



**MINISTÈRE
DE LA TRANSITION ÉCOLOGIQUE,
DE LA BIODIVERSITÉ
ET DES NÉGOCIATIONS
INTERNATIONALES
SUR LE CLIMAT ET LA NATURE**

*Liberté
Égalité
Fraternité*

**CAHIER des
CLAUSES
TECHNIQUES
PARTICULIÈRES**

relatif à l'appel d'offres ouvert pour

**le développement d'évolutions de MASCARET et TELEMAT,
études hydrauliques 1D et 2D, assistance, formations et réunions.**

Le présent document comporte 31 pages numérotées de 1 à 31

Sommaire

1 - Contexte.....	4
1.1 - Présentation générale du réseau Vigicrues.....	4
1.2 - Contexte de l'outil à développer.....	5
1.2.1 - Contexte général de Telemac-Mascaret.....	5
1.2.2 - Contexte de l'utilisation de Mascaret et Telemac par le réseau Vigicrues.....	6
1.3 - Attentes vis-à-vis du prestataire.....	6
2 - Présentation de l'accord cadre.....	8
2.1 - Objet et périmètre de l'accord cadre.....	8
2.2 - Forme de marché.....	8
2.3 - Durée du marché.....	9
2.4 - Annexes.....	9
2.5 - Cadrage technique.....	9
2.5.1 - Référentiels ministériels.....	10
2.5.1.1 - Référentiel général d'amélioration de l'accessibilité – RGAA.....	10
2.5.1.2 - Protection des données personnelles - RGPD.....	10
2.5.1.3 - Cadre de cohérence technique spécifique du ministère.....	11
2.5.1.4 - Cadre de cohérence technique de développement d'applications en python.....	11
2.5.2 - Exigences et bonnes pratiques logicielles.....	12
2.5.2.1 - Qualité de conception.....	12
2.5.2.2 - Qualité d'implémentation.....	12
2.5.2.3 - Intégration continue et environnements.....	12
2.5.2.4 - Suivi des anomalies et dépôts partagés des sources.....	13
2.5.2.5 - Formatage du code.....	15
2.5.2.6 - Linters.....	15
2.5.2.7 - Tests unitaires.....	15
2.5.2.8 - Qualité de code.....	16
2.5.2.9 - Livraison du code.....	16
2.5.2.10 - Qualité de documentation et journaux.....	16
2.5.3 - Hébergement.....	17
2.5.4 - Graphique.....	18
2.5.5 - Administration et supervision informatique.....	18
2.5.6 - Réseaux.....	18
2.5.7 - Exigences de performance.....	19
2.5.8 - Niveaux de service de la solution.....	19
2.5.9 - Migrations.....	19
3 - Prestations.....	20
3.1 - Développement d'évolution (DEV).....	20
3.1.1 - Analyse du besoin (ANA).....	20
3.1.1.1 - Objectifs.....	20
3.1.1.2 - Déroulement.....	20
3.1.1.3 - Livrables.....	21
3.1.2 - Spécifications (SPEC).....	21
3.1.2.1 - Spécifications générales.....	21
3.1.2.2 - Spécifications détaillées.....	21
3.1.3 - Développements (DEV).....	23
3.1.3.1 - Objectifs.....	23
3.1.3.2 - Déroulement.....	23
3.1.3.3 - Livrables.....	23

3.1.4 - Livraisons.....	24
3.1.4.1 - Procédures de livraison.....	24
3.1.4.2 - Documentation.....	25
3.1.5 - Relecture et validation.....	25
3.1.6 - Recette.....	25
3.1.6.1 - Phase de réception et d'installation.....	26
3.1.6.2 - Vérification d'Aptitude (VA).....	26
3.1.6.3 - Vérification de Services Réguliers (VSR).....	26
3.2 - Accompagnement (ACC).....	26
4 - Gestion de projet.....	28
4.1 - Comité de suivi (Cosui).....	28
4.2 - Comité de pilotage élargi.....	29
5 - Clauses environnementales.....	30

1 - Contexte

1.1 - Présentation générale du réseau Vigicrues

Le réseau Vigicrues regroupe le service central Vigicrues (SCV), les services de prévision des crues (SPC), les unités d'hydrométrie (UH) et les cellules de veille hydrologique (CVH) en outre-mer.

Le service central Vigicrues est un service à compétence nationale du Ministère de la Transition écologique (MTE), créé par arrêté interministériel le 2 juin 2003 dans le cadre de la réforme de l'annonce des crues. Il est rattaché au service des risques naturels (SRN) de la direction générale de la prévention des risques (DGPR), et localisé à Toulouse au sein de la Météopole.

Le service central Vigicrues établit et diffuse, en coordination avec les services de prévision des crues (SPC), une information continue de vigilance « crues », publiée sur le site www.vigicrues.gouv.fr. Il collabore avec Météo France pour l'établissement de la vigilance « pluie – inondation ». Il intervient en appui des 17 services de prévision des crues (SPC). À ce titre, il exerce une mission d'organisation, d'animation, d'assistance, de conseil et de formation auprès des services et des établissements intervenant dans le domaine de la prévision de crues, et plus généralement, de l'hydrologie et de l'hydrométrie.

