

# Résumé des exigences pour l'hébergement

Cette page résume les exigences pour que votre application web soit hébergée à l'Ifremer sur l'infrastructure Docker, que celle-ci soit exposée sur Internet ou accessible uniquement en intranet.

Un niveau est associé à chaque exigence :

- **Obligatoire** : exigence devant être impérativement respectée.
- **Préconisé** : exigence qu'il est préférable d'appliquer mais qui ne revet pas un caractère Obligatoire.

Il est possible que cette liste ne soit pas totalement exhaustive.

Dans tous les cas, si vous avez un doute, contactez l'équipe infrastructure pour en discuter.

## Sommaire

- [Résumé des exigences pour l'hébergement](#)
  - [Organisation](#)
  - [Code source et génération des artefacts](#)
  - [Conception](#)
  - [Runtime](#)
  - [Logs](#)
  - [Volumes \(fichiers / répertoires persistés\)](#)
  - [Réseau](#)
  - [Authentification](#)
  - [Monitoring](#)
  - [Front](#)

## Organisation

ID	Niveau	Description
EX- ORGA- RESP- APPLI	Obligatoire	L'application dispose à tout moment d'un <b>responsable applicatif IFREMER en CDI</b> .

ID	Niveau	Description
EX-ORGA-FICH-SIGN	Obligatoire	L'application est décrite dans une <a href="#">fiche signalétique</a> .
EX-ORGA-FICH-PROJ	Obligatoire	Si l'application fait partie d'un projet conséquent, ce projet a fait l'objet d'une <b>fiche projet validée en amont</b> par la direction.
EX-ORGA-MCO	Obligatoire	Le responsable applicatif doit assuré le <b>Maintien en Condition Opérationnel (MCO)</b> de l'application <b>pendant toute sa durée de vie</b> , de la 1 <sup>ère</sup> mise en exploitation, jusqu'à son décommissionnement complet.
EX-ORGA-MCS	Obligatoire	<p>Le responsable applicatif doit assuré le <b>Maintien en Condition de Sécurité (MCS)</b> de l'application <b>pendant toute sa durée de vie</b>, de la 1<sup>ère</sup> mise en exploitation, jusqu'à son décommissionnement complet.</p> <p>Le MCS inclut la <b>fourniture régulière d'une image Docker à déployer en production intégrant les derniers patches de sécurité</b>, notamment :  - la mise à jour des dépendances de l'application (ex : librairies)  - la mise à jour de l'image Docker de base ayant servi à construire l'image. Par exemple, si le Dockerfile de l'application débute par <code>FROM debian:12-slim</code>, l'image Docker de l'application devra être rebuildée et redéployée régulièrement pour intégrer les correctifs de sécurité intégrés dans la dernière version de l'image <code>debian:12-slim</code>.</p>
EX-ORGA-EOL	Obligatoire	<p>Une <b>date de décommissionnement prévisionnelle</b> doit être fournie et indiquée dans la fiche signalétique.</p> <p>Cette date pose un jalon pour évaluer si l'application doit être maintenue (i.e. prolongation de la date de décommissionnement) ou si elle peut être supprimée.</p> <p>Cette durée prévisionnelle est de 5 ans au maximum sachant qu'elle pourra être reconduite.</p>
EX-ORGA-VAL	Obligatoire	<p>L'application doit être <b>testée en amont sur la plateforme de validation</b> pour vérifier son bon fonctionnement.</p> <p>Cet environnement est très proche de l'environnement de PROD.</p> <p>Ceci est d'autant plus nécessaire en amont de sa mise en exploitation initiale.</p>

## Code source et génération des artefacts

ID	Niveau	Description
EX-CODE-HEBERG	Obligatoire	<p>Le code source de l'application est <b>hébergé sur le GitLab de l'IFREMER</b> et <b>versionné</b>.</p> <p>Le projet ne doit pas être présent dans un groupe personnel de l'utilisateur.</p> <p>Le projet doit être positionné de préférence dans un sous-groupe lié à la thématique ou au département/service porteur de l'application.</p>

