MINISTÈRE
DE L'INTÉRIEUR
Liberté
Égalité
Fraternité

Cadre de Cohérence Technique (CCT)

Au profit de l'intégration de l'Intelligence Artificielle dans les applications

"Socle IA MIRAI[1]"

VERSION DE TRAVAIL

Date : 12/02/2026

Auteur : DTNUM, Ministère de l'Intérieur

[1] Mirai (MI - r - AI) signifie ***futur*** en japonais

VERSION DE TRAVAIL

A propos de l'appel à commentaire de ce document.

Ce même document est disponible sur le repository GitHub de la direction du numérique du MI :

https://github.com/IA-Generative/CCT-Mirai

Vous pouvez remonter vos commentaires et suggestions sur ce document de plusieurs manières:

- 1/ utiliser le fichier de relecture proposé Fichier pour commentaires (réutilisation d'un document existant):
  - https://github.com/dnum-mi/CCT-Cloud-Native/blob/main/gabarit-pour-commentaires.ods
  - et l'envoyer à : dnum-architecture-entreprise@interieur.gouv.fr
- 2/ enregistrer des issues dans github

VERSION DE TRAVAIL

TABLE DES MATIÈRES

--- page vide ---

1 - Introduction

Ce présent volet, du cadre de cohérence technique porte sur les conditions d'utilisation et la contribution au socle ministériel d'intelligence artificielle du ministère pour permettre une utilisation optimum des ressources et la rapidité de déploiement des solutions dans une architecture permettant la maîtrise des enjeux cyber et d'eco-responsabilité.

Ce document s'adresse aux développeurs, architectes et en général aux acteurs se projetant dans la planification, l'élaboration et la maintenance de produits numériques devant s'intégrer dans l'écosystème MirAI[2].

La construction de ce socle est menée de manière itérative et incrémentale. Le backlog est révisé tout les trimestre.

Le présent document, maintenu à jour régulièrement, présente l'état actuel et les ajouts de services envisagés. Cela permet dans un esprit de collaboration ministere et interministériel de permettre la mutualisation des moyens et la partage de la connaissance. Le lecteur intéressé est invité à prendre contact avec l' équipe.

Le document présente le cadre et les exigences pour permettre à une direction d'application de faciliter la construction d'application de qualité et l'accès à l'offre de service proposée.

Ce document et les ressources associées ont pour objectifs de :

- guider les concepteurs d'applications afin d'optimiser les architectures produites selon des normes industrielles rigoureuses, tout en maintenant une capacité d'innovation ;
- mettre à disposition un référentiel d'exigences favorisant les bonnes pratiques et la conformité ;
- optimiser la consommation de ressources (financière, RH, énergétique) par la réduction de la quantité de code à produire et la modularité, l'efficience des architecture applicatives, la rationalisation des composants utilisés et l'optimisation des ressources de calcul disponibles ;
- de s'assurer de la compatibilité avec le socle mis à disposition ;
- de prendre en compte les spécificités de l'IA dans les déploiements des solutions, l'homologation en continu, le maintien en qualité ;
- de favoriser la conformité *by design* sur la sécurité, la protection et circulation de la donnée, notamment la donnée qui fait référence pour les usages métiers ;
- de soutenir le socle de sécurité facilitant l'homologation des systèmes;
- mettre en place un modèle de responsabilité et de collaboration adapté ;
- de disposer d'une trajectoire soutenable pour ceux en charge de maintenir les applications et les éléments du socle Mirai ;
- Favoriser l'adoption des pratiques liées au mode produit et à l'IA.

Le lecteur est invité à vérifier qu'il dispose de la dernière version de ce document de présent ainsi que de la liste d'exigences. contenu susceptible de changer régulièrement.

[2] Mirai (MI - r - AI) signifie ***futur*** en japonais

2 - Le contexte, les enjeux, la vision

Audience : ce paragraphe s'adresse à tout acteur considérant l'usage du socle Mirai du ministère de l'intérieur pour développer des solutions basées sur l'intelligence artificielle.

Ce document "Cadre d'intégration et de Cohérence Technique IA" vise à établir un cadre structuré pour l'intégration et la gestion des applications basées sur l'intelligence artificielle au sein du ministère de l'Intérieur et des Outre-Mer.
Il définit les principes directeurs, les exigences techniques, et les modalités d'intégration du socle IA Mirai, garantissant ainsi la cohérence, la sécurité et la performance des solutions développées.

Objectifs principaux :

- Alignement stratégique : Assurer que les initiatives IA s'inscrivent dans une vision globale et cohérente.
- Gouvernance et responsabilité : Clarifier le rôle des différents acteurs impliqués
- Établir un modèle de pilotage efficace.
- Exigences techniques : Standardiser l'architecture des applications, garantir l'interopérabilité et optimiser l'utilisation des infrastructures Cloud et On-Prem.
- Sécurité et conformité : Assurer la protection des données sensibles et la conformité avec les réglementations en vigueur (RGPD, IA Act).
- Optimisation des ressources : Encourager l'éco-conception et la rationalisation des coûts, notamment en mutualisant les infrastructures et en réduisant la dette technique.
- Automatisation et efficacité opérationnelle : Intégrer des pratiques DevSecOps et MLOps pour une gestion fluide du cycle de vie des modèles IA et des applications.
- Suivi et amélioration continue : Mettre en place des dispositifs d'audit, de monitoring et de gestion des dérives algorithmiques.

Le document sert donc de référentiel technique et organisationnel pour faciliter la conception, le déploiement et l'exploitation d'applications IA dans un cadre sécurisé, évolutif et aligné sur les exigences ministérielles.

3 - Principes généraux du socle MirAi

Audience : ce paragraphe s'adresse à la communauté des concepteurs et architectes solutions.

Ce document normalise les différents domaines associés à l'élaboration et au maintien des ressources partagées nécessaires à la mise à disposition de solutions numériques de qualité répondant au besoin.

Il favorise que l'ensemble peut-être mis en œuvre de manière cohérente avec une consommation minimisée des ressources : financière, RH et éco responsable tout en étant conforme *by design* aux référentiels de normes de l'Etat.

Il recommande ou fixe les mesures permettant d'atteindre l'objectif, tout en favorisant l'innovation, la prise en compte de l'obsolescence régulière des technologies et la manœuvre RH nécessaire (formation continue, recrutement …)

Les applications basées sur les réseaux de neurones, dont l'IA générative peuvent nécessiter l'accès à une ressource scalable de calcul haute performance (HPC) incluant des accélérateurs de traitement simultané de données en masse (GPU).[3]

Ces ressources de calcul consomment plus de ressource énergétique qu'une infrastructure classique pour permettre l'accélération et le traitement de modèle d'IA large tel que les modèles de langage large (LLM[4]) pouvant atteindre des poids dépassant la centaine de gigaoctet.

Il est essentiel de bien concevoir les solutions afin d'optimiser l'accès à ces ressources coûteuses, c'est l'objectif du socle MirAi, permettre de mutualiser ce qui peut l'être, sans freiner l'innovation et accélérer la réalisation des applications en leur proposant des services communs d'accélération. La trajectoire consiste à identifier les besoins de mutualisation et de mettre en œuvre une trajectoire d'évaluation de l'opportunité au regard des ressources accessibles.

La feuille de route MIrAi inclut la construction de service directement accessible par l'agent par une IHM et d'APIs à destination des constructeurs d'applications.

Un second aspect concerne la gestion de l'accès et la circulation de la donnée, notamment l'accès à des données faisant références, tel que les textes législatifs de références, les corpus de connaissances applicables, les rapport et instructions disponibles au sein d'un métier ou plus simplement à des données partagées.

Un troisième aspect concerne la mise en place d'une architecture et un cadre permettant l'orchestration de service et permettant la mise en place d'agents intelligents[5].

---

[3] https://fr.wikipedia.org/wiki/Single_instruction_multiple_data
[4] https://fr.wikipedia.org/wiki/Grand_mod%C3%A8le_de_langage
[5] https://cloud.google.com/discover/what-are-ai-agents

Un quatrième aspect concerne la circulation et la valorisation de la donnée opérationnelle et l'accès aux services métiers qui pourraient être intégrés dans les agents conversationnel et à compétences IA pour aider l'agent ou l'usager. (agentique IA)

Périmètres du document et configurations prises en compte

Ce document concerne le socle Mirai dans les configurations précisées ci-dessous.

- Utilisation sur les plateformes mis à disposition et managées par le ministère;
- Utilisation de ce socle dans le cas où le commun numérique est déployé sur une infrastructure non maîtrisée par le ministère.

