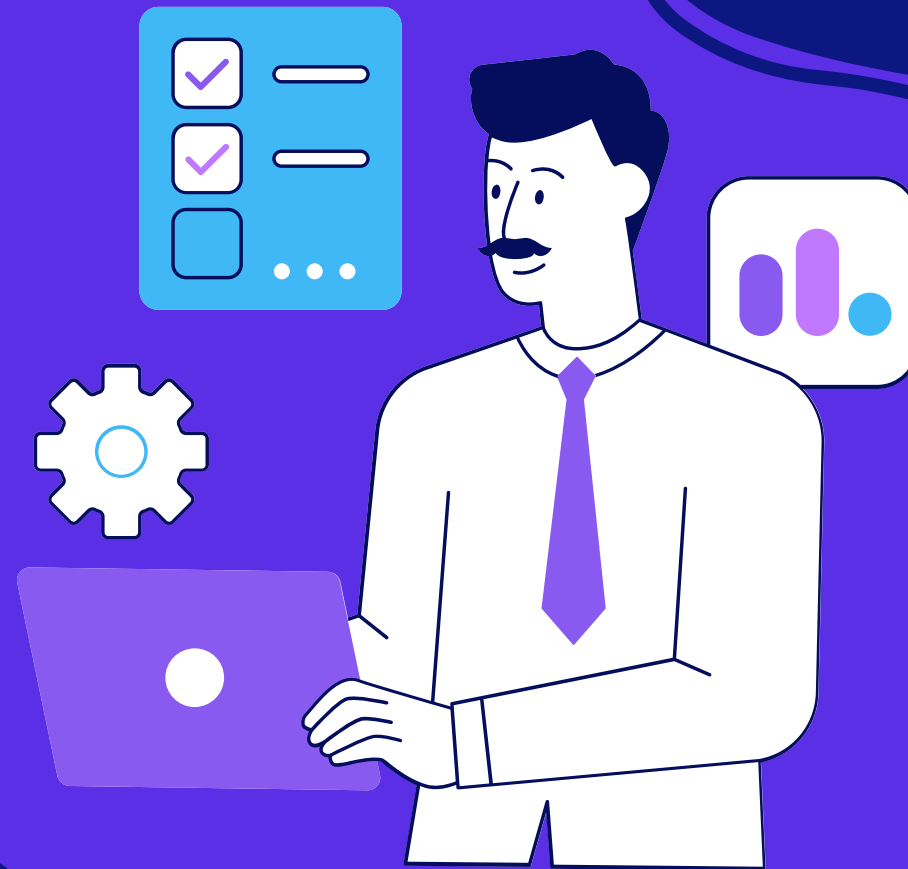
Réalisation

Objectif Projet

Specification des Besoins

Gérer Teamspace

Diriger configuration du Serveur Prometheus



Gérer infrastructure Kubernetes

Gérer configuration Grafana

Administrateur

Context et

Analyse et

Conception

Réalisation

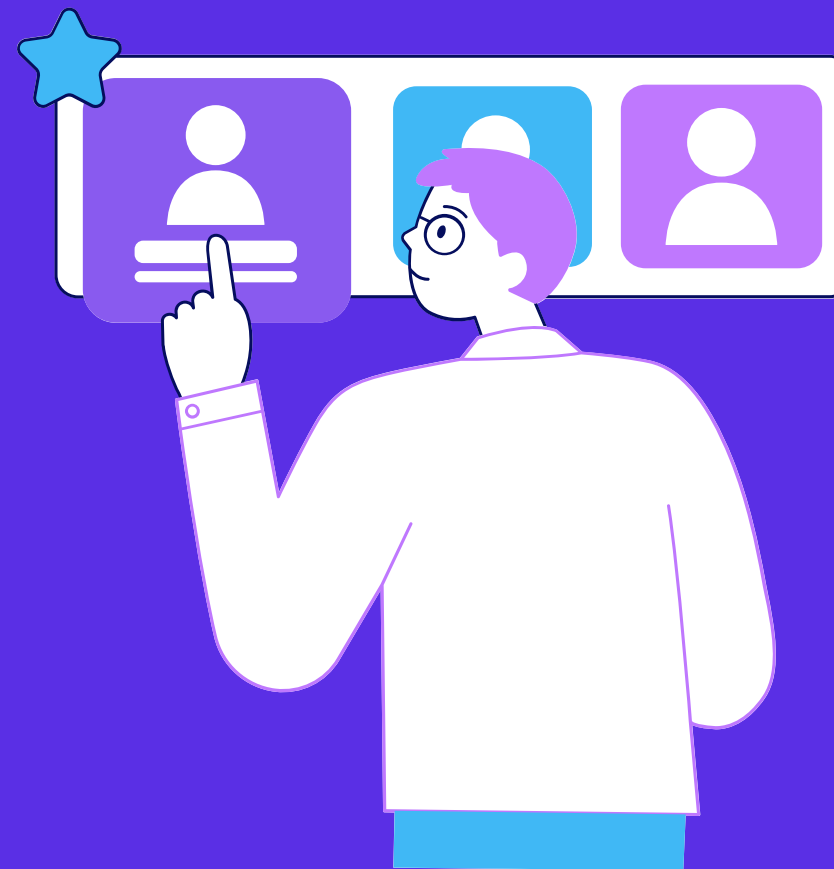
Objectif Porjet

Specification des Besoins

Gérer Namespaces

Scanner Sécurité d'application

Gérer Teamspace



Chef d'équipe

Context et

Analyse et

Conception

Réalisation

Objectif Porjet

Specification des Besoins

Surveillance système

Créer teamSpace



Développeur

Gérer ressources kubernetes

Rejoindre teamSpace

Context et

Analyse et

Conception

Réalisation

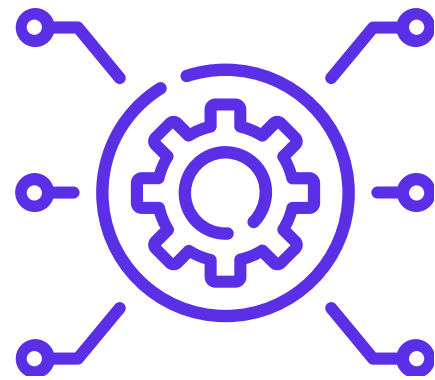
Objectif Porjet

Specification des Besoins

LES BESOINS NON FONCTIONNELS

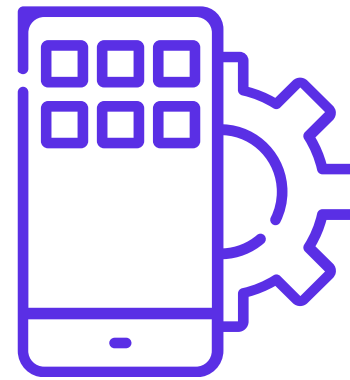


BESOINS NON-FONCTIONNELS



Performance :

gestion de charge
simultanée, réponse rapide,
haute disponibilité, système
robuste.