ID	Niveau	Description
EX-CODE-SECRET-ABS	Obligatoire	Le code source de l'application ne doit contenir <b>aucune secret</b> de validation ni de production.
EX-CODE-SECRET-LEAK	Obligatoire	Si par mégarde, un <b>secret de production est poussé par erreur sur le repository Git</b> , il faut notifier les responsables pour que celui-ci soit modifié en production car le secret restera à jamais présent dans l'historique du repository Git.
EX-CODE-VISIBILITE	Obligatoire	<p>Le code source de l'application doit garder la <b>visibilité la plus stricte possible</b>, notamment car l'instance GitLab est exposée sur Internet. Il est préconisé :</p> <ol style="list-style-type: none"> <li>1. de conserver une visibilité <b>PRIVATE</b> et de gérer la liste des membres autorisés à accéder au projet manuellement. Les membres sont soit directement associés au projet ou hérités des groupes parents.</li> <li>2. à défaut, d'utiliser une visibilité <b>INTERNAL</b> . ⚠ Le code source sera accessible par toute personne disposant d'un compte extranet (plusieurs milliers), ce qui inclut des personnes extérieures à l'IFREMER. ⚠</li> <li>3. en dernier recours, d'utiliser une visibilité <b>PUBLIC</b> . ⚠ 🦋 Ceci exposera le projet à n'importe qui sur Internet. 🦋 ⚠ Dans ce cas, ce choix doit être argumenté. En cas de visibilité <b>PUBLIC</b> , le code source ne devra <b>JAMAIS</b> avoir inclus des éléments renseignant un attaquant sur le SI de l'Ifremer (ex : nom de compte, nom de machines, chemin vers des espaces disques, etc...).</li> </ol>
EX-CODE-TAG-RELEASE	Obligatoire	Les artefacts déployés en validation/production doivent être <b>associés à un tag du code source</b> dans le repository Git.
EX-CODE-TAG-FORMAT	Préconisé	<p>Les versions des artefacts/tags suivent la norme <b>Semantic Versioning 2.0.0</b>.</p> <p>Étant donné un numéro de version <b>MAJEUR.MINEUR.CORRECTIF</b> , il faut incrémenter :</p> <ul style="list-style-type: none"> <li>- le numéro de version <b>MAJEUR</b> quand il y a des <b>changements non rétrocompatibles</b></li> <li>- le numéro de version <b>MINEUR</b> quand il y a des <b>ajouts de fonctionnalités rétrocompatibles</b></li> <li>- le numéro de version de <b>CORRECTIF</b> quand il y a des <b>corrections d'anomalies rétrocompatibles</b> (bugs fonctionnels, mises à jour de sécurité...) .</li> </ul>
EX-ARTEFACT-VERSION	Obligatoire	<p>La <b>version des artefacts</b> déployée en validation/production doit correspondre <b>strictement à la valeur du tag sur le code source</b>. A défaut, la valeur du tag sur le code source doit être inclus dans la version du livrable.</p> <p>Ceci permet d'associer facilement un livrable avec son code source de référence</p>

ID	Niveau	Description
EX-ARTEFACT-BUILD	Obligatoire	<p>Les artefacts déployés en validation/production doivent être <b>générés via la CI/CD de GitLab</b> en utilisant un tag sur le code source.</p> <p>Ceci permet de décrire techniquement une "recette" pour générer à nouveau un livrable.</p>
EX-ARTEFACT-STOCKAGE	Obligatoire	<p>Les artefacts déployés en validation/production doivent être <b>stockés dans les registries associées au serveur GitLab de l'Ifremer</b>. Il est possible d'activer des registries sur des groupes et des projets GitLab.</p> <p>Les images Docker sont stockées dans des Container Registries. Les packages (ex : fichier WAR) sont stockés dans des Package Registry</p> <p>Si un livrable ne peut être stocké dans l'une de ces registries, ce choix devra être argumenté et validé avec l'équipe infrastructure.</p>
EX-ARTEFACT-UNIQ	Obligatoire	<p><b>Un seul artefact</b> (ex : image Docker) doit être généré <b>pour l'ensemble des environnements</b> (validation/production). Les spécificités propres à un environnement seront fournies via une configuration externe.</p>
EX-ARTEFACT-FORMAT	Obligatoire	<p>L'application web doit être livrée sous l'une des formes suivantes :</p> <ul style="list-style-type: none"> <li>- une <b>image Docker</b></li> <li>- ou une <b>archive WAR</b> qui sera intégrée via des process automatiques Ifremer dans une image Docker contenant un serveur Tomcat. Seules les <b>dernières versions stables de Tomcat</b> sont supportées.</li> </ul>