Il coordonne le domaine de la prévision des crues sur les plans technique et scientifique, en liaison avec les organismes scientifiques et techniques de l'État, en pilotant et en impulsant une démarche nationale pour doter le réseau de la prévision des crues de modèles et plus généralement d'outils ou de méthodes fournissant des prévisions hydrologiques en temps réel.

Par ailleurs, il collecte à un niveau national et en temps réel les données hydrométéorologiques des différents producteurs de données. Il gère et fait évoluer la banque de données hydrométriques. Il en assure la conservation, l'expertise et la mise à disposition de manière à permettre leur utilisation dans de multiples situations, en particulier en situation de crues ou d'étiage. Il appuie et coordonne les 19 unités d'hydrométrie (UH) dans les DREAL et des 5 cellules de veille hydrologique (CVH).

Dans les deux domaines, prévision des crues et hydrométrie, le service central Vigicrues traite de l'analyse des phénomènes, du développement des méthodes et des outils, de la synthèse et de l'appui aux SPC, UH et CVH. En cas de difficulté particulière, il intervient également en matière de programmation des ressources humaines et financières sur l'ensemble des activités d'hydrométrie et de prévision de crues.

Pour répondre aux enjeux, le réseau Vigicrues s'est doté d'un projet stratégique pluriannuel dont la prochaine échéance est celle de 2025-2029. Le projet phare de cette stratégie est la couverture totale du territoire par la vigilance crues, dénommée Vigicrues 2030.

Le service central Vigicrues pilote l'ensemble de l'activité du réseau Vigicrues et pour ce faire, il développe et déploie des outils informatiques nationaux utiles à l'ensemble du réseau Vigicrues. Le service central Vigicrues doit associer dans la conduite de ses développements des outils les usagers finaux que sont, dans la plupart des cas, les SPC, UH et CVH, mais également le grand public (outils Vigicrues et Banque Hydro en particulier), les bureaux d'études et les collectivités territoriales.

1.2 - Contexte de l'outil à développer

1.2.1 - Contexte général de Telemac-Mascaret

Telemac et Mascaret sont des codes hydrodynamiques, respectivement à 2 dimensions et à 1 dimension, à surface libre, qui résolvent les équations de Barré de Saint-Venant.

Ils font partie d'une suite de logiciels « Open Telemac » libres et gratuits et sont développés dans le cadre du consortium « Open Telemac » qui regroupe différentes entreprises (EDF-LNHE, ARTELIA, HR Wallingford, Daresbury Laboratory, CEREMA, CERFACS, BAW...).

Le portail de cette suite <http://www.opentelemac.org/> permet d'échanger avec la communauté des utilisateurs et des développeurs qui effectuent le maintien en condition opérationnelle de ces codes.

Une conférence des utilisateurs Telemac-Mascaret se réunit chaque année comme en 2025 à Menai Bridge (Pays de Galles). cf. <https://gitlab.pam-reted.fr/otm/telemac-mascaret/-/wikis/TELEMAC-User-Conference-2025>.

Telemac 2D

Telemac 2D (à deux dimensions horizontales) est développé par le Laboratoire National d'Hydraulique et Environnement (LNHE) d'EDF. Il repose intégralement sur un maillage non structuré, de type éléments finis. Il peut être utilisé pour représenter les inondations dues aux crues des cours d'eau mais également pour des applications maritimes ou pour des plans d'eau.

Plusieurs interfaces utilisateurs existent : plugins QGIS¹, BlueKenue², Salome³... ou sont en cours de préparation : Qtelemac (open source, Outil standalone Python – Qt pour Windows – Linux, 1ère version prévue pour 2026)

Mascaret

Historiquement Mascaret est issu d'une collaboration entre EDF et le CETMEF. Mascaret fait désormais partie de la suite de codes libres et gratuits (cf. <https://en.wikipedia.org/wiki/TELEMAC> et <http://opentelemac-mascaret-consortium.software.informer.com/>) qui sont utilisés dans de nombreuses études à travers le monde.

Concernant les pré et post traitements de Mascaret, l'utilisation de l'interface historique Fudaa est remplacée par le plugin QGIS Mascaret maintenu en conditions opérationnelles par ARTELIA dans le cadre du consortium Telemac-Mascaret. cf. <https://github.com/Artelia/Mascaret>. Le SCV a financé un certain nombre d'évolutions (ouvrages mobiles, amélioration générale des interfaces du plugin...) dans le cadre d'un accord cadre précédent de 2022 à 2026.

La documentation utilisateurs, essentiellement en langue française pour l'instant, est accessible sous l'onglet [Wiki](#).

Ce plugin QGIS (disponible sur le dépôt officiel QGIS) permet la construction d'un modèle Mascaret en s'appuyant sur les fonctionnalités SIG. Son fonctionnement repose sur une base Postgres. Il permet le lancement de Mascaret, avec le noyau de calcul sélectionné, sur différents épisodes simultanément.