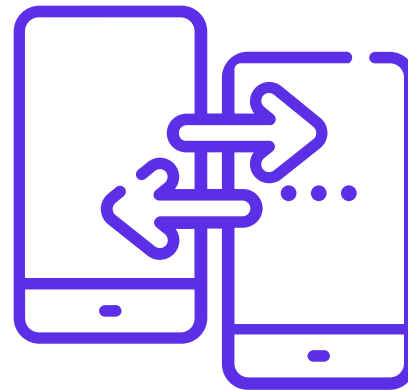
Caching :

optimiser performances,
réduire chargement,
limiter requêtes.



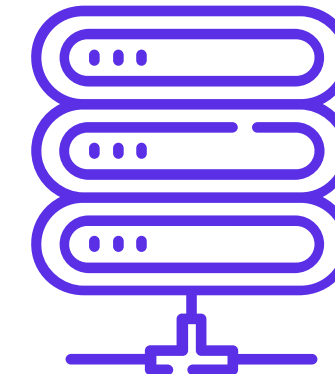
Sécurité :

confidentialité,
protection, intégrité,
prévention, fiabilité



Fiabilité :

disponibilité continue,
expérience fiable,
cohérence assurée.



Evolutivité :

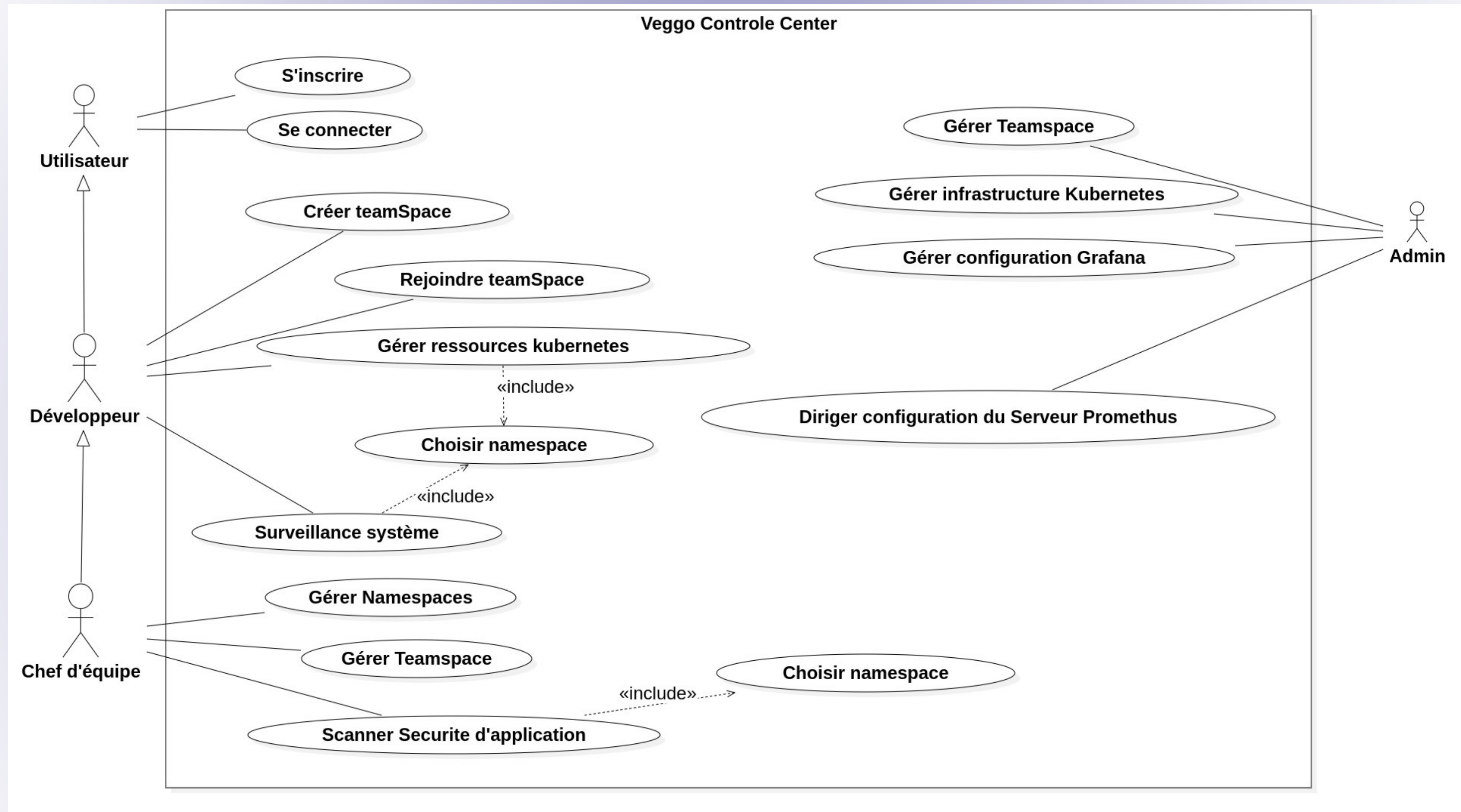
adaptation efficace à la
croissance utilisateurs
et données.