## Conception

ID	Niveau	Description
EX-CONCEPT-12FACTOR	Obligatoire	L'application doit suivre au maximum les bonnes pratiques mentionnées dans les <b>12-factors</b> .
EX-CONCEPT-OWASP	Obligatoire	L'application doit suivre au maximum les bonnes pratiques mentionnées dans les <b>cheatsheets de l'OWASP</b> et les <b>développeurs doivent être sensibilisés/conscients des failles de sécurité majeures dans les applications web</b> et sur les bonnes pratiques pour les éviter.

ID	Niveau	Description
EX- CONCEPT- REVUE-TECH	Obligatoire	<p>Lors de sa phase de conception, <b>le responsable applicatif devra échanger avec les équipes techniques</b> pour vérifier / valider que l'application telle qu'elle sera conçue puisse s'intégrer correctement sur la plateforme d'hébergement.</p> <p>Ceci vise à éviter qu'une application ne soit développée dans une direction qui la rende non-déployable/non-compatible avec l'infrastructure de production.</p> <p>Cette revue technique peut être itérative et doit également survenir dès lors qu'une modification d'architecture profonde est envisagée.</p>
EX- CONCEPT- GESTION- ERREUR	Obligatoire	<p>L'application web doit être conçue pour <b>gérer correctement les erreurs</b>.</p> <p>Ceci inclut notamment de retourner des <b>statuts HTTP cohérents</b> à l'utilisateur. Par exemple, si une erreur inattendue survient coté serveur, elle devra retourner un statut HTTP <code>500</code> (= <code>SERVER INTERNAL ERROR</code>).</p> <p>Si l'utilisateur essaie d'accéder à une ressource dont il ne redispone pas des droits, la réponse peut être <code>403</code> (= <code>FORBIDDEN</code>) pour préciser qu'il n'a pas les accès ou <code>404</code> (= <code>NOT_FOUND</code>) si l'application ne veut pas indiquer qu'il existe une ressource à cette URL.</p> <p>Par ailleurs, en production, l'application ne doit <b>retourner aucune information technique à l'utilisateur final en cas de problème qui soit exploitable par un attaquant</b> (ex : stacktrace, hostname/port du serveur BDD si la connexion a échoué, etc...).</p>

## Runtime

ID	Niveau	Description
EX- RUNTIME- DOCKER	Obligatoire	L'application web doit <b>s'exécuter dans un conteneur Docker</b> dans un environnement Unix.
EX- RUNTIME- USER	Obligatoire	<p>L'application web ne doit <b>pas être exécutée sous l'utilisateur <code>root</code>, ni avec un utilisateur disposant de privilèges supplémentaires</b>.</p> <p>L'utilisation de commandes de type <code>sudo</code> est proscrite dans le conteneur.</p> <p>L'UID (utilisateur), le GID (groupe Unix principal) et éventuellement les groupes secondaires seront précisés au lancement du conteneur.</p>