Des synergies de développements sont intéressantes à trouver avec les autres outils en cours de développements par les membres du Consortium : Plugin QGIS PreCourlis (open source),

1 cf. <https://plugins.qgis.org/plugins/q4ts/> , <https://plugins.qgis.org/plugins/PostTelemac/> , https://plugins.qgis.org/plugins/mesh_tools/ , etc.

2 cf. <https://nrc.canada.ca/en/research-development/products-services/software-applications/blue-kenuetm-software-tool-hydraulic-modellers>

3 cf. <https://www.edf.fr/groupe-edf/inventer-l-avenir-de-l-energie/rd-un-savoir-faire-mondial/nos-offres/nos-logiciels-et-codes-de-calcul/salome>

QMasCourlis (open source, Outil standalone Python – Qt (Windows – Linux) disponible sur demande, distribution en 2026).

1.2.2 - Contexte de l'utilisation de Mascaret et Telemac par le réseau Vigicrues

La modélisation pour la **prévision des crues** est réalisée par les SPC. Le service central Vigicrues préconise l'utilisation d'un nombre réduit de plateformes et de modèles dits modèles « nationaux », c'est-à-dire maintenus et bénéficiant d'améliorations définies par le réseau Vigicrues, en fonction des besoins exprimés. Ces outils (GRP, Plathynes, Mascaret et Telemac) reposent sur des services supports (publics et/ou privés) français pour des questions d'autonomie stratégique et de souveraineté. Ils sont maintenus compatibles avec les outils nationaux, en particulier la Plateforme Opérationnelle pour la Modélisation (POM), les outils permettant d'afficher des prévisions avec des incertitudes, OTAMIN et le Superviseur. La capacité du réseau Vigicrues à continuer à bénéficier d'outils souverains (c'est-à-dire français) reste un objectif dans le cadre du présent accord cadre.

D'autre part, les modèles Mascaret et Telemac sont utilisés afin d'obtenir des **cartographies des inondations**. Le réseau Vigicrues vise à compléter la prévision des crues par la prévision des inondations en associant à une prévision ponctuelle (hauteur ou débit à une station hydrométrique de référence), une emprise inondée. Cette opération doit permettre de répondre au besoin des gestionnaires de crise de traduire les informations ponctuelles diffusées sur Vigicrues (<https://www.vigicrues.gouv.fr>) en conséquences à l'échelle d'un territoire. Les scénarios produits sont caractérisés par leur rattachement à une hauteur d'eau à la station hydrométrique la plus représentative dans le référentiel national.

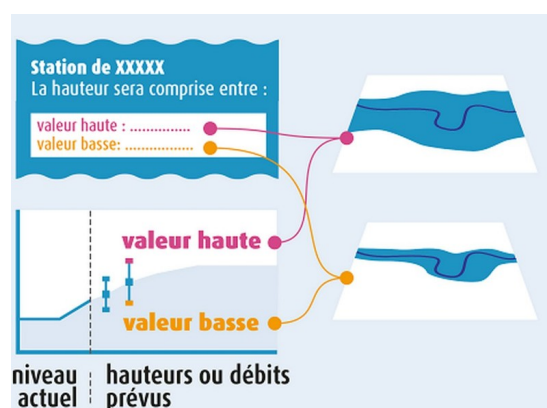


Figure 1: Schéma de principe de la prévision des inondations

Enfin ces modèles sont également utilisés dans le cadre de l'hydrométrie, en particulier pour calculer des courbes de tarage permettant de relier les hauteurs aux débits au niveau d'une section de cours d'eau.

Pour ces différents usages et pour faciliter l'ergonomie de travail du modélisateur, le SCV finance certaines évolutions des modèles Mascaret et Telemac et de leurs interfaces en particulier le plugin QGis Mascaret.

D'autre part le réseau Vigicrues sous-traite complètement ou partiellement la réalisation d'études de modélisation comme citées ci-dessus.

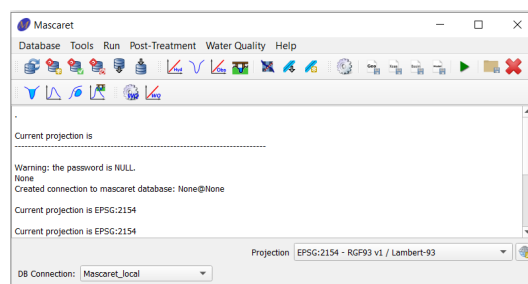


Figure 2: Interface du plugin QGis Mascaret

Dans ce marché, le service central Vigicrues est le service coordonnateur, maître d'ouvrage.

Les DREAL et DEAL sont les services coordonnés.