03

Conception



Cas d'utilisation générale



Context et

Analyse et

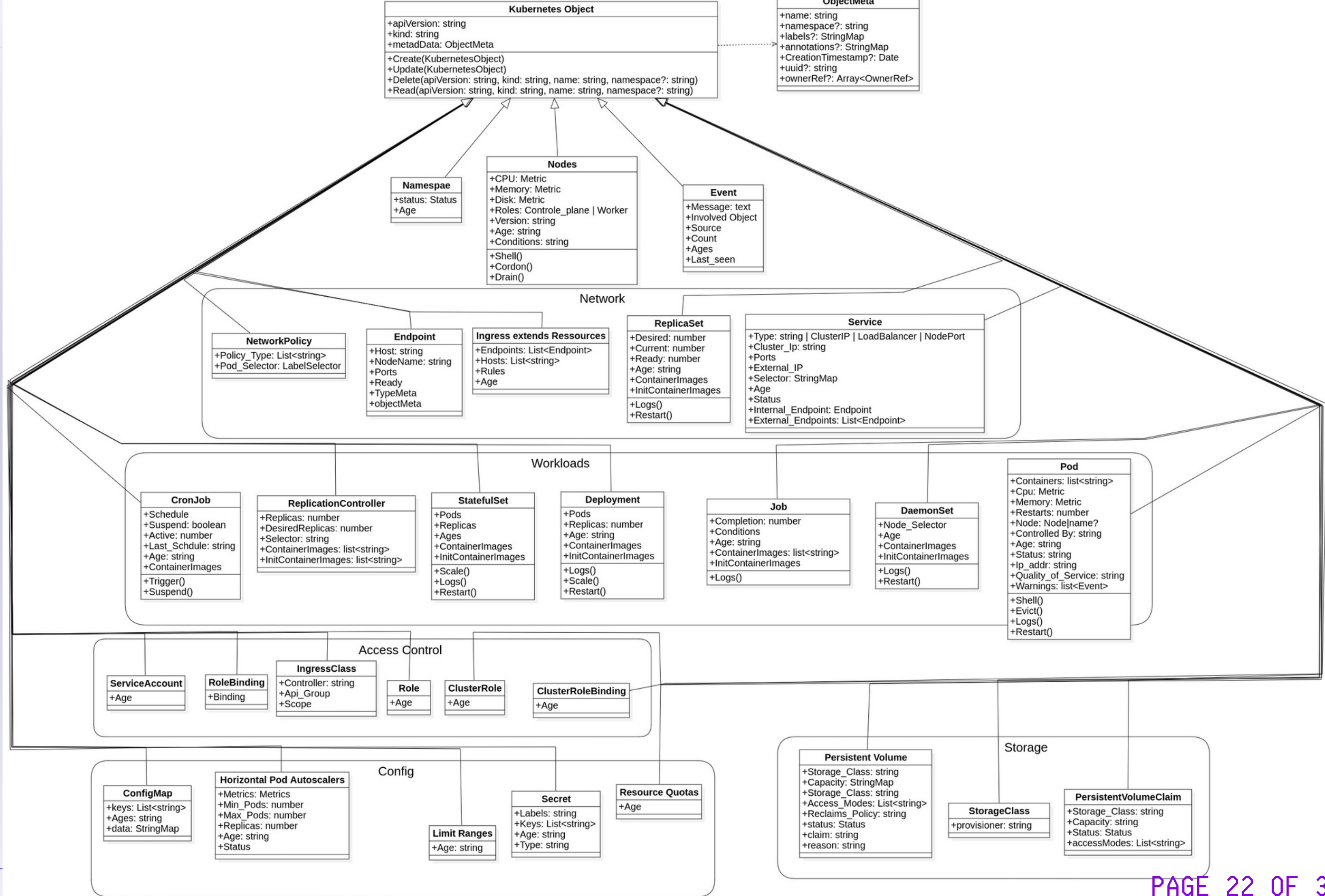
Conception

Réalisation

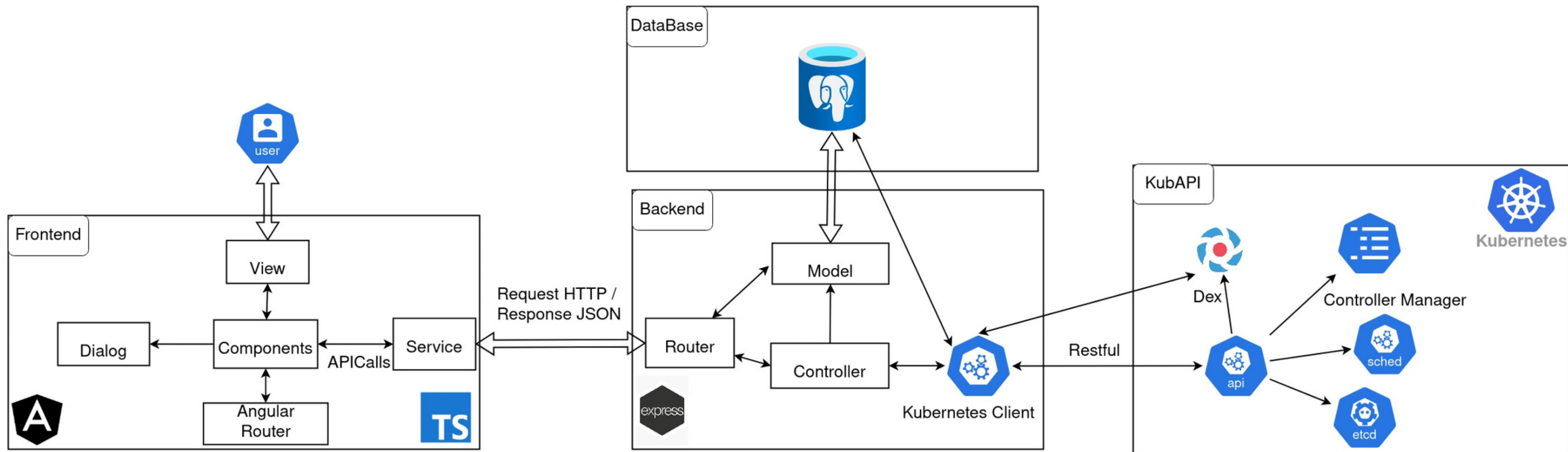
Objectif Projet

Specification des Besoins

Diagramme du classe de ressources Kubernetes



Architecture de l'application



Context et

Analyse et

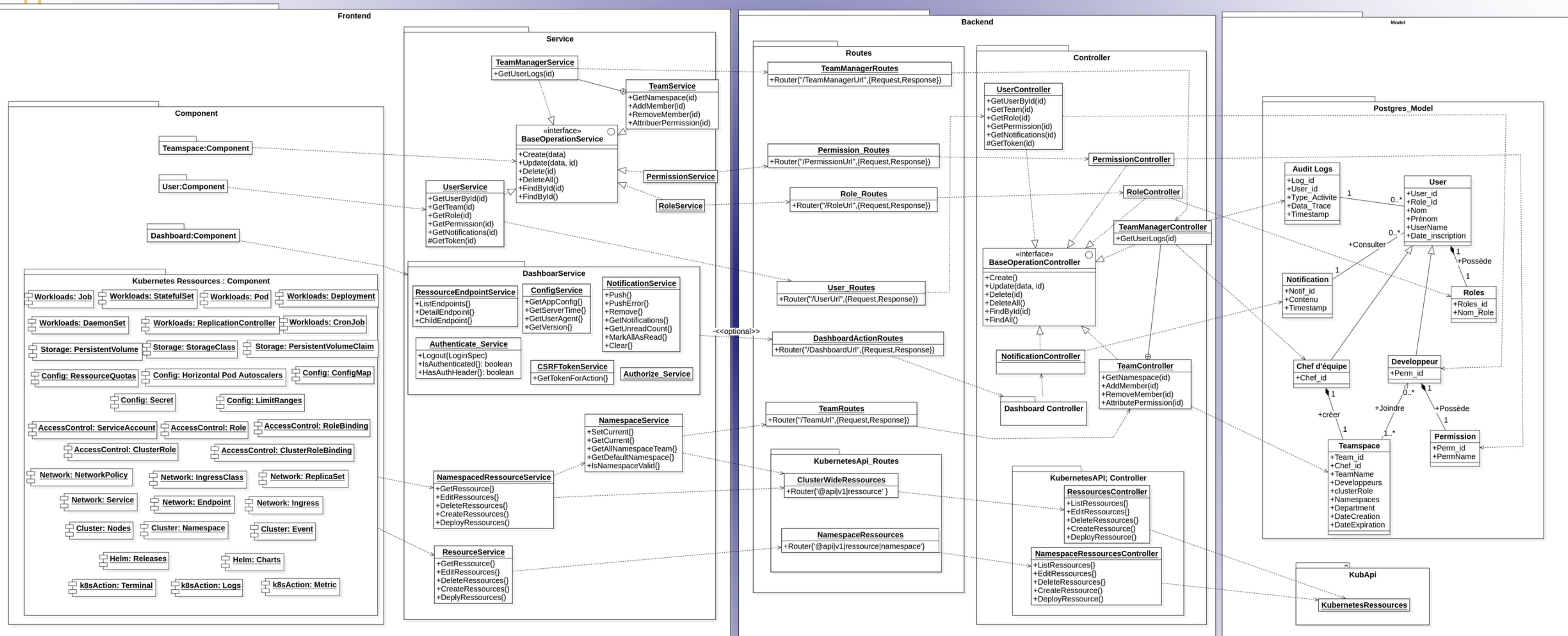
Conception

Réalisation

Objectif Projet

Specification des Besoins

Architecture de l'application



Context et

Analyse et

Conception