ID	Niveau	Description
EX-RUNTIME-CONFIG	Obligatoire	<p>Les <b>configurations</b> de l'application qui diffèrent entre les environnements (validation, production) doivent être <b>externalisées</b>.</p> <p>Il est préférable de passer les configurations via des <b>variables d'environnements</b>.</p> <p>Cependant, si les configurations sont extrêmement compliquées, un <b>fichier de configuration externe propre à l'environnement</b> pourra être monté dans le conteneur lors de son lancement en read-only.</p> <p>L'image Docker ou le livrable <b>ne doit pas contenir des informations ou des références spécifiques à un environnement</b>.</p> <p>Ex : les informations de production comme des chemins d'accès ne doivent pas être présentes dans le livrable car elles seraient alors accessibles dans l'environnement de validation.</p>
EX-RUNTIME-EXIT-CODE	Obligatoire	<p>L'application web doit retourner un <b>exit code cohérent</b> qui permette la <b>distinction entre un arrêt normal d'un arrêt suite à une erreur</b>.</p> <p>Par défaut sous Unix, un programme s'arrêtant correctement retourne un exit code 0 et une autre valeur numérique autrement.</p> <p>Si l'application retourne des exit status de succès différents de 0, ceux-ci devront impérativement être précisés pour qu'ils soient correctement gérés.</p> <p>L'exit code sert notamment à relancer automatiquement l'application en cas de crash.</p>
EX-RUNTIME-SHUTDOWN	Obligatoire	<p>L'application web doit <b>s'arrêter correctement et rapidement en cas de réception d'un signal SIGTERM</b> (signal envoyé au process avec le PID 1 dans le conteneur).</p> <p>Ceci survient à minima 1 fois par semaine lors de la mise à jour de la machine hébergeant l'application.</p> <p>Si l'application ne s'arrête pas dans le délai imparti, un signal SIGKILL sera transmis pour forcer son arrêt.</p>
EX-RUNTIME-RES-LIMIT	Obligatoire	<p>Les <b>valeurs maximales en terme de RAM et CPU</b> devront être précisées pour positionner des limites sur le conteneur.</p>

## Logs

ID	Niveau	Description
----	--------	-------------

ID	Niveau	Description
EX-LOGS- PERTINENCE	Obligatoire	<p>L'application web doit générer à minima des <b>logs pertinents</b> qui doivent permettre de <b>comprendre immédiatement la cause d'un problème</b> en production.</p> <p>Ces logs peuvent inclure des informations concernant le contexte d'exécution (ex : identifiant de l'utilisateur courant, valeur d'un paramètre...).</p> <p>Si ces erreurs/comportements anormaux font suite à la levée d'une exception, il est impératif de <b>tracer la stacktrace qui doit inclure l'exception d'origine</b>.</p> <p>Par exemple,</p> <ul style="list-style-type: none"> <li>- en Java : <code>logger.error("Le message en Java", e)</code> plutôt que <code>logger.error("Le message en Java: " + e.getMessage())</code> dans un bloc <code>catch(MyException e) { ... }</code></li> <li>- en Python : <code>logger.exception("Le message en Python")</code> plutôt que <code>logger.error("Le message en Python")</code> dans un bloc <code>except:</code></li> </ul>
EX-LOGS- NIVEAU	Obligatoire	<p>L'application web doit <b>générer des logs en distinguant les niveaux</b> (ERROR, WARN, INFO...).</p> <p>Les développeurs doivent donc utiliser des <b>librairies adéquates</b> (ex : SLF4J/Logback/Log4j2 en Java, le module <code>logging</code> en Python, etc...).</p> <p>L'utilisation de directives comme <code>System.out.println("Le message en Java")</code> ou <code>print("Le message en Python")</code> est donc à proscrire ; il faut plutôt utiliser des loggers et configurer les appenders/handlers pour renvoyer les flux vers STDOUT.</p> <p>En production, le niveau appliqué est généralement <code>INFO</code>, <code>WARN</code> ou <code>ERROR</code>.</p> <p>Il pourra être ponctuellement relevé (ex : passage de certains loggers en <code>DEBUG</code>) si un problème récurrent n'arrive pas à être résolu.</p>
EX-LOGS- DESTINATION	Obligatoire	<p>Les logs doivent être <b>envoyés vers la sortie standard (STDOUT) et/ou la sortie d'erreur (STDERR)</b>.</p> <p>Ils seront rapatriés ailleurs sur le SI par un mécanisme externe à l'application.</p> <p>Cependant, si les logs doivent être séparés, il sera possible de monter un espace disque particulier pour y stocker les fichiers de logs. Ce choix devra être argumenté et validé avec l'équipe infrastructure. ⚠ Les logs principaux devront toujours être envoyés vers la sortie standard. ⚠</p>
EX-LOGS- VOLUMETRIE	Obligatoire	<p>L'application web doit générer une <b>quantité de logs qui soit raisonnable/supportable</b> notamment en production.</p>