1.3 - Attentes vis-à-vis du prestataire

Il est attendu du prestataire qu'il ait :

- une connaissance fine des territoires des bassins versants français pour répondre de manière

pertinente au volet modélisation

- une maîtrise de outils de modélisation hydraulique, et en particulier de Mascaret et Telemac
- la capacité de concevoir des modèles Mascaret et/ou Telemac aussi bien pour la prévision des crues opérationnelle que pour la production de cartographie des inondations
- capacité de montée en puissance en cas de commandes simultanées tant sur les volets développements qu'études

2 - Présentation de l'accord cadre

2.1 - Objet et périmètre de l'accord cadre

Le présent accord-cadre a pour objet principal la réalisation des prestations suivantes :

- Analyses, spécifications et développement d'évolutions de Mascaret et Telemac 2D (codes et interfaces) : elles consistent à faire évoluer une application pour modifier son comportement ou pour proposer de nouvelles fonctionnalités.
- Etudes de modélisation hydraulique grâce à des outils du monde libre comme Mascaret et Telemac afin de construire des modèles pour la prévision des crues et/ou la cartographie des inondations et production de cartographies
- Assistances, formations ou réunions sur Mascaret et Telemac

grâce à des prestations d'analyse et développement d'évolutions, d'étude, assistance, formation et réunion.

Dans la suite du document le terme « maître d'ouvrage » cible le Service central Vigicrues (SCV) ou les Services de prévisions des crues (SPC). En effet, les différents bons de commande pourront être effectués par le SCV ou les SPC.

2.2 - Forme de marché

Le présent marché est organisé de la façon suivante :

Prestations		Déclenchement	Prix
Maintenance évolutive (ME)	Analyse (ANA)	bon de commande	unitaire
	Spécifications (SPEC)	bon de commande	unitaire
	Développement (DEV)	bon de commande	unitaire
Études (ETU)		bon de commande	unitaire
Accompagnement (ACC)		bon de commande	unitaire

Il est possible de faire un Bon de commande (BC) qui couvre plusieurs types de prestation.

2.3 - Durée du marché

La durée de l'accord cadre est de 24 mois reconductibles 1 fois.

2.4 - Annexes

Voici la liste des annexes au CCTP :

1. Spécifications fonctionnelles détaillées du PIM et du PIT.
2. Évolutions envisagées

2.5 - Cadrage technique

Les développements effectués dans le cadre de cet accord cadre devront respecter les exigences techniques listées dans les différents documents applicables suivants dans leurs versions en vigueur au moment de l'exécution de la prestation. Dans le cas contraire, les choix techniques différents devront être impérativement validés au préalable par le maître d'ouvrage. Le titulaire devra se conformer à ces documents de référence et pourra proposer des adaptations de l'existant pour améliorer la conformité globale du système.

Pour faciliter la mise en place d'un numérique responsable, le Gouvernement a mis en place des référentiels et les règlements suivant que l'on cherchera à respecter. En lien avec ces référentiels, les documents de référence du SI de l'État (référentiels interministériels) en ligne sont indiqués.

- Référentiel Général d'Interopérabilité (RGI) : <https://www.marche-public.fr/Marches-publics/Definitions/Entrees/Dematerialisation/Referentiel-general-interoperabilite.htm>
Pour normer les standards techniques des administrations techniques et optimiser la communication entre eux et avec les usagers.
- Référentiel Général d'Accessibilité des Applications (RGAA) : <https://accessibilite.numerique.gouv.fr>
Ensembles de normes visant à rendre les services numériques accessibles à tous, notamment aux personnes en situation d'handicap. Plus de détail dans le chapitre 2.5.1.1.
- Référentiel Général de Sécurité (RGS) : <https://cyber.gouv.fr/le-referentiel-general-de-securite-rgs>
Pour garantir la confidentialité, intégrité, disponibilités et traçabilité des données en France.
- Référentiel Général d'Écoconception de Services Numériques (RGESN) : <https://ecoresponsable.numerique.gouv.fr/publications/referentiel-general-ecoconception>
<https://institutnr.org/referentiel-general-ecoconception-de-services-numeriques>
Guider la conception de service numérique afin de minimiser son impact sur l'environnement tout au long de son cycle de vie.
- Référentiel Général sur la protection des données (RGPD) : <https://www.cnil.fr/fr/reglement-europeen-protection-donnees>
Pour cadrer la protection des données personnelles et harmoniser les pratiques de gestion des données au sein de l'U.E. Plus de détail dans le chapitre 2.5.1.2.
- Référentiel Général de Gestions des Archives (R2GA) : https://francearchives.gouv.fr/fr/circulaire/R2GA_2013_10
Pour normer, organiser et conserver des archives conformément aux obligations légales et institutionnelles.

- Charte Internet de l'État : <https://www.info.gouv.fr/marque-de-letat/charte%20graphique-introduction>

Les solutions préconisées s'appuient sur les préconisations SILL (<https://www.data.gouv.fr/fr/datasets/socle-interministeriel-de-logiciels-libres>) et les standards OGC (<https://www.ogc.org>).

Les sous-chapitres ci-dessous résument ou complètent les indications des référentiels.

2.5.1 - Référentiels ministériels

2.5.1.1 - Référentiel général d'amélioration de l'accessibilité - RGAA

L'accessibilité numérique consiste à rendre les services en ligne accessibles aux personnes en situation de handicap. La direction interministérielle du numérique (DINUM) édite le référentiel général d'amélioration de l'accessibilité (RGAA, anciennement référentiel général d'accessibilité pour les administrations), aujourd'hui en version 4.1 :

<https://www.numerique.gouv.fr/publications/rgaa-accessibilite/>

Les services publics numériques **ont l'obligation d'être accessibles de façon équivalente à tout citoyen**, qu'il soit ou non en situation de handicap (visuel, auditif, moteur, trouble dys...).