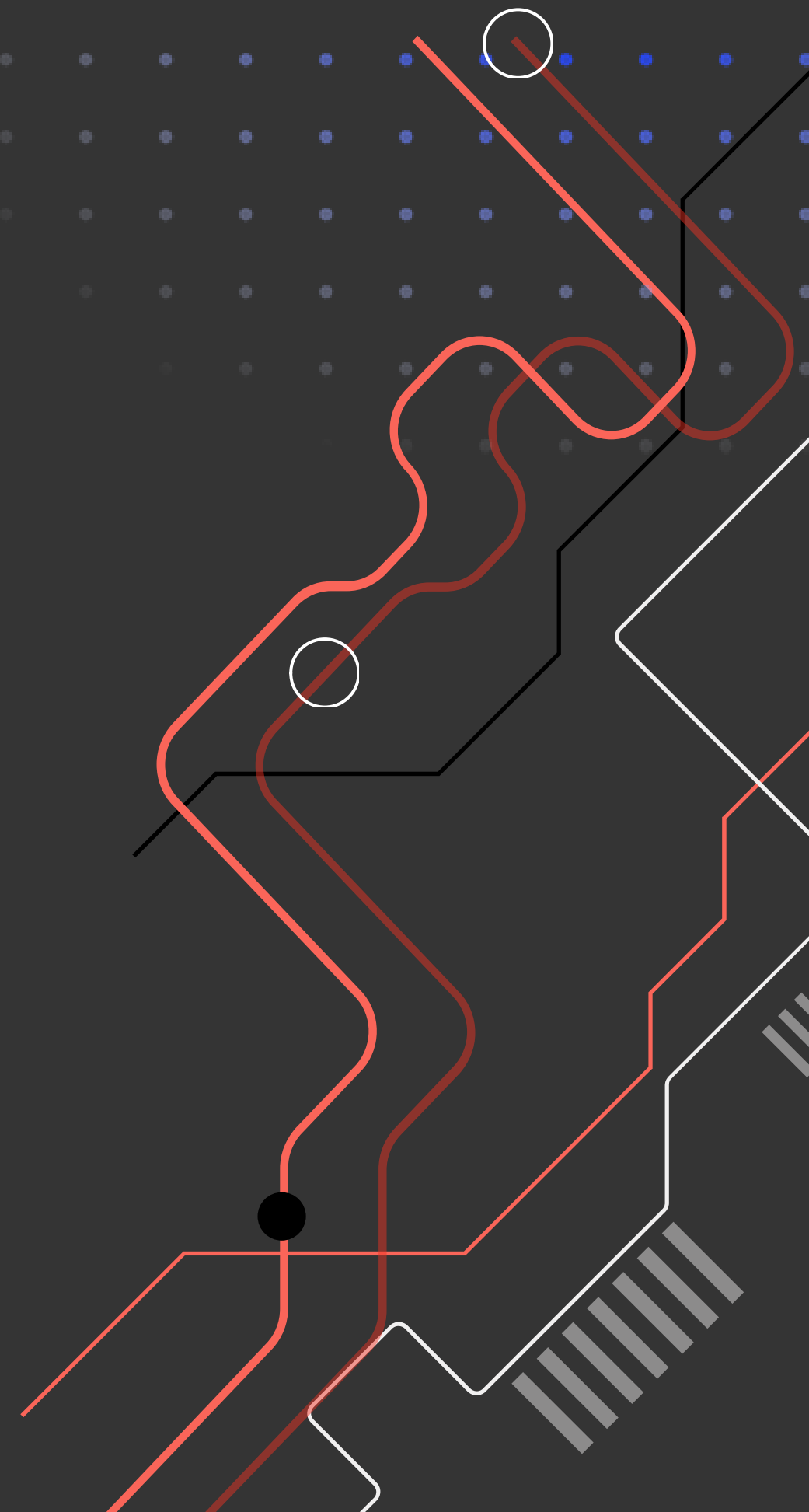
Réalisation

Objectif Projet

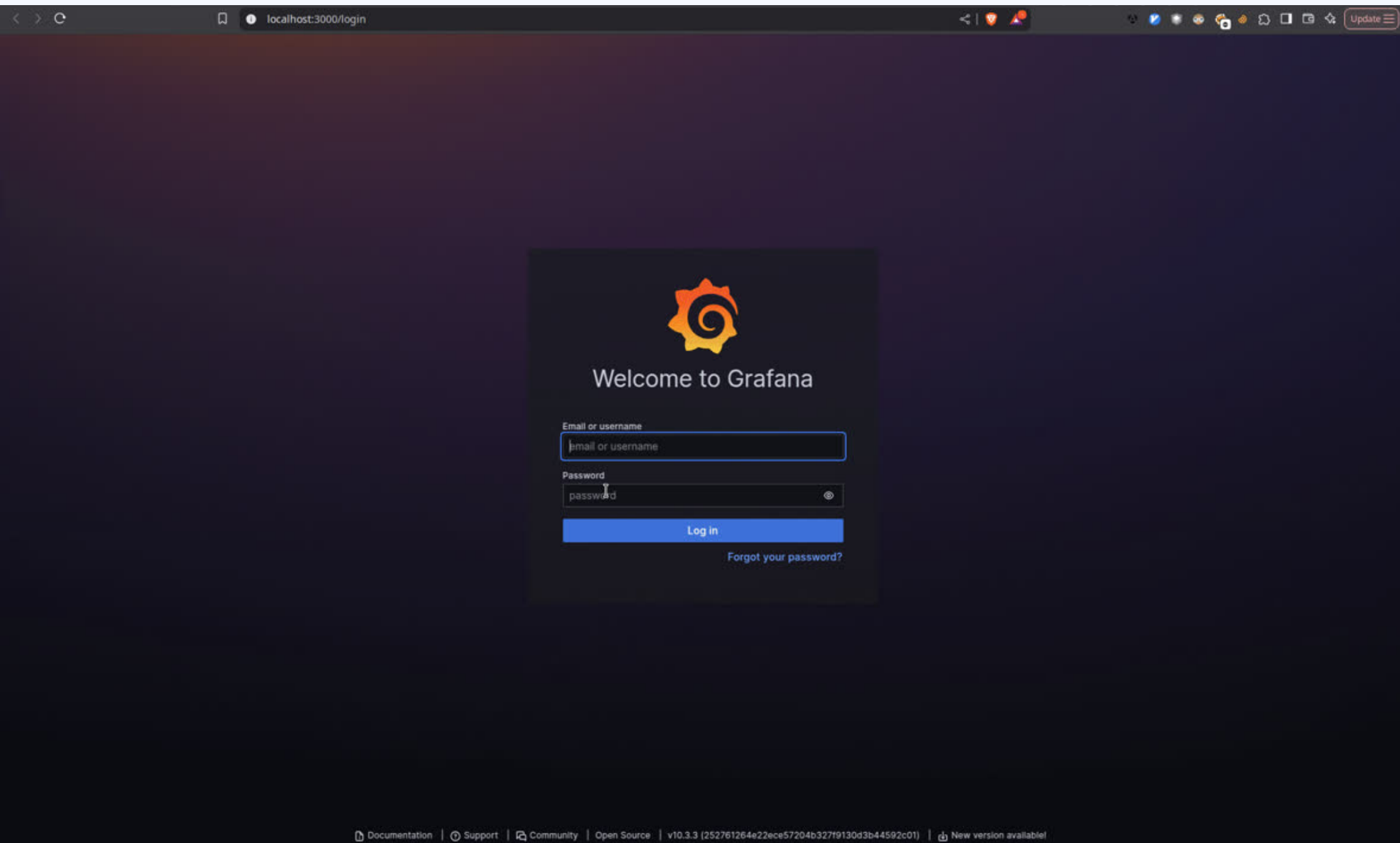
Specification des Besoins

04

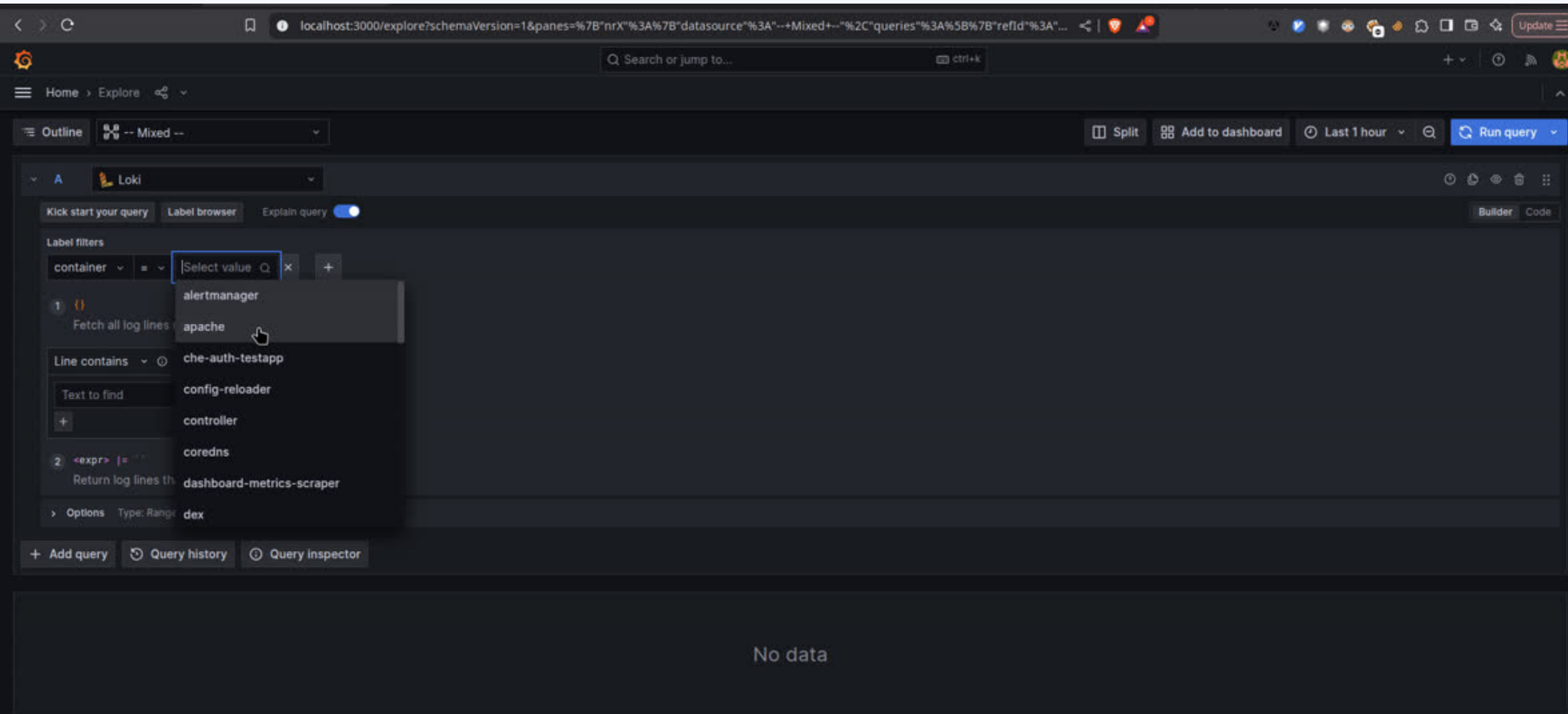
Réalisation



Interface Grafana pour l'administrateur



Interface Grafana pour l'administrateur



Interface Prometheus pour l'administrateur

The screenshot displays the Prometheus web interface in a browser window. The address bar shows 'localhost:4200/prometheus'. The interface has a dark theme. At the top, there's a navigation bar with 'Prometheus Server' and a user profile 'AN AnalysisTeam'. Below this is a sub-navigation bar with 'Prometheus', 'Query', 'Alerts', and 'Status'. The 'Query' tab is active. A query editor contains the query: `>_ rate(container_cpu_usage_seconds_total{image!=""}) [5m]`. Below the query editor are tabs for 'Table', 'Graph', and 'Explain'. The 'Table' tab is selected, showing an 'Evaluation time' header and several empty rows. A '+ Add query' button is visible at the bottom left of the query area.