ID	Niveau	Description
EX-LOGS-ACCESSLOG	Préconisé	<p>L'application web peut générer des <b>access logs coté backend</b> pour faciliter les investigations.</p> <p>Pour distinguer les access logs des autres messages, ils peuvent être préfixés par <code>ACCESSLOG \</code>.</p> <p>Un exemple d'access log pour une application :</p> <pre>ACCESSLOG \ 11.22.33.44 - [2024-05-23T16:50:14.559+0200] 200 \"GET /path/to/my/resources?param1=value1&amp;param2=value2 HTTP/1.1\" 9ms 1661o \\\"https://myapp.ifremer.fr/path/to/other/page\\\" \\\"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/125.0.0.0 Safari/537.36\\\"</pre> <p>Si l'adresse IP de l'utilisateur est tracée, la <b>liste des proxies de confiance doit pouvoir être précisées en configuration</b> pour traiter correctement le header <code>X-Forwarded-For</code> et <b>éviter problème de sécurité</b>.</p>

## Volumes (fichiers / répertoires persistés)

ID	Niveau	Description
EX-VOLUME	Obligatoire	<p><b>Tout fichier / répertoire devant être persisté</b> entre plusieurs relances du conteneur devra être <b>écrit dans un volume externe</b> au conteneur qui sera monté lors de son démarrage.</p> <p>En règle générale, l'application n'a pas à connaître l'emplacement réel des données dans le reste du SI. Par exemple, le volume contenant les données peut être monté dans le conteneur sous <code>/data</code> mais l'emplacement réel sur le SI différera en validation et en production.</p> <p>Lorsqu'une arborescence de fichiers/répertoires de données est montée dans un conteneur correspondant à une <b>application exposée sur Internet</b>, cette arborescence ne <b>doit contenir que des données publiques</b> (i.e. aucune donnée privée).</p> <p>Idéalement, le <b>responsable du volume et des données qui y sont présentes doit donner son accord</b> pour autoriser le montage de celui-ci dans l'application exposée sur Internet et confirmer qu'il ne contient que des données publiques.</p>

## Réseau

ID	Niveau	Description
----	--------	-------------



ID	Niveau	Description
EX-RESEAU- FLUX-MATRICE	Obligatoire	<p>Tous les flux réseaux entrants et sortants devront être listés exhaustivement, sous la forme d'une <b>matrice de flux complète</b> pour permettre l'ouverture de flux au strict minimum.</p> <p>Ex :</p> <ul style="list-style-type: none"> <li>- Flux entrant - HTTP - TCP/80</li> <li>- Flux sortant - Envoi d'e-mail - SMTP - TCP/25</li> <li>- Flux sortant - BDD PostgreSQL - TCP/5432</li> </ul> <p>En cas de communication vers des URLs externes en HTTP / HTTPS, l'application devra obligatoirement passée par un <b>proxy sortant et être configurée en conséquence</b>. La <b>configuration concernant le proxy devra être passée au lancement de l'application</b>, idéalement via des variables d'environnements (ex : <code>HTTP_PROXY</code> , <code>HTTPS_PROXY</code> , <code>NO_PROXY</code> ...).</p> <p>La liste des domaines de destination devra être précisée au préalable à l'équipe infrastructure (<b>liste blanche de domaines</b>). Seuls les flux HTTP (port 80) et HTTPS (port 443) sur les ports standards sont supportés.</p>
EX-RESEAU- FLUX- AUTORISES	Obligatoire	<p>Seuls certains flux sont autorisés. Une matrice des flux autorisées est disponible dans la <a href="#">documentation</a>.</p>

## Authentification

ID	Niveau	Description
EX- AUTH	Obligatoire	<p>Seuls les protocoles d'authentification et d'autorisation suivants sont actuellement supportés :</p> <ul style="list-style-type: none"> <li>- <b>SAML v2</b></li> <li>- <b>CAS</b></li> <li>- <b>LDAP</b></li> </ul> <p>Le protocole OpenID Connect (OIDC) n'est actuellement pas supporté.</p>