Dans le cas d'un outil interne, il reste important de prendre en compte les questions d'accessibilité en compte sur tous les nouveaux développements pour pouvoir être utilisées par tous les agents du réseau Vigicrues. Des propositions pourront également être formulées par le prestataire pour améliorer l'accessibilité des interfaces existantes.

2.5.1.2 - Protection des données personnelles - RGPD

Chacune des parties veillera à ce que son personnel, son groupe et ses éventuels sous-traitants respectent les engagements prévus dans le présent article.

Le service central Vigicrues demeure « responsable de traitement », au sens de la réglementation applicable, pour les données à caractère personnel qu'il fournirait titulaire du marché, dans le cadre de l'exécution des présentes.

Le titulaire du marché est un « sous-traitant », au sens de la réglementation applicable et agit uniquement pour le compte et sur instruction du service central Vigicrues, sur la base des conditions du présent contrat.

Obligations du titulaire du marché :

Dès lors que le titulaire du marché, dans le cadre de l'exécution du présent contrat, est amenée à traiter des données à caractère personnel (exemple : sauvegardes, journaux de sécurité, ...) confiées par le service central Vigicrues, le titulaire du marché s'engage à :

- prendre toutes mesures pour garder les dites données confidentielles ;
- prendre des mesures de sécurité organisationnelles, physiques et techniques appropriées afin de protéger lesdites données ;
- informer le service central Vigicrues immédiatement après détection d'une fuite de données le concernant ;
- n'effectuer aucun transfert desdites données personnelles en dehors du territoire de l'Union européenne, sauf avec l'accord préalable du service central Vigicrues, et dans un cadre conforme à la réglementation applicable, c'est à dire soit vers des pays présentant un niveau

de protection dit « adéquat » au sens des autorités européennes de protection des données, soit vers des entités ayant signé des clauses contractuelles types telles qu'édictées par les autorités européennes.

2.5.1.3 - Cadre de cohérence technique spécifique du ministère

Dans le cadre des nouveaux développements le titulaire devra prendre en considération les dernières recommandations en vigueur :

- Prise en compte du responsive web design.
- Compatibilité avec des navigateurs et versions cibles habituelles interministérielles.

Le titulaire devra respecter a minima les règles PSR (PHP Standards Recommendation) pour les nouveaux développements.

2.5.1.4 - Cadre de cohérence technique de développement d'applications en python

Documents de référence

- not_cadrage-technique-python.pdf (https://intra.schapi.e2.rie.gouv.fr/cadrage-technique-python-a3398.html?id_rub=533)

NB : Une revue sera réalisée durant la phase d'initialisation pour valider le cadre technique à date.

Généralités

Langage Python >=3.11 (version recommandée 3.13)* avec utilisation systématique des « type hints » et le respect des « bonnes pratiques » décrites notamment par les PEP 8, 257, 484 et 526 (voir <https://peps.python.org>).

Utilisation de « Semantic Versioning » (<https://semver.org>) et d'un CHANGELOG (<https://keepachangelog.com>) par produit.

** la version de Python packagée avec Debian 12 Bookworm est la 3.11 et celle avec Debian 13 Trixie est la 3.13.*

Dépendances

- **Gestion des dépendances**

Avec pip et des fichiers de « requirements » dont au moins un pour le projet lui-même et un autre pour installer les paquets supplémentaires nécessaires au développement (linters et formateurs).

Les dépendances non disponibles sur internet (PyPI · The Python Package Index) seront mises à disposition sur la forge (registre de paquets du projet GitLab) afin de pouvoir être utilisés en intégration continue. Elles ne feront pas partie du dépôt de code.

- **Dépendances non standards**

Il s'agit ici des dépendances qui ne sont pas disponibles avec une installation de base de Python.

Liste de dépendances externes recommandées et autorisées :

- toml pour les fichiers de configurations
- numpy et pandas pour les tableaux et séries de données temporelles
- pytz, python-dateutil et arrow pour une gestion avancée du temps (dans un ordre croissant de fonctionnalités nécessaires)
- requests pour les requêtes HTTP
- psycopg3 pour un connecteur PostgreSQL
- lxml
- pyproj, rasterio (gdal), fiona, shapely pour la cartographie
- protobuf, messagepack pour la sérialisation si cpickle ne suffit pas
- PyQt (5 ou 6) et matplotlib pour les interfaces graphiques
- sphinx pour la documentation avancée
- les bibliothèques métier fournies par le service central Vigicrues : libhydro, libbdimage...

Les autres dépendances sont à approuver au cas par cas par le maître d'ouvrage après consultation de l'architecte technique Python.