Dans le cas où le socle est déployé en dehors du ministère, le bon usage et le respect du corpus d'exigence applicable est assuré par l'utilisateur.

Ce cadre de cohérence et les solutions portées par le socle Mirai s'appuient sur le socle Cloud Pi native.

Afin d'assurer la conformité aux exigences étatiques et faciliter les homologations des systèmes, les applications construites doivent donc respecter au delà des bonnes pratiques de construction d'applications :

- **le corpus d'exigences associées au socle Mirai (ce document)**
- **le corpus d'exigence de Pi Native[6] dont l'usage de la chaîne DevSecOps.**

Gestion des non-conformités, dérogations et contribution

L'évolution rapide des technologies cloud et IA peuvent conduire à ce que les informations ou exigences contenues dans ce document restreignent l'innovation.

Il est également souhaité, pour éprouver le modèle, de notifier le département architecture d'entreprise du Ministère de l'intérieur au plus tôt des éventuelles impossibilités ou limitations remarquées afin de rechercher des alternatives de conception ou faire évoluer ce cadre.

Les directions d'applications ou les organisations utilisatrices peuvent contribuer, via un échange préalable, à enrichir les fonctionnalités de l'offre ou du cadre lui-même. Sur l'offre la contribution est effectuée directement sur le repository open source de la solution via un pull request.

En cas de non-conformité aux exigences de ce document ou absence de contribution à l'offre, une demande de dérogation dûment motivée sera formulée à l'avance par la direction d'application. Seule la notification de la décision permet d'amender le besoin de conformité au cadre, temporairement ou de manière pérenne dans le cadre d'une dérogation. Dans le cadre d'une dérogation, la direction d'application prend à sa charge le surcoût complet de possession. (formation, homologation, personnel assurant la tme, etc…)

Toute organisation souhaitant décliner ce cadre dans un document de norme inférieur pour un besoin propre est invitée à référencer la dernière version de ce document (tel que mis à

[6] Le cadre Cloud Pi NAtive est disponible : https://github.com/cloud-pi-native/cct-cloud-native

disposition sur le répertoire github) et d'éviter de dupliquer le contenu. Dans la hiérarchie des normes, une instruction de niveau inférieur ne peut entrer en conflit ou contredire ce présent document.

Le modèle organisationnel, de responsabilité et de collaboration autour de Mirai

L'architecture, le modèle de responsabilité et d'organisation à mettre en place est orienté pour maximiser la qualité, la sécurité, la fluidité opérationnelle et l'évolutivité du produit en tirant parti au maximum des possibilités offertes par la technologie IA, kubernetes, flux de production DevSecOps et une collaboration étendue entre les acteurs. (Ce chapitre reprend le contenu du cadre de cohérence Pi Native pour faciliter la lecture de ce document)

L'élargissement de la responsabilité du développeur et de l'équipe produit

La responsabilité de l'équipe produit est élargie dans le cadre Cloud Native. Elle élabore et exploite une solution qui répond au besoin métier généralement une automatisation d'un ou plusieurs processus métiers . L'équipe s'assure de la qualité et de la disponibilité du service rendu à l'usager selon le précepte : « You build it, you run it ». L'équipe s'organise de façon intégrée, si nécessaire avec de l'externalisation, pour couvrir l'ensemble des aspects nécessaires de la conception à l'exploitation des produits.

Le développeur, en particulier, met à disposition d'un point de vérité du code sous la forme d'un ou plusieurs dépôts de code logiciel fonctionnel et d'infrastructure. Il met en place un flux intégré et continu de production en s'appuyant sur un orchestrateur primaire DevSecOps qu'il construit et opère.

Le développeur initialise et supervise ses *pipelines* primaire et secondaire. Il intègre les étapes de vérification de sécurité génériques imposées par le ministère et spécifiques issus de la démarche d'homologation.

L'ensemble combiné des orchestrateurs primaire et secondaire soutient la fonction d'homologation et de déploiement en continu du produit numérique.

Dans le cas de la détection d'une non-qualité critique, telle une vulnérabilité critique, la progression du déploiement est bloquée, le développeur est prévenu en temps réel et doit corriger au plus tôt les défauts remontés. Cette approche permet de garantir un niveau de qualité, évite des régressions et maintient la dette technique au niveau le plus bas.

Sur le plan organisationnel le développeur met généralement en place :

- un contrôle de qualité au plus tôt, par exemple par un assistant et la revue de code ;
- l'agilité avec des itérations courtes de constructions et de vérification des usagers ;
- le découpage des livraisons en lot de taille de réduite ;
- la mise en place d'une culture de collaboration étendue et des pratiques intégrant la sécurité à toutes les étapes.

La répartition des rôles et responsabilités s'établit de la manière suivante :

L'équipe produit intégrée :

- est responsable de l'application, de la qualité du code et du bon fonctionnement de l'application pendant l'ensemble du cycle de vie de l'application.
- est responsable de définir et d'ajuster l'infrastructure en s'appuyant sur l'élasticité du cloud.(sur la base de l'offre Cloud adaptée selon la sensibilité des données), l'utilisation des APIs disponiblent (ou planifiée) pour désigner la solution.
- S'assurer de la préparation et du maintien de la qualité des données manipulées ;
- met en place des pratiques DevSecOps visant un maintien de la qualité dans le temps avec les composantes suivantes (cf outillage DevSecOps) :
    - test driven development ;
    - couverture de test unitaire à 100% du back-end de l'application ;
    - couverture significative des tests du front de l'application ;
    - analyse statique de qualité du code ;
    - analyse récursive des vulnérabilités des librairies importées ;
    - utilisation exclusivement d'images sources maintenues en condition de sécurité et certifiées (distribution LTS) ;
    - conception des tests d'intégration en sandbox ;
    - fourniture des outils nécessaires à la remontée de l'état de santé des briques applicatives destinées à fonctionner en production (healthcheck) ;
    - fourniture des indicateurs nécessaires au suivi en temps réel de la qualité en condition opérationnelle de sa solution (exports prometheus) ;
    - exploitation des logs remontés.
- met en place un hébergement sur une plateforme kubernetes afin d'assurer la démonstration du bon fonctionnement de l'application avec la solution qu'il préfère soit internalisée (avec un moyen de mener des démonstrations) ou sur cloud public.
- maintient un point de vérité du code logiciel ainsi que celui du code d'infrastructure. Celui-ci est accédé par la chaîne DevSecOps étatique, la sécurisation d'accès issus par token.
- est responsable de la surveillance de l'ensemble des pipelines, y compris pour celui géré côté ministère.
- met en place une intégration du flux de retour d'anomalie "shift-left" des orchestrateurs afin de permettre une correction au plus tôt des anomalies.
- effectue l'apprentissage comportemental du firewall applicatif Web (WAF) vis-à-vis de l'application dans le cadre fixé par le ministère. (anticipation avant la mise en production)
- est invité à mettre en œuvre ce pipeline au plus tôt dans le processus de réalisation ;
- met en place une gouvernance et un suivi lié à l'IA responsable et aux biais algorithmiques ;
- Met en place une organisation chargée de veiller à la qualité et la conformité réglementaire des données et de leur utilisation.

L'équipe de développement respecte les règles suivantes permettant une qualité de code en progression et un maintien de la sécurité :

- minimise la portion spécifique de code développés en s'appuyant sur le catalogue des services proposés. (revoir régulièrement)
- met en place une couverture de test unitaire complète du back-end (et fourni les moyens de vérification automatisé à la chaîne secondaire)

- mener une analyse de code systématique le plus tôt possible (les langage et IDE modernes fournissent des fonctions de ce type)
- mener une analyse de CVE des dépendants importées et apporter des corrections.

L'équipe intégrée met en œuvre une activité continue de refactoring du code produit. Ainsi, la qualité du code ne peut pas être décroissante.

Elle fournit les preuves que des tests de sécurité, de qualité, de robustesse des algorithmes ont été mis en œuvre, et qu'ils n'ont pas remontés de vulnérabilités ou d'erreurs majeures. En s'appuyant notamment sur les logs des analyses des outils de la chaîne primaire. Elle fournit la preuve (ex: le document) des normes de développement et pratiques permettant de maîtriser la qualité du code produit. (refactoring, peer review, etc..)

Note : l'équipe s'assure qu'après le dernier déploiement stable de l'application, toutes nouvelles vulnérabilités critiques et importantes seront détectées et corrigées. En cas de non correction des anomalies dans un délai de plusieurs mois et surtout si l'application est exposée sur internet, l'hébergement de la solution pourra être suspendu pour maintenir en intégrité les données.