## Monitoring

ID	Niveau	Description
EX- MONITORING- URL	Obligatoire	<p>Le responsable applicatif devra fournir un <b>ensemble d'URLs à tester périodiquement</b> pour valider le bon fonctionnement de l'application.</p> <p>Par exemple, si l'application dépend d'une base de données, l'une des URL doit permettre de valider que l'application accède correctement à la BDD.</p> <p>Le statut de la requête HTTP doit être <code>200</code> en cas de succès et une valeur différente généralement <code>500</code> en cas d'erreur.</p>

## Front

ID	Niveau	Description
EX-FRONT-FQDN	Obligatoire	<p>L'application web sera exposée via un sous-domaine de <code>ifremer.fr</code>.</p> <p>Autrement dit, l'URL pour accéder à l'application débutera par : <code>https://monappli.ifremer.fr</code>.</p> <p>Ceci permet :</p> <ul style="list-style-type: none"> <li>- de faciliter et rationaliser la gestion du DNS, des certificats TLS (achat/renouvellement), des adresses e-mails génériques (ex : <code>abuse@ifremer.fr</code>), etc...</li> <li>- d'éviter qu'un nom de domaine abandonné ne tombe entre de mauvaises mains d'ici quelques années.</li> </ul>
EX-FRONT-ADMIN-URL	Obligatoire	<p>Les <b>URLs d'administration</b> ne doivent pas être exposées sur Internet.</p> <p>Idéalement, le backend doit être accédé via 2 FQDN différents (ex : <code>monappli.ifremer.fr</code> et <code>admin-monapp.ifremer.fr</code>).</p> <p>Le site <code>https://admin-monapp.ifremer.fr</code> permet d'accéder aux interfaces d'administration et n'est accessible que depuis l'intranet Ifremer ou via le VPN.</p> <p>La liste des <b>URLs d'administration</b> devra être fournie à l'équipe infrastructure qui pourra le cas échéant ajouter des protections sur le frontal pour l'exposition de ces URLs à une liste d'adresse IP par exemple.</p>
EX-FRONT-VISIBILITE	Obligatoire	<p>L'application web ne sera pas accessible en "direct" par l'utilisateur final.</p> <p>L'accès à l'application par les utilisateurs s'effectuera au travers une <b>stack de reverse proxies (= front)</b>.</p> <p>L'application web sera exposée <b>par défaut en Intranet</b>.</p> <p>Si celle-ci doit être <b>exposée sur Internet</b>, ce choix devra être <b>justifié</b>.</p> <p>Si une application est constitué de plusieurs conteneurs, mais que certains n'ont pas besoin d'être exposés aux utilisateurs finaux (ex : cas d'un conteneur "frontal / gateway" qui reçoit toutes les requêtes des utilisateurs et les transmet à d'autres conteneurs spécialisés), ceux-ci ne seront pas exposés en dehors de la machine qui héberge la stack applicative et pourra être accédé uniquement par les autres conteneurs constitutifs de l'application s'ils tournent dans le même réseau Docker.</p>

ID	Niveau	Description
<div>EX-FRONT-</div> <div>WAF</div>	Obligatoire	<p>Un <b>pare-feu applicatif (= WAF)</b> géré par l'équipe infrastructure sera positionné en amont de l'application si elle est exposée sur Internet. Des blocages faux-positifs peuvent survenir.</p> <p>Le responsable applicatif devra <b>vérifier exhaustivement le bon fonctionnement de son application en production pour détecter le plus de blocages faux-positifs lors de la mise en exploitation</b> et ainsi permettre aux équipes infrastructures d'ajuster les règles du WAF.</p> <p>Néanmoins, il peut arriver que des blocages faux-positifs surviennent après la mise en exploitation. Le responsable applicatif notifiera l'équipe infrastructure si des utilisateurs lui ont remonté de potentiels blocages pour que les règles du WAF soit ajustées.</p>