2.5.2 - Exigences et bonnes pratiques logicielles

2.5.2.1 - Qualité de conception

- Les choix architecturaux et techniques devront être en cohérence avec les différents documents de cadrage technique fourni et devront être validés par le maître d'ouvrage.
- Le prestataire devra respecter les bonnes pratiques de conception avec le Framework Symfony : https://symfony.com/doc/6.4/best_practices.html, notamment les règles ou préconisations d'organisation du code, de nommages des « concepts de programmation » d'écriture. Il devra aussi respecter les bonnes pratiques de tout autre élément intégrée au projet ;
- En respectant la règle précédente, les règles de nommage seront cohérentes sur l'ensemble du code source, quel que soit le langage utilisé ;

2.5.2.2 - Qualité d'implémentation

- Les noms des « concepts de programmation » doivent être explicites ;
- Il est rigoureusement interdit d'utiliser espaces, caractères spéciaux, ou mots réservés pour nommer les « concepts de programmation » ;
- Le code source doit être indenté et espacé afin de mettre en valeur l'algorithme et, le cas échéant, de respecter les normes du langage de programmation ;
- Le code pourra être soumis à un audit de code pour valider le respect des standards d'implémentation.

2.5.2.3 - Intégration continue et environnements

Le titulaire devra mettre en place une plateforme d'intégration continue comprenant au minimum :

- Une gestion du contrôle de version basée de préférence sur Git
- Une chaîne d'intégration continue en charge à minima :
 - Des test unitaires
 - Des tests de qualité de code et de sécurité

Le résultat de ces tests fera partie intégrante des livrables de version.

L'environnement de développement devra être défini en concertation avec le maître d'ouvrage ; il sera fait usage de manière prioritaire d'outils issus du monde libre ; tout usage d'un logiciel dit « propriétaire », en particulier lorsque la licence est payante, se fera sous accord expresse du maître d'ouvrage ; en ce cas, le coût de la licence sera à la charge du maître d'ouvrage ;

L'environnement de production est nécessairement distinct de celui de développement ; il sera défini conjointement par le prestataire et le maître d'ouvrage en faisant usage de manière prioritaire d'outils issus du monde libre ; tout usage d'un logiciel dit « propriétaire », en particulier lorsque la licence est payante, se fera sous accord expresse du maître d'ouvrage ; en ce cas, le coût de la licence sera à la charge du maître d'ouvrage.

2.5.2.4 - Suivi des anomalies et dépôts partagés des sources

Le service central Vigicrues utilise l'outil de gestion de versions Git pour ses différents projets. Les codes sources des applications et des tests unitaires seront à déposer sur le dépôt qui sera précisé pendant la phase d'initialisation (cf. 3.1).

L'utilisation d'un outil de suivi des anomalies (bug tracker) est demandée tout au long de la vie du présent accord-cadre.

Le choix des outils de suivi des anomalies et de suivi du projet sera réalisé conjointement et pourra être celui utilisé par le service central Vigicrues (Gitlab).

Le prestataire peut proposer un outil comparable et mettre en avant ses avantages et la possibilité de mettre à disposition son contenu à la fin de la prestation.

Un suivi de la correspondance entre les versions applicatives et les tickets résolus doit être assuré à minima en notes de version.