Les opérateurs des socle Mirai et Pi native :

Des pratiques complémentaires sont introduites :

Le "GitOps", contraction de git et opération, est indispensable à la gestion des applications Cloud Native avec Kubernetes. Ce mode d'organisation du code d'infrastructure permet de maîtriser la description de l'infrastructure de production avec les mêmes pratiques de revue collaborative que celle du logiciel. Il est par exemple strictement interdit de faire des modifications «à la main » sur l'environnement de production, toute variation est supprimée, l'infrastructure réelle est strictement celle décrite par les fichiers d'infrastructure.

Le **"shift-left"** (vers la gauche du processus de développement) fait référence à la remontée le plus tôt possible vers le développeur des anomalies identifiées par la chaîne de déploiement et de vérification DevSecOps. Ce flux est notamment mis en œuvre depuis la chaîne secondaire.

Les pratique "MLOPS" liées à l'utilisation des réseaux de neurone tel que celle décrites ici :

https://neptune.ai/blog/mlops-best-practices

Préconisations générales d'architecture

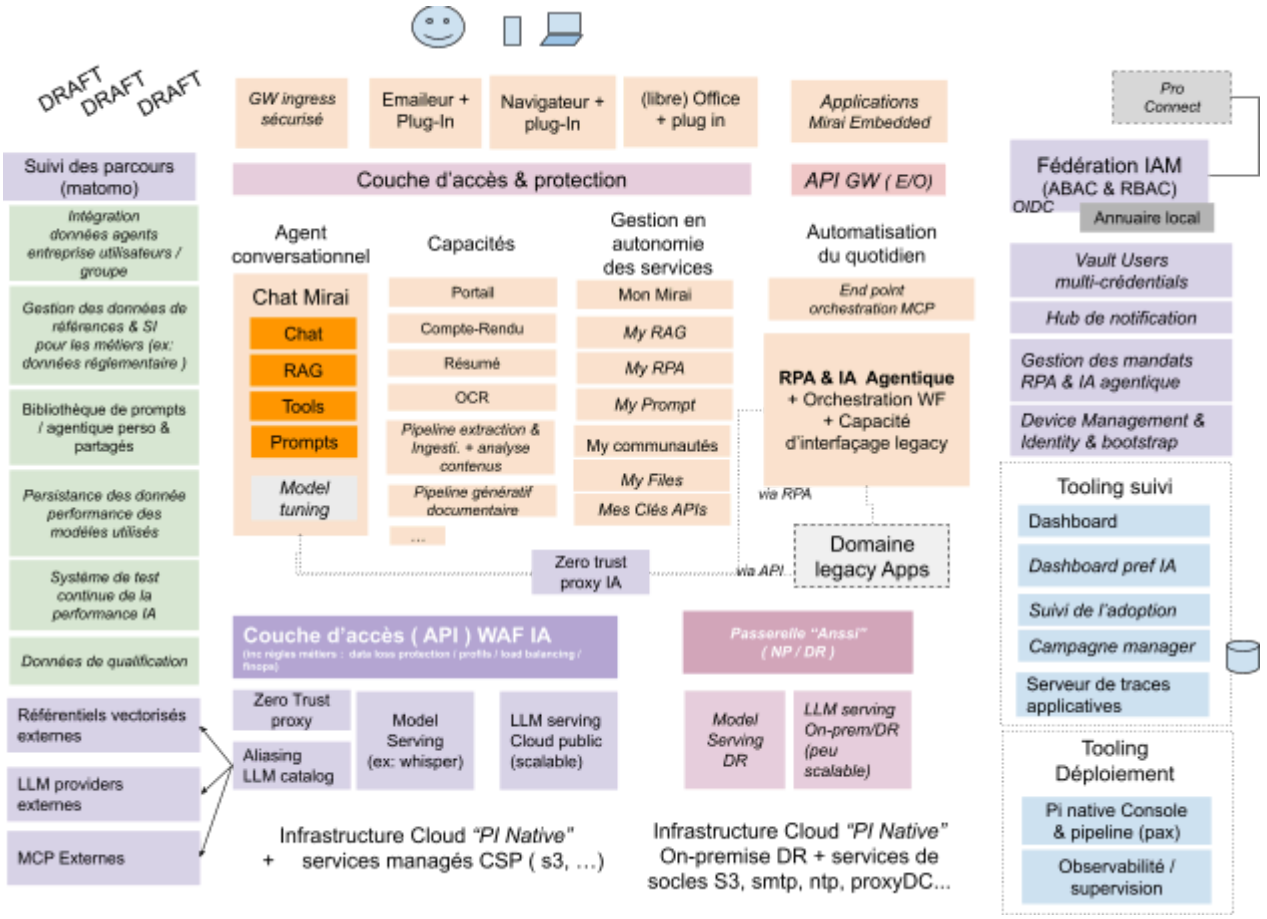
Ce chapitre précise les aspects importants liés à l'usage du socle Mirai dans le cadre du ministère de l'intérieur. Il est attendu que les acteurs soient correctement formés à la solution kubernetes, à l'utilisation d'API , de l'IA et mènent une veille régulière, la technologie évoluant rapidement.

C'est le respect de ces normes qui permet à la fois d'adresser les enjeux de performance en termes de vitesse de livraison et de qualité de service, mais aussi de normaliser les applicatifs pour une meilleure évolutivité et maîtrise de la dette technique. Enfin, elles assurent une intégration fluide au sein des systèmes d'information Ministériels.

Un des principes cœurs est de laisser un certain degré de liberté au concepteur/développeur sur le fonctionnement interne de son application mais de cadrer fortement les interfaces externes.

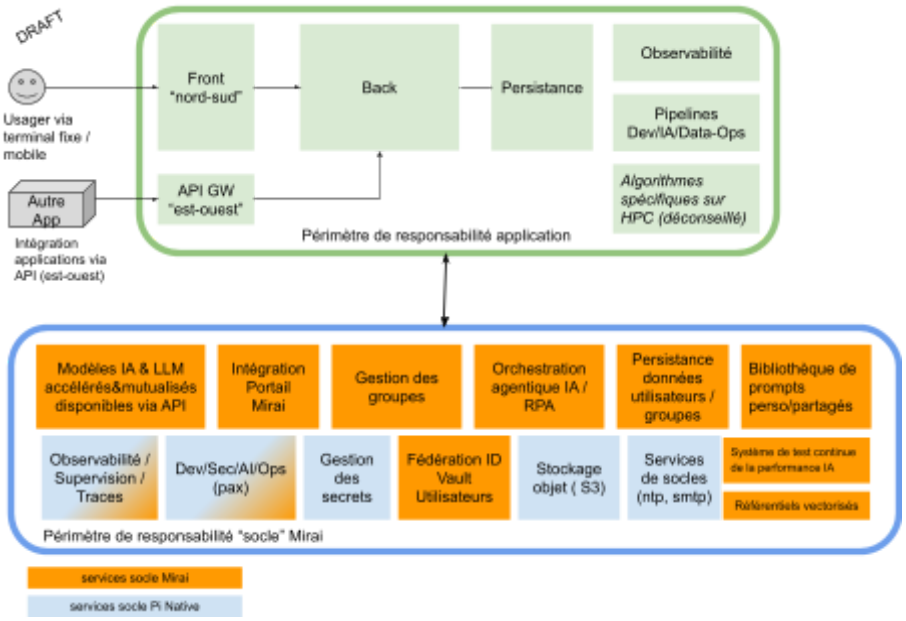
Les schémas en pages suivantes décrivent le modèle de référence (construit incrémentalement) et le cadre d'intégration d'application.