Context et

Analyse et

Conception

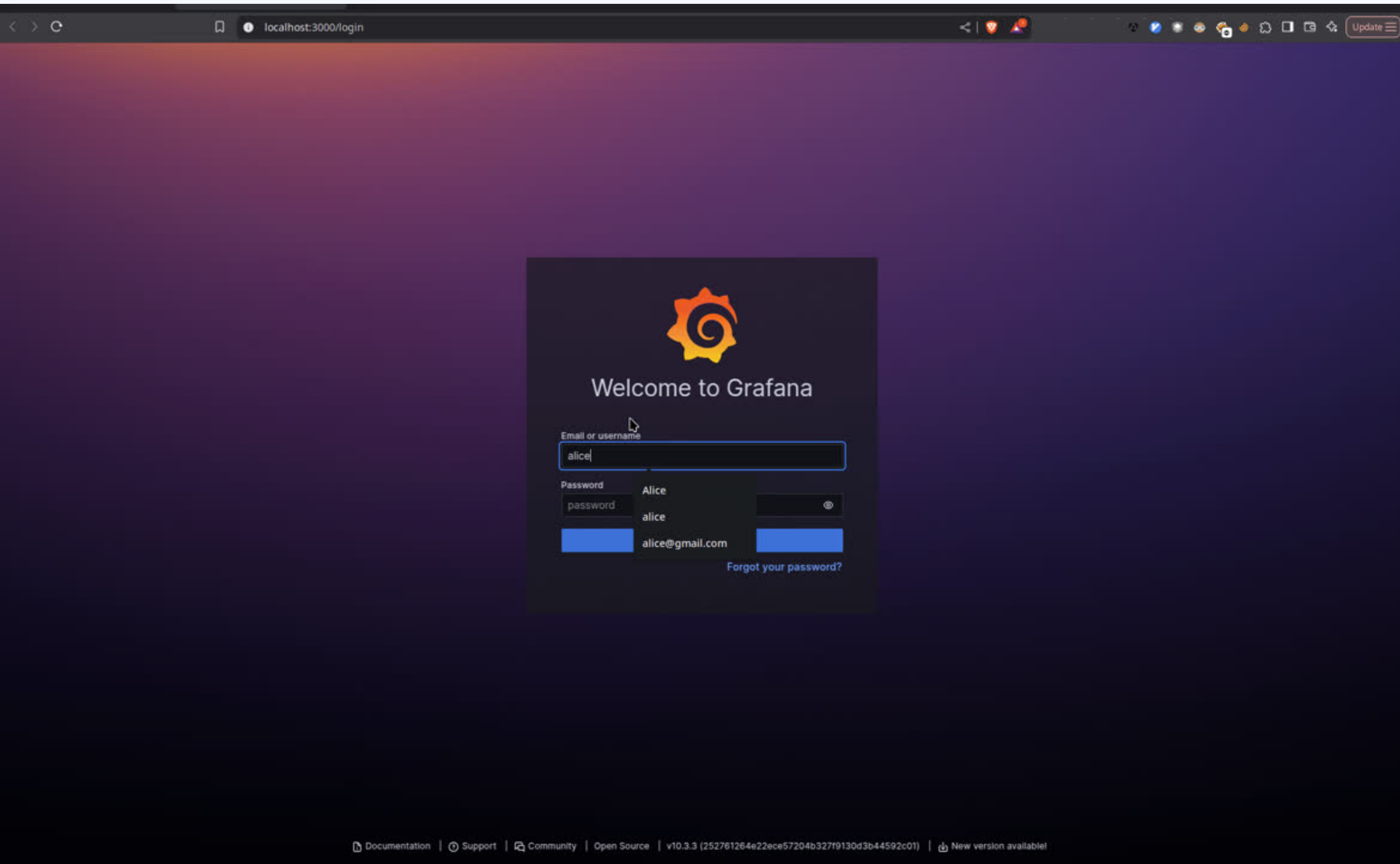
Réalisation

Objectif Projet

Specification des Besoins

PAGE 28 OF 38

Interface Grafana pour le chef d'équipe



Context et

Analyse et

Conception

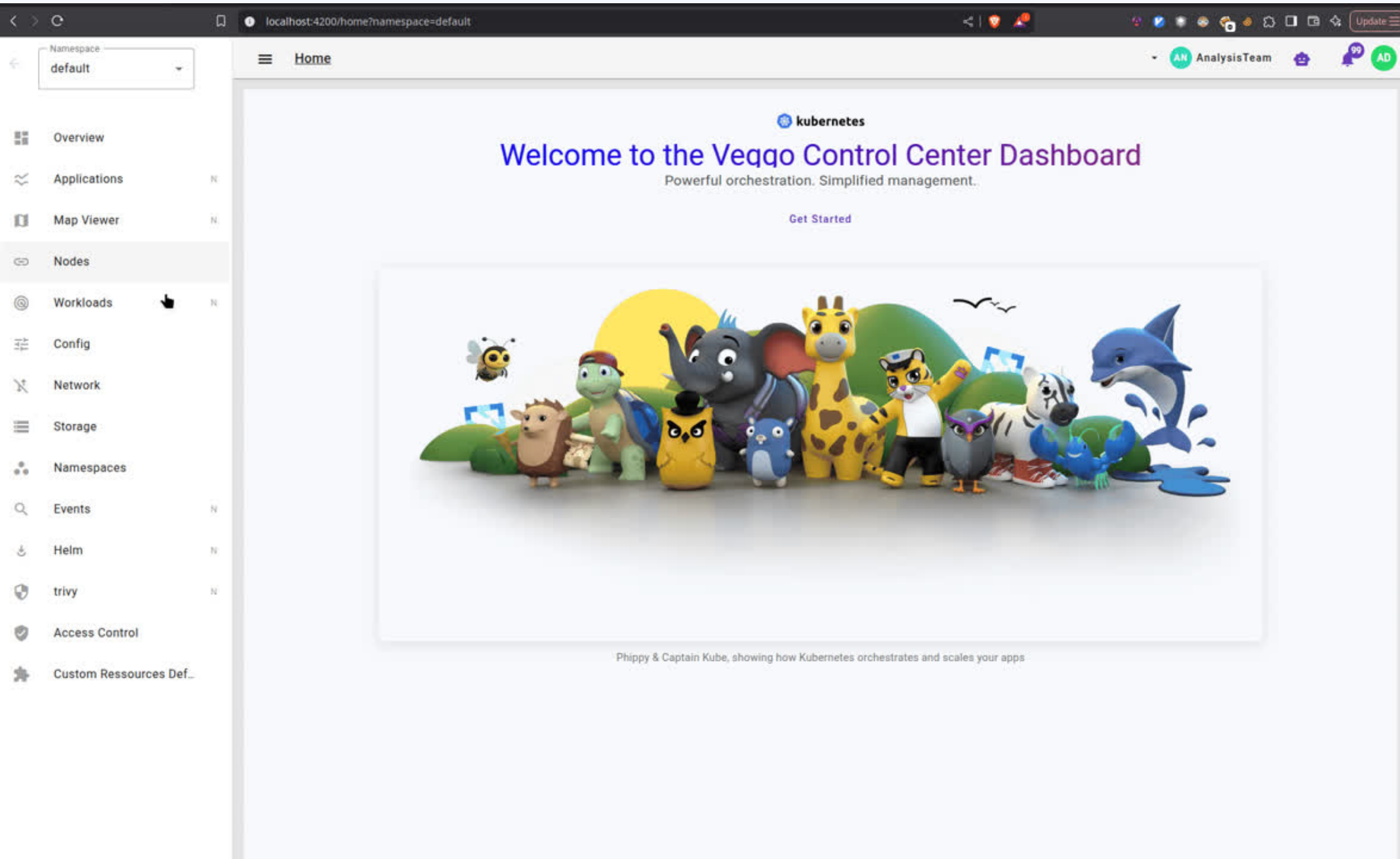
Réalisation

Objectif Projet

Specification des Besoins

PAGE 29 OF 38

Interface cartographique de visualisation des ressources Kubernetes



Context et

Analyse et

Conception

Réalisation

Objectif Projet

Specification des Besoins

PAGE 30 OF 38

Workloads Interface

The screenshot displays the Kubernetes Workloads interface in a web browser. The URL is `localhost:4200/workloads/replicaset?namespace=default`. The interface includes a sidebar with navigation options: Overview, Applications, Map Viewer, Nodes, Workloads, Config, Network, Storage, Namespaces, Events, Helm, trivy, Access Control, and Custom Ressources Def... The main content area shows a list of Deployments under the 'replicaset' tab. The Deployments table has columns for Name, Images, Labels, Pods, Created, and Actions. The following table represents the data shown in the screenshot:

Name	Images	Labels	Pods	Created	Actions
my-apache	<code>docker.io/bitnami/apache:2.4.63-debian</code>	<code>app.kubernetes.io/instance: my-apache</code> <code>app.kubernetes.io/managed-by: Helm</code> <code>app.kubernetes.io/name: apache</code> Show all	1 / 1	Jun 9, 2025	⋮
prometheus-kube-state-metrics	<code>registry.k8s.io/kube-state-metrics/kube-</code>	<code>app.kubernetes.io/component: metrics</code> <code>app.kubernetes.io/instance: prometheu</code> <code>app.kubernetes.io/managed-by: Helm</code> Show all	1 / 1	Jun 14, 2025	⋮
prometheus-prometheus-pushgateway	<code>quay.io/prometheus/pushgateway:v1.1)</code>	<code>app.kubernetes.io/instance: prometheu</code> <code>app.kubernetes.io/managed-by: Helm</code> <code>app.kubernetes.io/name: prometheus-p</code> Show all	1 / 1	Jun 14, 2025	⋮
prometheus-server	<code>quay.io/prometheus-operator/promethe</code> <code>quay.io/prometheus/prometheus:v3.4.1</code>	<code>app.kubernetes.io/component: server</code> <code>app.kubernetes.io/instance: prometheu</code> <code>app.kubernetes.io/managed-by: Helm</code> Show all	1 / 1	Jun 14, 2025	⋮

Below the Deployments table, there is a section for Replica Sets, which is currently empty, displaying the message: "There is nothing to display here. No resources found."

The bottom of the screenshot shows a system tray with various icons, including a VPN icon, IP address `192.168.1.11`, battery level `56.3%`, CPU usage `54.8%`, memory usage `5.5%`, and the date/time `Mon 23/06 11:54`.

Context et

Analyse et

Conception

Réalisation

Objectif Projet

Specification des Besoins

Nodes Interface

localhost:4200/node?namespace=default

Namespace: default

Nodes

Items: 1

Overview Applications Map Viewer Nodes Workloads Config Network Storage Namespaces Events Helm trivy Access Control Custom Ressources Def...

Memory limits (bytes) Memory capacity (bytes) Pods Created ↑ Action

476.00Mi (6.35%) 7.32Gi X Jun 9, 2025

Edit a resource

YAML JSON

```
5 kubeadm.alpha.kubernetes.io/cni-socket: unix:///run/containerd/containerd.sock
6 node.alpha.kubernetes.io/ttl: '0'
7 volumes.kubernetes.io/controller-managed-attach-detach: 'true'
8 creationTimestamp: '2025-06-09T09:05:59.000Z'
9 labels:
10   beta.kubernetes.io/arch: amd64
11   beta.kubernetes.io/os: linux
12   kubernetes.io/arch: amd64
13   kubernetes.io/hostname: minikube
14   kubernetes.io/os: linux
15   minikube.k8s.io/commit: f8f52f5de11fc6ad8244afac475e1d0f96841df1-dirty
16   minikube.k8s.io/name: minikube
17   minikube.k8s.io/primary: 'true'
18   minikube.k8s.io/updated-at: 2025_06_09T09_05_59_0700
19   minikube.k8s.io/version: v1.36.0
20   node-role.kubernetes.io/control-plane: ''
21   node.kubernetes.io/exclude-from-external-load-balancers: ''
22 managedFields:
23   - apiVersion: v1
24     fieldType: FieldsV1
25     fieldsV1:
26       f:metadata:
27         f:annotations:
28           .: {}
29         f:volumes.kubernetes.io/controller-managed-attach-detach: {}
30         f:labels:
31           .: {}
32         f:beta.kubernetes.io/arch: {}
33         f:beta.kubernetes.io/os: {}
34         f:kubernetes.io/arch: {}
35         f:kubernetes.io/hostname: {}
36         f:kubernetes.io/os: {}
37 manager: kubelet
38 operation: Update
39 time: 2025-06-09T09:05:59.000Z
```

Update Cancel

VPN 192.168.1.11 56.3% 61.8% 7.5% Mon 23/06 12:00 49% 0%

Context et

Analyse et

Conception

Réalisation

Objectif Projet

Spécification des Besoins

Config Interface

localhost:4200/config/secret?namespace=default

Unknown > SecretsConfig MapsConfiglist

AN AnalysisTeam

list configmap secret

Secrets Items: 3

Name	Labels	Type	Created	Actions
sh.helm.release.v1.my-apache.v1	modifiedAt : 1749484691 owner : helm Show all name : my-apache	helm.sh/release.v1	Jun 9, 2025	Edit Delete
sh.helm.release.v1.prometheus.v1	modifiedAt : 1749923104 owner : helm Show all name : prometheus	helm.sh/release.v1	Jun 14, 2025	
sh.helm.release.v1.prometheus.v2	modifiedAt : 1749923104 owner : helm Show all name : prometheus	helm.sh/release.v1	Jun 14, 2025	

Context et

Analyse et

Conception

Réalisation

Objectif Projet

Specification des Besoins

PAGE 33 OF 38

Networking Interface

The screenshot displays a web-based interface for managing Kubernetes networking. The left sidebar contains a navigation menu with the following items: Overview, Applications, Map Viewer, Nodes, Workloads, Config, Network, Storage, Namespaces, Events, Helm, trivy, Access Control, and Custom Ressources Def... The main content area is titled 'EndpointsNetworkList' and shows a table of endpoints. The table has columns for Host, Namespace, Ip, Endpoint Protocol, and Created. There are 8 items listed. The interface also includes a top navigation bar with a search bar, a user profile (AN AnalysisTeam), and a language selector (AD). The URL in the browser is localhost:4200/network/endpoint?namespace=default.