Git-flow sera utilisé comme workflow de branchement Git.2

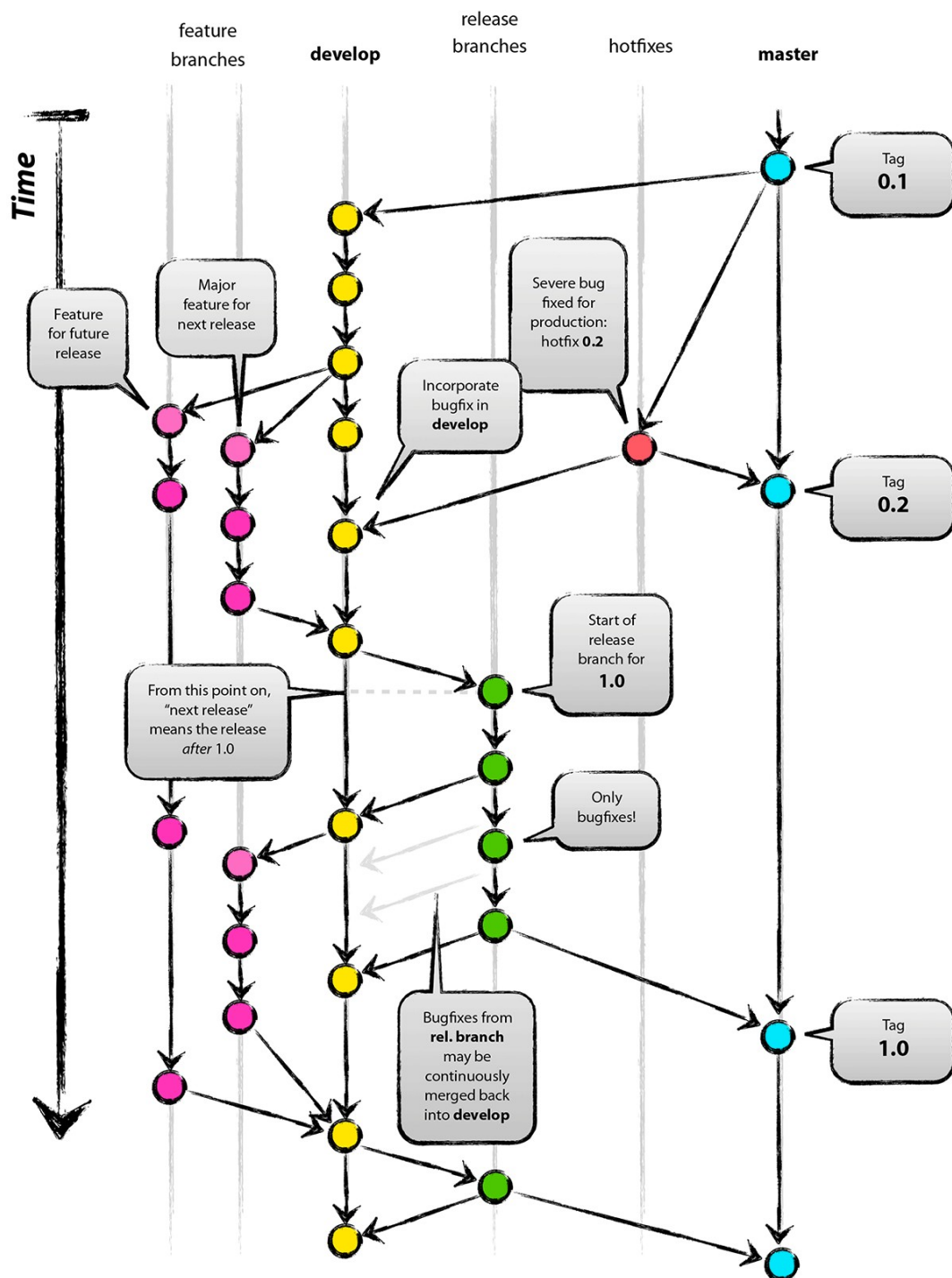
Dans le schéma donné en exemple ci-dessous on distingue les branches principales, fixes et immuables :

- master est la branche où tout est stable. Chaque commit correspond à une version stable du projet (release) qui peut être déployée en production et taguée en conséquence (vX.Y.Z).
- develop est la branche sur laquelle s'effectue le développement proprement dit. On y prépare les changements en vue de la prochaine release dans master.

Puis, si besoin (à discuter avec le service central Vigicrues), les branches secondaires qui se font et se défont avec le temps :

- feature part de develop et se merge dans develop. On crée une branche feature/xxx lorsque l'on travaille sur une fonctionnalité en particulier. Lorsqu'elle est terminée, on la merge dans develop pour ajouter la feature stable dans le scope de la prochaine release.
- release part de develop et se merge dans master et develop. On crée une branche release/xxx à partir de develop lorsque celle-ci reflète l'état désiré de la release (l'ensemble des fonctionnalités du scope ont été mergées). Ainsi, on peut préparer la prochaine release tranquillement, corriger d'éventuels bugs et poursuivre le développement en parallèle. Une fois que la release est prête (stable) on merge alors la branche dans master, mais aussi dans develop pour mettre à jour les modifications apportées.

- hotfix part de master et se merge dans master et develop / release. On crée une branche hotfix/xxx lorsque l'on veut résoudre un bug critique en production rapidement. C'est un peu comme une release non planifiée.
- Lorsque le correctif est développé, on le merge dans master avec le numéro de version qui convient, ainsi que dans develop (ou la branche release en cours, le cas échéant) pour mettre à jour les modifications apportées.



2.5.2.5 - Formatage du code

L'utilisation des formateurs de code est obligatoire afin de faciliter la lisibilité du code fourni.

Dans le cadre de développement Python, l'utilisation d'un formateur de code est obligatoire pour tous les fichiers Python, ceux de l'application elle-même et ceux des tests. On utilisera l'outil « ruff » avec une longueur de ligne de 110 caractères.

Un makefile ou mieux un « just file » (<https://github.com/casey/just>) permettant d'automatiser d'une part le contrôle et d'autre part le formatage devra être intégré au projet pour pouvoir exécuter ces tâches rapidement en local ou bien dans une forge.

La configuration devra être portée intégralement dans un fichier « pyproject.toml ». Un exemple de configuration et de makefile est fourni en annexe.

Ni les fichiers de l'application ni le dépôt de code ne doivent contenir de directives propres à un IDE ou éditeur. Des recommandations pour l'utilisation d'un éditeur particulier, ou bien des configurations peuvent si nécessaire être mises à disposition sur le Wiki de la forge.

2.5.2.6 - Linters

Un linter est un programme qui analyse un code source informatique en vue de détecter des erreurs de programmation ou de vérifier qu'il respecte un ensemble de normes d'écriture.

Les linters utilisés devront obligatoirement permettre l'ensemble des vérifications de qualité et de sécurité de code spécifiés.