Modèle d’architecture logique du socle Mirai (en cours d’élaboration)



Modèle d’intégration pour les applications

(travail en cours)



Description des composants du socle Mirai

(travail en cours)

Réf	Désignation	Fonction	Etat de maturitē
ACCUSER	Terminaux d'accès	Permettre l'accès au service. Terminaux navigateurs : firefox, Safari et famille Chromiium Chrome/Edge Plug-ins : xOffice, Thunderbird (client de messagerie melanie), navigateur Mobile	https://github.com/IA-Generative/mirai-assistant https://github.com/IA-Generative/AssistantMiraiLibreOffice https://www.chromium.org/chromium-projects/
ACCAPP	Accès pour les applications	Gateway API, permet de fournir un accès à l'ensemble des services. doit être en lien avec la gestion autonome des clés APIs	
APPINT	Applications internes	Applications internes conformes au socle Mirai, déployées sur les infrastructures ministérielles, s'appuyant sur les services mutualisés du socle. Accessibles sur terminaux fixes ou mobiles, elles peuvent exposer des API, des applications intégrées ou des interfaces de présentation consommant des services distants.	Solution déployée à partir de la stack cloud pi native. https://github.com/cloud-pi-native Codes des applications internes disponible sur : https://github.com/IA-Generative Agent conversationnel : https://github.com/open-webui/open-webui
APPEXT	Applications externes	"Ensemble des applications fournies par un tier de confiance, elles s'executent en dehors des infrastructures maitrisées par le ministère. Permet de référencer ou de présenter ces ressources, d'y inclure des informations additionnelles, bonne pratique d'usage, etc... (ex: Grist)"	
COMP001	Couche d'accès & protection	Contrôle des accès et filtrage des requêtes malveillantes (ex. pare-feu, WAF).	En place

COMP002	Portail de présentation et d'intégration	Portail assurant la présentation et la personnalisation des services pour les usagers mobiles et postes fixes. Il permet l'intégration d'applications internes ou externes, la diffusion d'informations et de contenus métiers, ainsi que la valorisation de services en expérimentation. La personnalisation repose sur le profil et les attributs de l'usager. La gestion des contenus s'appuie sur un CMS (« Sites Faciles ») sans compétence technique requise.	En place pour Mirai (solution site facile dinum) https://github.com/numerique-gouv/sites-faciles
COMP003	Fédération ID & calcul des profils (ABAC)	Met à disposition des applications les données d'organisation nécessaires à l'autorisation des usagers et au calcul automatique des droits. Les comptes sont issus des référentiels d'entreprise et des fournisseurs d'identité disponibles, dans le cadre d'un dispositif de fédération d'identité (ProConnect).	Solution Keycloak et fédération d'identité PRo Connect. https://github.com/keycloak/keycloak https://partenaires.proconnect.gouv.fr/
COMP004	Annuaire local	Annuaire assurant la gestion des comptes externes autorisés ainsi que des groupes et appartenances associées, en complément des référentiels d'entreprise. (intégré à keycloak)	intégré à keycloak.
COMP005	Gestion des groupes / profils	Composant permettant d'exposer, via des API, les informations étendues relatives aux utilisateurs et aux organisations, et d'assurer la gestion des groupes, notamment pour des communautés de pratique transverses (keycloak-comu).	Solution Custom déployée. https://github.com/IA-Generative/keycloak-comu
COMP006	Persistance données utilisateurs / groupes (prompt, données d'usage)	Composant fourni en mode service et exposé via API, chargé de la persistance et de la circulation des données des usagers et des groupes, dans une logique d'accès en modalité zero trust. Il porte les règles d'accès et assure la traçabilité des usages. Il permet notamment de persister des corpus de données pour un usager ou un groupe, des prompts partagés ou personnels, ainsi que des données de paramétrage.	
COMP007	Agents IA & orchestration	Ce composant porte la mise à disposition et le contrôle des accès et la traçabilité des utilisations d'agents	En expérimentation. https://github.com/n8n-io/n8n

		IA s'intégrant avec le legacy d'entreprise	https://github.com/microsoft/playwright
COMP008	Données de références & SI (catalogue)	Service fournissant un catalogue des corpus de données accessibles et facilite la découverte et réalisation des cas d'utilisation. Présente également les conditions d'utilisation de la donnée et les contacts éventuels. (ne fournit pas la donnée directement)	
COMP009	Bibliothèque de prompts perso/partagés	Référentiel libre d'accès de prompts éprouvé avec contribution et classement par les utilisateurs.	
COMP010	Couche d'accès AI (API) (et règles métiers : data loss protection / profils / load balancing / finops)	Ce composant permet de modulariser et découpler l'architecte des applications et de porter plusieurs règles métiers telles que le load balancing (résilience) entre plusieurs fournisseurs, l'optimisation du coût des requêtes, l'orientation selon la sensibilité des données, un niveau de protection contre la fuite des données et la traçabilité des requêtes.	An développement continue à partir de Light llm et de recherche en sécurité. https://github.com/ModelTC/LightLLM
COMP011	LLM serving Cloud (scalable GPU)	Hébergement scalable pour modèles IA avec GPU.	https://github.com/vllm-project/vllm
COMP012	LLM serving On-prem/DR (Gpu peu scalable)	Capacité de servir quelques modèles standard selon une architecture on-premise (peu élastique)	Service accessible via MirAI. {nécessite échange préalable avec l'équipe sur les conditions d'utilisations)
COMP013	LLM serving expérimentation GPU fixe	Capacité de servir un volume nombre de modèles d'IA ouvert pour l'expérimentation et le développement. Le passage d'un modèle à l'autre peut entrainer des temps de chargement et de déchargement de modèle	Service accessible via MirAI. {nécessite échange préalable avec l'équipe sur les conditions d'utilisations)
COMP014	Référentiels vectorisés (ex: données réglementaire)	Service fournissant directement (ou proxifiant) l'accès à des corpus de données indexées via API, tel que par exemple légifrance. Les corpus sont vectorisés et peuvent être consommée par un RAG ou une application.	en reflexion. Echange interministériel en cours
COMP015	Données de réentrainement / tuning modèles	Permet de persister les appréciations d'usage et de pertinence des réponses par exemple pour un tuning de modèle, des statistiques de	en reflexion. Echange interministériel en cours

		performance ou de dérive des modèles, etc…	
COMP016	LLM proxying (catalog)	Permet de proxifier et suivre l'usage (ex: mener des refacturation au token) de manière transparente vers des providers externes.	
COMP017	LLM providers externes	Fournisseur d'API IA externes tel que Mistral, Scaleway, etc…	en reflexion. Echange interministériel en cours
COMP018	Observabilité / supervision	Service d'observabilité et d'alerte sur les applications et le socle Mirai	solution déployée a partir de la stack cloud pi native.
COMP019	Dev/Sec/AI/Ops (pax)	Service DevSecOps permettant de mettre en place les pipeline pour les applications. Cf Cloud Pi Native.	cf https://github.com/cloud-pi-native
COMP020	Infrastructure Pi native (on-premise)	Offre de service de compute sous kubernetes managés par le ministère de l'Intérieur ou redéployé par l'usager sous sa responsabilité à partir du commun numérique.	offre de service Interministérielle PI Native. GPUs en approvisionnement
COMP021	Services de socle	Permet l'envoie de messages, l'accès à des services réseaux, etc…	offre de service Interministérielle PI (SMTP, Proxy de sortie, poste d'administration, NTP, etc...).
COMP022	Gestion des mandats	Gestion des mandats RPA et des tokens d'autorisation agentiques, assurant la délégation sécurisée des droits, le contrôle des accès, la traçabilité et la révocation des autorisations.	*en conception*
COMP023	Hub de notifications	Permettre la publication / sub d'evènement et de gestion de campagnes / annonces	https://github.com/IA-Generative/hub-notification
COMP024	Couche d'acces AI	Permet d'assurer la non fuite de donnée, la gestion des accès au llm	https://github.com/protectai/llm-guard
COMP025	ZeroTrustProxyIA	Permet de contribuer à la maitrise des risques sur le fonctionnement agentique avec une observation/filtrage selon le niveau de risque	https://github.com/protectai/llm-guard

Enjeux lié à l'IA responsable et aux biais algorithmiques

Audience : ce paragraphe s'adresse au développeurs d'applications IA

L'IA responsable repose sur des principes éthiques et techniques garantissant une utilisation fiable, transparente et équitable des technologies d'intelligence artificielle.

Un enjeu majeur est la gestion des **biais algorithmiques**, qui peuvent résulter de données d'entraînement déséquilibrées, de modèles mal calibrés ou de décisions opaques.

Ces biais peuvent entraîner des discriminations involontaires et remettre en question la légitimité des décisions prises par l'IA.

Il est donc essentiel de mettre en place des mécanismes de **détection, d'évaluation et de correction des biais** dès la phase de conception des modèles.

Cela passe par des méthodologies comme l'audit des jeux de données, l'application d'algorithmes d'équité et l'intégration de métriques de diversité et d'inclusivité.

Par ailleurs, la **traçabilité et l'explicabilité des décisions** sont fondamentales pour renforcer la confiance des utilisateurs et assurer la conformité réglementaire (IA Act, RGPD).

L'implémentation de systèmes de **suivi des performances et des dérives** des modèles IA permet d'anticiper les évolutions des algorithmes et d'ajuster les décisions en fonction des nouvelles données.