Host	Namespace	Ip	Endpoint Protocol	Created
kubernetes	default	192.168.49.2	TCP of type https on 8443	2025-06-09T09:05:57.000Z
my-apache	default	10.244.0.19	TCP of type https on 8443 TCP of type http on 8080	2025-06-09T15:58:11.000Z
prometheus-alertmanager	default	10.244.0.8	TCP of type http on 9093	2025-06-14T17:45:04.000Z
prometheus-alertmanager-headless	default	10.244.0.8	TCP of type http on 9093	2025-06-14T17:45:03.000Z
prometheus-kube-state-metrics	default	10.244.0.7	TCP of type http on 8080	2025-06-14T17:45:04.000Z
prometheus-prometheus-node-exporter	default	192.168.49.2	TCP of type metrics on 9100	2025-06-14T17:45:04.000Z
prometheus-prometheus-pushgateway	default	10.244.0.6	TCP of type http on 9091	2025-06-14T17:45:04.000Z
prometheus-server	default	10.244.0.5	TCP of type http on 9090	2025-06-14T17:45:04.000Z

Context et

Analyse et

Conception

Réalisation

Objectif Projet

Specification des Besoins

Access Control Interface

The screenshot displays the 'Access Control Interface' for managing Kubernetes resources. The main view shows a list of Namespaces, with 'default' selected. A modal dialog titled 'Edit a resource' is open, showing the YAML configuration for the 'default' Namespace. The configuration includes metadata, managedFields, and spec details. The dialog also provides a note that the action is equivalent to running 'kubectl apply -f <spec.yaml>' and buttons for 'Update' and 'Cancel'.

```
9.  kubernetes.io/metadata.name: default
10. managedFields:
11.   - apiVersion: v1
12.     fieldType: FieldsV1
13.     fieldsV1:
14.       f:metadata:
15.         f:labels:
16.           .: {}
17.         f:kubernetes.io/metadata.name: {}
18.       manager: kube-apiserver
19.       operation: Update
20.       time: '2025-06-09T09:05:55.000Z'
21.   - apiVersion: v1
22.     fieldType: FieldsV1
23.     fieldsV1:
24.       f:metadata:
25.         f:annotations:
26.           .: {}
27.         f:kubectl.kubernetes.io/last-applied-configuration: {}
28.       manager: kubectl-client-side-apply
29.       operation: Update
30.       time: '2025-06-14T17:49:33.000Z'
31. name: default
32. resourceVersion: '46168'
33. uid: b0d12cdf-3f6d-4cd2-97df-e68e4a887e66
34. spec:
35.   finalizers:
36.     - kubernetes
37. status:
38.   phase: Active
39.
```

This action is equivalent to: `kubectl apply -f <spec.yaml>`

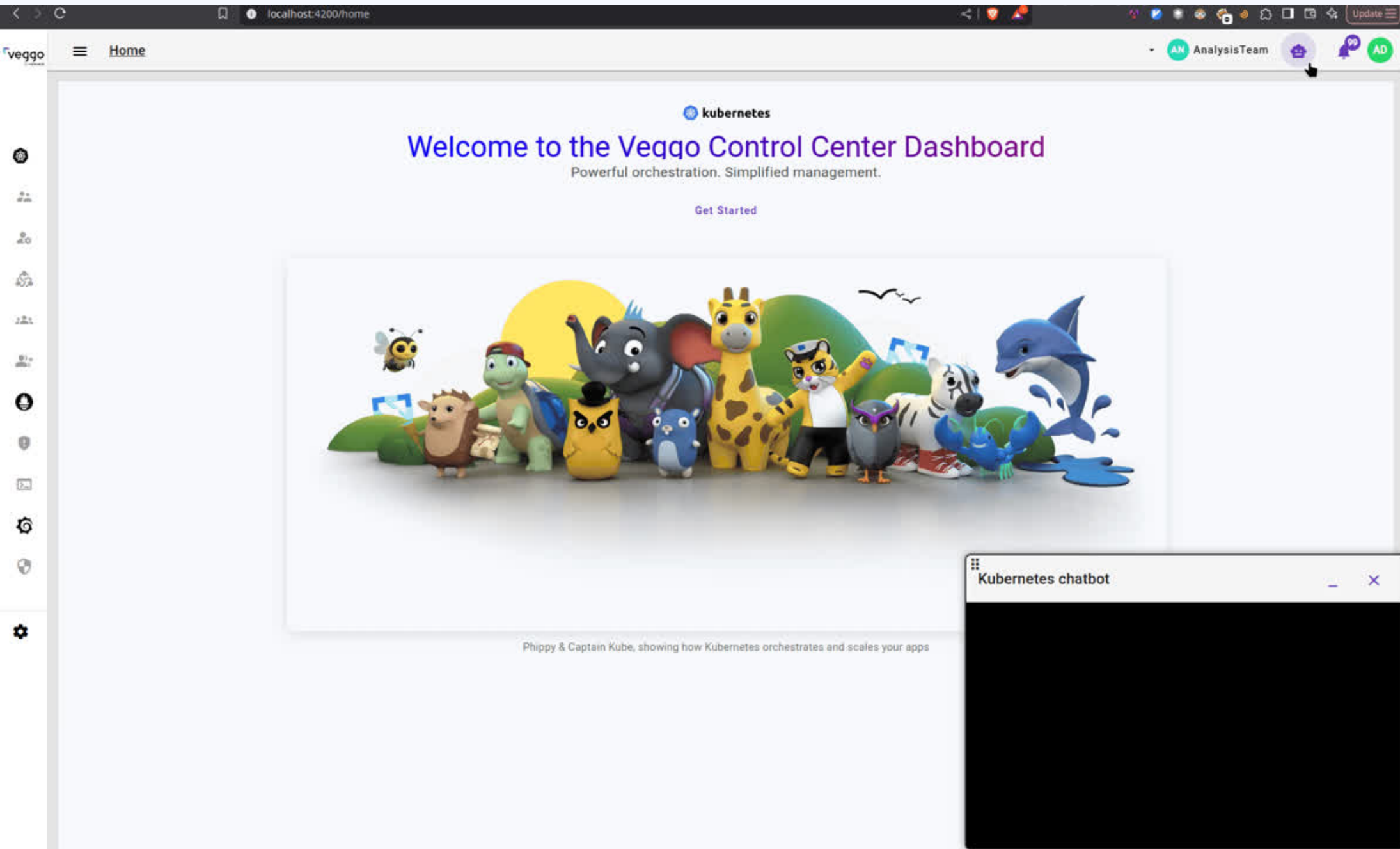
[Update](#) [Cancel](#)

Conclusion et perspectives

Dans une perspective d'évolution, l'application nécessitera le développement de fonctionnalités supplémentaires afin d'être pleinement compatible avec les services Kubernetes managés des principaux fournisseurs cloud, tels que Google Kubernetes Engine (GKE) sur Google Cloud Platform, Elastic Kubernetes Service (EKS) sur AWS, ou Azure Kubernetes Service (AKS) sur Microsoft Azure.

Par ailleurs, des notions plus avancées de Kubernetes devront être intégrées, telles que la gestion multi-cluster.

Chatbot Kubectl-ai



Context et

Analyse et

Conception

Réalisation

Objectif Projet

Specification des Besoins

PAGE 37 OF 38

MERCI POUR VOTRE ATTENTION

Avez-vous des questions ?