Enfin, une gouvernance claire, incluant des comités d'éthique et une validation humaine des résultats critiques, doit être intégrée dans le cadre de développement et de déploiement du socle technique IA pour garantir un usage responsable et aligné avec les exigences sociétales et légales.

Enjeux liés à la sécurité & RPA & Gestion des mandats

Lorem lipsum

Spécificités à prendre en compte autour de la qualité et de la sécurité des applications

L'objectif d'ensemble est de s'assurer que le code produit est de qualité constante ou accrue, exempt de vulnérabilités algorithmiques ou importées néfastes.

Pour atteindre ces objectifs plusieurs mécanismes doivent être mis en place par l'équipe de développement intégrée :

- minimiser la portion spécifique de code développés en s'appuyant sur le catalogue de service proposé.
- mettre en place une couverture de test unitaire complète du back-end (et fournir les moyens de vérification automatisé à la chaîne secondaire)

- mener une analyse de code systématique le plus tôt possible (les langage et IDE modernes fournissent des fonctions de ce type)
- mener une analyse de CVE des dépendants importées

La chaîne secondaire reconstruit les images à partir des codes sources et procède aux mêmes tests avec des outils complémentaires. L'orchestration du pipeline secondaire est gérée par l'équipe et intègre les tests de vérification issue de la démarche d'homologation de l'application qui fixe les seuils de blocage de déploiement.

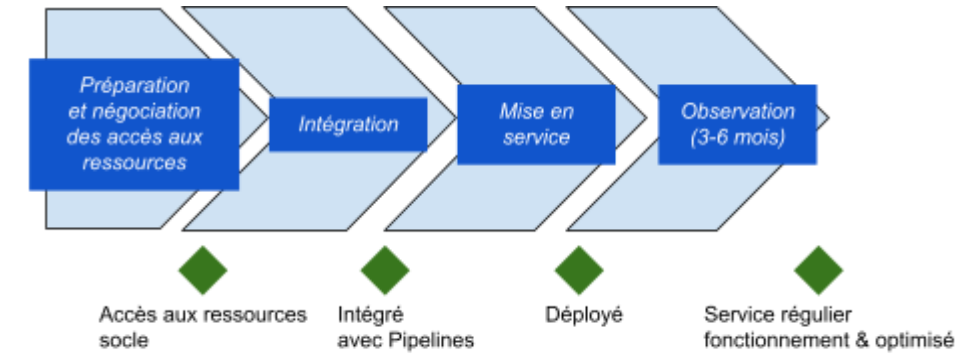
Les tests typiques consistent à vérifier la qualité du code (et la bonne couverture des tests), et le bon fonctionnement de l'application (non régression) et le scan de vulnérabilité.

L'équipe de développement reçoit via l'interface "shift left" une notification des rapports qui doit être intégrée au flux de travail pour correction.

L'équipe intégrée est invitée à mener une activité constante de refactoring du code produit et du suivi des vulnérabilités de sécurité.La chaîne secondaire est susceptible de bloquer les déploiements si la qualité d'ensemble du code est en baisse ou que le scan fait remonter des Cves critiques. L'équipe est invitée à vérifier et prendre en compte également le résultat des scans de vulnérabilité après la dernière version stable déployée de l'application et de corriger les Cve critiques et importantes.

L'équipe prend en compte que si le suivi des plans d'action n'est pas mis en oeuvre et que de surcroît des vulnérabilités critiques sont détectées depuis le dernier déploiement stable,et que l'équipe projet n'a pas pris en compte les injonctions de correction, l'application est susceptible d'être suspendue jusqu'à la remédiation pour garantir l'intégrité et la protection de ses données

Modèle d'intégration d'une application dans le cadre de Mirai



(à confirmer pendant les phases pilotes)

La phase préparation et négociation des accès aux ressources consiste à mettre en place les convention d'usage des services du socle Mirai et Pi Native afin de fournir un premier modèle de coût et permettre à l'équipe d'architecture efficacement l'application

Les phases d'intégration et de mise en service correspondent aux activités classiques d'un produit / projet avec par exemple la mise en place des pipelines, mesure de la qualité, mise en place de l'observabilité et de l'alerting, sélection du modèle IA et des référentiels nécessaire, etc..

La phase d'observation permet de confirmer que l'opportunité bénéfice coût est optimal, elle inclut côté projet des activités d'optimisation des modèles et d'organisation / processus.

4 - Présentation du socle MirAI ses évolutions pressenties

La composition de l'offre est amenée à évoluer en termes de catalogue de service selon la demande et les financements disponibles, ces évolutions permettent la diminution de la quantité de code produit par les équipes de développement et l'accélération des performances, typiquement : fonctions as services, services managés, gpu, modèles et services à disposition.

La construction du socle Mirai est opéré selon le cadre SAFe selon un cadencement tous les 3 mois environ et coordonné avec celui du socle Pi Native.

5 - Référentiel d'exigences et modalités d'usage

Les exigences du CCT sont classées en 2 niveaux d'exigence (périmètre du Ministère de l'Intérieur) :

- Primordial : L'exigence est impérative et traitée administrativement.
- I – Important : Exigence prise en compte pour la notation technique de la solution

Précisions sur le cas de l'exclusion administrative (périmètre du Ministère de l'Intérieur) :

- La non-conformité au cadre de norme entraîne l'exclusion administrative lors du dépouillement et la mise en œuvre des actions de remédiation du marché lors de l'exécution du marché.
- La non-conformité aux exigences d'architecture entraîne l'impossibilité d'utilisation du socle de sécurité associé à l'offre Cloud Native

Par défaut les règles du CCT s'imposent. Elles peuvent être précisées dans le cas d'un appel d'offres dans le règlement de consultation pour le dépouillement des offres et dans le CCAP pour l'exécution du marché. Une demande de dérogation est possible. (cf paragraphe ad hoc)

6 - Annexes

Les normes industrielles, institutionnelles applicables

La conception de système d'information dans le cadre de l'État est encadrée par un ensemble de recommandations ou règles à mettre en œuvre. Ces normes sont citées ci-dessous. Le lecteur est invité à vérifier qu'il dispose des versions les plus à jour.

Norme industrielle	**Kubernetes : https://kubernetes.io/fr/** **ArgoCD : https://argo-cd.readthedocs.io/en/stable/**
Guides & outils pour la conception	**DSFR : Design System FR. La charte internet de l'État (qui intègre le RGAA)** **https://www.systeme-de-design.gouv.fr/** **Guide d'éco conception :** **https://ecoresponsable.numerique.gouv.fr/publications/referentiel-general-ecoconception/** **Divers guides de conceptions logiciels:** **https://guides.etalab.gouv.fr** **https://catalogue.numerique.gouv.fr** **https://schema.gouv.fr** **https://code.gouv.fr**
Cadres de pratiques de conception et de conduite de projet agile	https://www.numerique.gouv.fr/actualites/guide-pour-allier-agilite-et-securite-numeriques/
Logiciel libre	Socle InterMinistériel des Logiciels Libres (SILL) de par sa fonction de source pour le référentiel de produits du CCT Ministériel : https://sill.etalab.gouv.fr/fr/software
Normes interMinistérielles de conception de solutions	**Doctrine cloud de l'état :** **https://www.legifrance.gouv.fr/circulaire/id/45205** **Référentiel Général d'Accessibilité pour les Administrations :** **https://accessibilite.numerique.gouv.fr/**
Référentiel Général de Sécurité, en association avec le règlement européen et l'EIDAS.	https://www.ssi.gouv.fr/entreprise/reglementation/confiance-numerique/liste-des-documents-constitutifs-du-rgs-v-2-0/

Référentiel Général de Gestion des Archives	https://www.ssi.gouv.fr/entreprise/reglementation/confiance-numerique/le-reglement-eidas/
Référentiel Général de Gestion des Archives	https://francearchives.fr/fr/circulaire/R2GA_2013_10
règlement européen sur la protection des données personnelles	https://www.cnil.fr/fr/reglement-europeen-protection-donnees
L'IA ACT (réglement européen)	https://digital-strategy.ec.europa.eu/fr/policies/regulatory-framework-ai https://www.cnil.fr/fr/entree-en-vigueur-du-reglement-europeen-sur-lia-les-premieres-questions-reponses-de-la-cnil

Liens vers autres contenus utiles(informatif)

https://kubernetes.io/fr/

https://www.rancher.com/products/k3s

https://www.redhat.com/en/technologies/cloud-computing/openshift

https://argo-cd.readthedocs.io/en/stable/

https://www.redhat.com/en/topics/microservices/what-is-a-service-mesh

https://www.redhat.com/en/topics/devops/what-is-gitops

https://www.cloudcomputingpatterns.org/

https://12factor.net/fr/

https://tanzu.vmware.com/content/blog/beyond-the-twelve-factor-app

https://www.techworld-with-nana.com/devops-bootcamp

https://ecoresponsable.numerique.gouv.fr/publications/bonnes-pratiques/bonnes-pratiques/#bonnes-pratiques-services-numeriques

La documentation sur le CloudPI :

https://pi.interieur.rie.gouv.fr/ (accessible via le rie uniquement)

https://cloud-pi-native.fr/

https://github.com/Cloud-pi-Native/

Glossaire

Terme	Description / définition
Agile	Une méthode agile est une méthode de développement informatique permettant de concevoir des logiciels en impliquant au maximum le demandeur (client), ce qui permet une grande réactivité à ses demandes. Les méthodes agiles se veulent plus pragmatiques que les méthodes traditionnelles. Elles visent la satisfaction réelle du besoin du client, et non d'un contrat établi préalablement. La notion de méthode agile est née à travers un manifeste signé par 17 personnalités (parmi lesquelles Ward Cunningham, l'inventeur du Wiki), créateurs de méthodes ou dirigeants de sociétés. (Source : https://www.techno-science.net/definition/743.html)
ADR	Enregistrement des décisions d'architecture suivant le modèle MADR
API	Une API est le moyen « standard » désormais, par lequel est exposée une ressource, afin d'en permettre l'accès. Le qualificatif « RESTFULL » renvoie à la conformité de l'API au modèle « REST » qui est un modèle de représentation de l'URL de l'API. Une API est assortie d'un contrat de service qui décrit son fonctionnement. Ce contrat doit être conforme au standard « OPEN API V3 » et accessible aux développeurs.
API Management	Processus de gestion de la totalité du cycle de vie d'une API, de son idéation jusqu'à son retrait de service. Décrit dans la Stratégie d'API. Une plateforme d'exposition d'API existe à la DNUM : api.minint.fr , ainsi qu'une autre, de niveau interMinistérielle : api.gouv.fr
BATN	Bureau Appui à la Transformation Numérique
CaaS	Les CaaS ou Containers en tant que Service (Containers as a Service en anglais) sont une catégorie de services Cloud permettant aux développeurs de logiciels de télécharger, d'organiser, d'exécuter, de gérer, de mettre à l'échelle et d'arrêter des containers en utilisant l'interface web ou l'API d'un fournisseur. Source : www.lebigdata.fr
CCAP	Cahier des Clauses Administratives Particulières
CCT	Cadre de Cohérence Technique du MIOM.
CCTP	Cahier des Clauses Techniques Particulières
CCU	Cadre Commun d'Urbanisation

CERFA	Centre d'Enregistrement et de Révision des Formulaires Administratifs
CHAP	Challenge Hash Authentication Protocol
CI/CD	CI/CD signifie « Continuous Integration/Continuous Delivery », ou intégration continue/livraison continue ; c'est une méthode de mise en œuvre logicielle utilisée par les équipes de développement pour apporter des modifications de code plus fréquentes et plus fiables. Le CI/CD englobe deux ensembles de pratiques complémentaires, chacune reposant fortement sur l'automatisation. (Source : https://www.splunk.com/fr_fr/data-insider/what-is-ci-cd-pipeline.html#:~:text=CI%2FCD%20signifie%20%C2%AB%20Continuous%20Integration,plus%20fr%C3%A9quentes%20et%20plus%20fiables.)
Client	Dans une architecture client-serveur, le client est celui qui est à l'initiative des requêtes faites au serveur
Cluster	Cluster (grappe) : plusieurs systèmes sont interconnectés soit pour augmenter la puissance de calcul (on parle alors de cluster de performance), soit pour offrir une tolérance de pannes accrue par la redondance des composants unitaires (on parle alors de cluster de haute disponibilité). Dans les deux cas, pour bénéficier de l'architecture en grappe, il faut que les applications aient été conçues en conséquence ou que le système d'exploitation, le compilateur et les logiciels sous-jacents (bases de données, middlewares, etc.) prennent en charge les fonctions adéquates de parallélisation des traitements ou de reprise sur incident.
Conteneur	Les conteneurs sont des unités exécutables de logiciel dans lesquelles le code d'application est empaqueté, avec ses bibliothèques et ses dépendances, de manière commune, afin qu'il puisse être exécuté n'importe où, que ce soit sur un ordinateur de bureau, dans un système informatique traditionnel ou dans le cloud. (Source : https://www.ibm.com/fr-fr/cloud/learn/containers#:~:text=Les%20conteneurs%20sont%20des%20unit%C3%A9s,traditionnel%20ou%20dans%20le%20cloud.)
CVE	Common Vulnerabilities and Exposures. ***Common Vulnerabilities and Exposures*** ou **CVE** est un dictionnaire des informations publiques relatives aux vulnérabilités de sécurité. Le dictionnaire est maintenu par l'organisme MITRE, soutenu par le département de la Sécurité intérieure des États-Unis.

DevSecOps (DSO)	Le DevSecOps inclut la composante sécurité (security) dans l'approche DevOps, qui lie elle-même le développement (développement) et l'exploitation (opérations). Le DevSecOps s'instancie par des mesures de formation, organisationnelles et des ajouts de système de vérification à chaque fois que l'application est construite.
DINUM	Direction Interministérielle du Numérique
DITP	Direction InterMinistérielle de la Transformation Publique (placée sous l'autorité du ministre de l'Action et des Comptes publics, chargée de la réforme de l'État).
DNUM	Direction du Numérique
DSFR	Design System FR. La charte internet de l'État.
ENT(A)	Environnement Numérique de Travail (de l'Agent)
FIP	Factory Instrumental Protocol (Flux d'Information Processus). Actuellement FIP est une norme française (NF C46 601 à NF C46 607) et une norme européenne (EN 50170-3). La promotion et une part d'assistance technique de ce réseau sont effectuées par l'organisation WorldFIP dont le siège se trouve en France. La cible privilégiée de WorldFIP est l'interconnexion de capteurs, actionneurs et automates dans les systèmes automatisés. Comme la quasi-totalité des réseaux de terrain WorldFIP a une structure en trois couches. (https://www.i3s.unice.fr/~map/Cours/LPIREEL/COURS3FIP.pdf)
Gitops	L'approche GitOps repose sur l'utilisation de référentiels Git comme unique source de vérité pour distribuer l'infrastructure en tant que code. Le code envoyé vérifie le processus d'intégration continue, tandis que le processus de distribution continue vérifie et applique les exigences relatives à certains aspects, comme la sécurité, l'infrastructure en tant que code (IaC), ou toute autre limite fixée pour le framework d'application. Toutes les modifications apportées au code font l'objet d'un suivi, ce qui facilite les mises à jour et le contrôle de versions en cas de restauration. (Source : https://www.redhat.com/fr/topics/devops/what-is-gitops#:~:text=Le%20GitOps%20peut%20%C3%AAtre%20consid%C3%A9r%C3%A9,les%20configurations%20de%20l'infrastructure.)
Hébergement	L'hébergement, dans son sens le plus générique, est un service par lequel des ressources de stockage et de calcul sont fournies à une personne ou à une organisation pour l'hébergement et la maintenance d'un ou plusieurs sites Web et services connexes.

	(Source : https://definir-tech.com/hebergement/)
IAM	Gestion des identités et des accès (Identity and Access Management)
Java	Java est un langage de programmation et une plate-forme de calcul lancé par Sun Microsystems en 1995. (Source : https://www.java.com/fr/download/help/whatis_java.html)
Kubernetes	Kubernetes est une plate-forme open-source extensible et portable pour la gestion de charges de travail (workloads) et de services conteneurisés. Elle favorise à la fois l'écriture de configuration déclarative (declarative configuration) et l'automatisation. C'est un large écosystème en rapide expansion. (Source : https://kubernetes.io/fr/docs/concepts/overview/what-is-kubernetes/#:~:text=Kubernetes%20est%20une%20plate%2Dforme,large%20%C3%A9cosyst%C3%A8me%20en%20rapide%20expansion.)
Logiciel Libre	Un logiciel libre est un logiciel dont la licence dite « libre » donne à chacun le droit d'utiliser, d'étudier, de modifier, de dupliquer, de donner et de vendre ledit logiciel sans contrepartie. La notion de logiciel libre ne doit se confondre ni avec celle de logiciel gratuit (freeware ou graticiels) ni avec celle de shareware (partagiciels). De même, les libertés octroyées par la licence d'un logiciel libre sont plus étendues que le simple accès au code source, ce qu'on appelle parfois logiciel « à sources ouvertes ».
MCO	Maintien en Condition Opérationnelle
MI	Ministère de l'intérieur
NFR	Non Functional Requirement. Exigence non fonctionnelle que doit embarquer l'équipe. Généralement des exigences d'architectures, de sécurité, etc…
MIOM	Ministère de l'intérieur et des Outre-Mer
Node	NodeJS est une plateforme construite sur le moteur JavaScript V8 de Chrome qui permet de développer des applications en utilisant du JavaScript. Il se distingue des autres plateformes grâce à une approche non bloquante permettant d'effectuer des entrées/sorties (I/O) de manière asynchrone. (Source : https://grafikart.fr/tutoriels/nodejs-intro-792)
Open API	Une API ouverte, parfois appelée API publique, est une interface de programmation d'application (Application Programming Interface) qui permet au développeur d'accéder à une application logicielle propriétaire par voie de programmation. (https://www.lemagit.fr/definition/API-ouverte#:~:text=Une%20API%20o

	uverte%2C%20parfois%20appel%C3%A9e,propri%C3%A9taire%20par%20voie%20de%20programmation.)
Openshift	OpenShift est un service de plate-forme en tant que service de la société Red Hat qui permet de déployer des projets dans des containers. Pour ce faire, OpenShift utilise les technologies Docker et Kubernetes. (https://fr.wikipedia.org/wiki/OpenShift)
Open Source	La désignation Open Source (« source ouverte » en français) s'applique aux logiciels dont la licence respecte des critères précisément établis par l'Open Source Initiative, c'est-à-dire la possibilité de libre redistribution, d'accès au code source, et de travaux dérivés. Les logiciels Open Source et les logiciels libres désignent les mêmes logiciels, ceux dont la licence est reconnue libre par l'Open Source Initiative ou la Free Software Foundation. Le terme « Open Source » est en concurrence avec le terme « logiciel libre » (Free Software) recommandé par la FSF. Le terme Freeware (graticiel) désigne des logiciels gratuits qui ne sont ni nécessairement ouverts, ni libres. (Source : https://fr.wikipedia.org/wiki/Open_source)
Objet Métiers	Représentation schématique d'un concept métier, instanciée sous la forme d'une donnée gérée par un système d'information maître et de l'organisation qui à la charge de maintenir cette donnée à jour et exempte d'erreur.
Pods	Un pod représente une collection de conteneurs d'applications et de volumes fonctionnant dans le même environnement d'exécution. Les pods, et non les conteneurs, sont les plus petits artefacts déployables dans un cluster kubernetes. Les applications s'exécutant dans le même pod partagent la même adresse IP et le même espace de nom réseau. Source : Kubernetes maitrisez l'orchestrateur des infrastructures du futur
PP	
Proxy	Un serveur proxy est une sorte de pont qui vous relie au reste d'Internet. Normalement, lorsque vous naviguez sur Internet, vous vous connectez directement au site Web qui vous intéresse. Un proxy établit à votre place la communication avec le site Web. (Source : https://www.avast.com/fr-fr/c-what-is-a-proxy-server#:~:text=Un%20serveur%20proxy%20est%20une,communication%20avec%20le%20site%20Web.)

Python	Le langage Python est un langage de programmation open source multi-plateformes et orienté objet. Grâce à des bibliothèques spécialisées, Python s'utilise pour de nombreuses situations comme le développement logiciel, l'analyse de données, ou la gestion d'infrastructures. (Source : https://www.futura-sciences.com/tech/definitions/informatique-python-19349/)
RACI	L'acronyme RACI (responsible, accountable, consulted et informed) ou RAM (responsibility assignment matrix) désigne dans le domaine du management une matrice des responsabilités. Elle indique les rôles et les responsabilités des intervenants au sein de chaque processus et activité. La matrice RACI donne une vision simple et claire de qui fait quoi dans le projet, en permettant d'éviter une redondance de rôles ou une dilution des responsabilités. (Source : https://fr.wikipedia.org/wiki/RACI)
R2GA	Référentiel Général de Gestion des Archives sur le portail national des archives
Restful API	Une API REST (également appelée API RESTful) est une interface de programmation d'application (API ou API web) qui respecte les contraintes du style d'architecture REST et permet d'interagir avec les services web RESTful. REST (Representational State Transfer). (Source : https://www.redhat.com/fr/topics/api/what-is-a-rest-api#:~:text=Une%20API%20REST%20(%C3%A9galement%20appel%C3%A9e,avec%20les%20services%20web%20RESTful.)
RGAA	Référentiel Général d'Accessibilité pour les Administrations
RGI	Référentiel Général d'Interopérabilité
RGPD	Le sigle RGPD signifie « Règlement Général sur la Protection des Données ». Le RGPD est le règlement européen sur la protection des données, articles de lois sur l'accessibilité des données à des fins de recherche
RGS	Référentiel Général de Sécurité, en association avec le règlement européen eIDAS

Rootless	Les conteneurs rootless font référence à la possibilité pour un utilisateur non privilégié de créer, d'exécuter et de gérer des conteneurs. Ce terme inclut également la variété d'outils autour des conteneurs qui peuvent également être exécutés en tant qu'utilisateur non privilégié. "Utilisateur non privilégié" dans ce contexte fait référence à un utilisateur qui n'a aucun droit d'administrateur et qui n'est "pas dans les bonnes grâces de l'administrateur" (en d'autres termes, il n'a pas la possibilité de demander que plus de privilèges lui soient accordés à eux, ou pour les progiciels à installer). (Source : https://rootlesscontaine.rs/)
SADC (CDS)	Service d'accès au data center. Correspond à la chaîne de protection et de pollution des des flux entrants vers le data center. Cela peut être appelé parfois "chaîne de service" (CDS).
SDID (ex SDITN)	Sous-Direction à l'Innovation et Données de la Direction de la transformation numérique du MIOM. (ex: sous direction de l'innovation et de la transformation numérique)
Shift left	Le Shift Left décrit un principe qui consiste à rendre les flux de travail des entreprises plus efficaces, grâce à des tests et avec des suivis précoces. Cette méthode vous permet de transmettre les connaissances de votre service d'assistance rapidement et facilement à tous les employés de votre entreprise. (Source : https://freshservice.com/fr/shift-left-blog/#:~:text=Le%20Shift%20Left%20d%C3%A9crit%20un,les%20employ%C3%A9s%20de%20votre%20entreprise.)
SI	Selon la définition restreinte donnée par Joël de Rosnay, « Un système est un ensemble d'éléments en interaction dynamique, organisés en fonction d'un but ». Le système d'information n'échappe pas à cette définition. Il est un ensemble dont les éléments sont les constituantes de toute organisation (entreprise, administration, association, groupement, …). Ces éléments sont de plusieurs natures : organisationnelle, informationnelle, métier, technique, technologique. Tous ces éléments forment un tout (plus ou moins cohérent) et participent à la réussite de l'organisation dans son objectif.
SIC	Systèmes d'Information et de Communication
SIEM	Security information and event management. Dans le domaine de la sécurité informatique, les produits et service logiciels de SIEM combinent la gestion des informations de sécurité (SIM) et la gestion des événements de sécurité (SEM) pour fournir des alertes en temps réel.

	Cet outillage est mis en œuvre un centre de supervision de la sécurité (Security Office Center : SOC)
SILL	Socle interministériel de logiciels libres. Il regroupe l'ensemble des logiciels libres préconisés au sein des ministères. Il est alimenté par des agents publics volontaires Ministériels, sous le contrôle de la DINUM
SPOC	Single Point Of Contention : littéralement « point individuel de contention ». Consiste dans un système à identifier, pour les différents composants (matériels et/ou logiciels), l'existence de points constituant un goulet d'étranglement. Ce composant est alors considéré comme un SPOC pour le système.
SPOF	Single Point Of Failure : littéralement « point individuel de défaillance ». Consiste dans un système à identifier, pour les différents composants (matériels et/ou logiciels), l'existence de points de défaillance pouvant générer un dysfonctionnement du système de par l'impossibilité de redonder ce composant ou de par le choix de ne pas le redonder. Ce composant est alors considéré comme un SPOF pour le système.
ST(SI)²	Service des Technologies et des Systèmes d'Information de la Sécurité intérieure. Le ST(SI)² est chargé de concevoir, de piloter et de conduire les projets liés aux systèmes d'information, de communication et de commandement pour l'ensemble des policiers et des gendarmes. Il contribue à la définition de l'action, de la stratégie et de la politique de sécurité du ministère de l'intérieur en matière de système d'information et de communication. Il coordonne les services SIC de proximité de la police et de la gendarmerie. Il anime la politique d'innovation technologique du ministère dans ce domaine. (Source : https://fr.linkedin.com/company/stsisi)
Swagger	Une définition Swagger spécifie un ensemble de métadonnées qui décrivent une API REST. Si vous avez un fichier Swagger définissant une API REST, vous pouvez l'ajouter à votre projet en tant que source de synchronisation externe. Cette source peut être synchronisée avec le projet. (Source : https://www.ibm.com/docs/fr/rtw/9.0.1?topic=testing-swagger-definitions)
TCP	TCP (Transmission Control Program) est un protocole permettant l'ouverture de circuits virtuels entre applications

VM	Une machine virtuelle, ou « virtual machine », est « le client » créé dans un environnement informatique, « l'hôte ». Plusieurs machines virtuelles peuvent coexister sur un seul hôte. Les principaux fichiers qui constituent une machine virtuelle sont un fichier journal, un fichier de paramètres de RAM non volatile, un fichier de disque virtuel et un fichier de configuration. (Source : https://www.vmware.com/fr/topics/glossary/content/virtual-machine.html)
VPN	Virtual Private Network, réseau privé virtuel (RPV) : Le principe du RPV consiste à créer un réseau privé au sein d'un réseau public. Cette démarche existe depuis longtemps : les opérateurs s'en servent pour gérer les lignes privées de leurs clients au sein des mêmes « tuyaux ». Aujourd'hui, on parle surtout de réseaux privés virtuels sur Internet. Les RPV mettent en œuvre des mécanismes de contrôle d'accès (authentification des utilisateurs) et assurent la confidentialité des données (cryptographie). Le terme de réseau privé virtuel s'applique aussi au réseau téléphonique : les opérateurs font ainsi transiter sur le réseau public des services évolués de téléphonie jusque-là cantonnés au réseau privé de l'entreprise appel en numérotant uniquement l'extension, renvoi d'appel, conversation à plusieurs, etc. Cette technologie s'étend aussi aux mobiles.
Windows Server	Windows Server est un système multi-tâches, multi-utilisateurs qui dans ses fonctionnalités peut se comparer au système UNIX/Linux. Il présente l'avantage que certains logiciels soient moins chers que leur équivalent fonctionnant sous UNIX, et plus rarement, Linux. Par ailleurs, la quasi-totalité des éditeurs propose des versions de leurs produits pouvant tourner sur serveur Windows.

--- fin du document ---

7 - Référentiel d’exigences applicables aux applications Mirai

Note: le terme développeur est générique et fait référence à l’individu ou l’organisation pluridisciplinaire qui est chargée de produire et maintenir : la base de code, le corpus de tests et les fichiers de description d’infrastructure et les documentation technique et usager.
Il est responsable de l'adéquation et de la qualité de la solution au besoin des usagers en collaborant de manières étendues avec les autres acteurs impliqués.

Version : VERSION EN COURS D’ELABORATION

ID	Type	Exigence	Catégorisation
Version		Version : liste des exigences, initialisé le 17 mars 2025 change management : Exigence P : primordiale, le non respect entraîne un rejet administratif de l'offre lors d'un appel d'offre et un plan de remédiation obligatoire lors de l'exécution du marché Exigence I : importante, permet de maximaliser la qualité de la solution	-
		---	
EXI-G-1	I	Dans le cadre d’un appel d’offre, en cas d’incohérence entre les documents et hors mentions explicites, le cct et la liste des exigences associées sont de niveaux supérieures.	hiérarchie des normes
EXI-G-2	I	Le respects des exigences du CCT MIRAI sont permanentes pour la direction d’application métier.	applicabilité

EXI-G-3	I	Respect des standards et des normes applicables industrielles, européennes et étatiques, pour la conception de solutions numériques hébergées dans le cloud native (kubernetes), Design Système de l'État, RGAA, RGS, RGI, doctrine Cloud au centre, Et IA ACT. cf paragraphe du volet de CCT : *Les normes industrielles, institutionnelles applicables*	Cadre de normes
EXI-G-4	P	Respect du guide d'éco-conception, optimisation des ressource de calcul haute performance et choix de solution performantes. La conception frugale vis à vis des ressources d'infrastructures consommées et l'impact vis-à-vis du terminal de l'usager. S'appuyer sur les services API mutualisées proposée par l'offre de service Mirai. (ils ont été conçu ou sélectionné optimisé) Dans le cas d'utiisation d'un modèle d'IA /LLM spécifiquement paramétré pour le cas d'utilisation, la direction d'application justifie son choix et précise les mesures qui seront prises pour optimiser la consommation des ressources. (ex: choix d'un hébergement avec un PUE optimisé)	Efficience de la solution
EXI-G-5	**P**	Choix d'un hébergement adapté à la nature des données manipulées (usuel / DR) et selon le cadre légal adapté. Cloud Pi (on-prem), Cloud public, Infrastructure dédiée ou mixte.	Infrastructure
EXI-G-6	**P**	L'application et le métier doivent veiller à la balance bénéfice vs coût d'utilisation de technologie IA. L'équipe doit rechercher systématiquement une solution moins coûteuse en repensant par exemple le modèle organisationnel, expérimenter des alternatives. Un point après 6 mois d'utilisation devra être proposé pour confirmer l'opportunité de gain vs coût de la solution	Infrastructure
EXI-G-7	P	Un modèle estimatif d'inducteur de coût consommation de calcul haute performance est établi par utilisateur ou par transaction (en moyenne). Il peut être exprimé en nombre de token LLM ou en nombre de GPU	Finops
EXI-G-8	**I**	L'application doit intégrer le service du socle Mirai permettant d'identifier de manière déclarative ou calculée le temps "gagné" et l'utilité de l'application	Gain efficacité de la solution
EXI-G-9	**P**	L'application intègre le service permettant de capitaliser la qualité des réponse afin de suivre la performance des modèles et permettre le cas échéant le réentrainement	Performance et biai des modèles

EXI-DATA-2 (Pi-native)	**P**	Pour la persistance de données personnelles soumises au RGPD, le modèle de données intègre dès la conception, un tag RGPD, des champs dupliqués dédiés à l'anonymisation et des règles et processus d'anonymisation ainsi qu'une politique de droits associés.	Données
EXI-DATA-3 (Pi-native)	I	Consommation systématique des données de référence ministérielles et offre de service mutualisées	Données
EXI-DATA-4 (Pi-native)	I	Référencement des objets métiers dans le catalogue des données métiers	Données
--- reprise des exigences clés Pi Native --			
EXI-ORG-1(Pi-native)	**P**	Conformité au modèle de responsabilité Cloud Pi Native : les développeurs/concepteurs sont responsables de la partie développement, du maintien en continu d'une qualité constante de la solution et l'absence de vulnérabilités exploitables notamment avant toute mise en production.	Modèle d'Opération

EXI-ARCH-8(Pi-Native)	**I**	Le déploiement est réalisé via gitops (argoCD) L'architecture de déploiement d'application est organisée pour déployer indépendamment les modules de l'application, sans couplage entre-eux. Typiquement l'application est découpée en une application maître d'ensemble référençant le déploiement de sous-applications. L'ensemble est géré en version dans un git. Chaque changement est réalisé via un git push ou via pull request si l'équipe a mis en place une relecture collective du code de version	
EXI-ARCH-9 (Pi-native)	**I**	L'équipe met en place une stratégie gitops et "d'immutability" de la configuration de l'infrastructure Chaque nouveau déploiement recrée un environnement nouveau ("écrase" l'ancien). Le déploiement met en oeuvre une stratégie de reprise de données applicative à chaque nouveau déploiement. (la sauvegarde des données applicatives doit être effectuée régulièrement, juste avant le nouveau déploiement) A titre d'exemple, l'opérator cnpg de postgres (https://cloudnative-pg.io/) permet de mettre en place ce type de pratique.	
EXI-ARCH-10 - (Pi-native)	**P**	Consommation systématique des services socles. (chaîne de service, IAM, Vault, etc...) Dans le cas où plusieurs services sont susceptibles d'être utilisés, privilégier les services dont l'enrôlement est automatisé.	Services Mutualisés
EXI-ARCH-11 (Pi-native)	**P**	Identification utilisateur : l'application doit obligatoirement utiliser le SSO Agent disponible et/ou France connect pour les citoyens	Architecture