Dans le cadre de développement Python, on utilisera pour tous les fichiers Python du projet et des tests :

- **ruff** (<https://github.com/astral-sh/ruff>), un linter rapide qui remplace avantageusement pyflakes, pycodestyle, mccabe, isort, pydocstyle, flake8, eradicate et pylint. Un exemple de configuration est fourni en annexe
- **vulture** (<https://pypi.org/project/vulture/>, find dead code) avec « min-confidence=80 »

On utilisera également « **mypy** » pour valider les types-hints du programme, les tests unitaires pouvant être exclus de ce contrôle. Le mode « strict » sera utilisé pour les nouveaux projets, les exceptions à justifier pour les projets incluant de la dette technique.

Un makefile ou mieux un « just file » (<https://github.com/casey/just>) permettant d'automatiser ces contrôles devra être intégré au projet pour pouvoir exécuter ces tâches rapidement en local ou bien dans une forge.

La configuration devra être portée intégralement dans un fichier « pyproject.toml ». Un exemple de configuration et de makefile est fourni en annexe.

NB : il est bien sûr essentiel de configurer son IDE pour vérifier en continu et corriger plus aisément les non-conformités. L'utilisation d'un hook GIT est également possible pour bloquer les commits contenant du code invalide (par exemple avec pre-commit : <https://pre-commit.com>).

2.5.2.7 - Tests unitaires

Les tests sont regroupés dans un sous-répertoire « test » du dépôt de code, lequel contient si besoin un répertoire « data » lorsque des données nécessaires aux tests sont sérialisées.

Couverture de test cible :

- 80 % pour les librairies ou modules
- 70% pour les programmes

Dans le cas de produit déjà développé, une première analyse sera réalisée afin de déterminer la cible de couverture des tests unitaires.

Le résultat de ces tests fera partie intégrante des livrables de version.

Les tests ne doivent pas être intégrés dans les packages d'installation, mais ils peuvent faire l'objet d'une archive accompagnant une release.

2.5.2.8 - Qualité de code

La qualité du code devra faire l'objet d'une analyse statique à chaque livraison pour prendre en compte à minima :

- La complexité cyclomatique
- La couverture de code
- La duplication de code
- La sécurité OWASP & CWE Top 25

Le rapport de ces analyses devra être fourni lors des livraisons.

La cible des indicateurs d'analyse sera définie conjointement dans la phase d'initialisation

2.5.2.9 - Livraison du code

Sur le dépôt GitLab du service central Vigicrues, sous la forme d'un dépôt GIT et sauf exception en une seule branche principale (main).

Les commits des releases doivent être tagués dans GIT avec le numéro de la future version :

```
git tag -a -m 'Release 1.0.2' 'v1.0.2' 8e227e0
```

Des releases GitLab doivent être créées pour chaque livraison et proposer au téléchargement le ou les paquets d'installation et la documentation qui est gérée avec le code (voir paragraphe dédié). Dans le cas d'une bibliothèque ou d'un logiciel pouvant être utilisé dans plusieurs configurations (Windows, Linux, dépendances non figées...) il sera utile d'ajouter à la release les jeux de tests sous forme d'une archive.

2.5.2.10 - Qualité de documentation et journaux

Documentation :

- A minima 100% du code spécifique au service central Vigicrues devra être documenté.
- Chaque modification du code source devra être documentée dans l'historique de modification du gestionnaire de version ainsi il sera possible d'en identifier la nature, la date, l'auteur et la raison. Chaque « commit » nécessitera un commentaire obligatoire.
- Le code source fera l'objet, a minima, d'une documentation à l'intérieur même du code source ; cette documentation ne doit pas systématiquement paraphraser le code, mais plutôt expliquer les raisons qui ont conduit le développeur à choisir telle ou telle implémentation ;

- Les « concepts de programmation » doivent être systématiquement documentés ; en particulier, et sans que la liste ci-dessous ne soit exhaustive :
 - Les classes et tables de base de données devront décrire le concept qu'elles portent ;
 - Les fonctions et méthodes devront décrire leur fonctionnement global, le(s) donnée(s) nécessaire(s) en entrée et le(s) donnée(s) renvoyée(s) (voir ci-dessous concernant les données) ;
 - Les procédures devront décrire leur fonctionnement global et le(s) donnée(s) nécessaire(s) en entrée (voir ci-dessous concernant les données) ;
 - Les variables et champs de base de données devront décrire leur nom, leur type, leur cardinalité et, le cas échéant, leurs valeurs possibles.

Journaux (le cas échéant) :

- Les journaux doivent être formatés de façon à faire apparaître, a minima :
 - Le type d'événement (information, avertissement, erreur, etc.) ;
 - L'horodatage de l'événement ;
 - L'origine de l'événement ;
 - Le cas échéant, les paramètres et/ou conditions ayant provoqué le déclenchement de l'événement ;
 - L'événement lui-même.
- Les journaux devront tracer les appels aux référentiels externes (accès Web Service)
- Le produit final devra pouvoir être configuré afin de fonctionner au minimum selon 4 modes différents : un mode de débogage (notamment utilisé durant les phases de tests), un mode dit verbeux (notamment utilisé durant les phases de pré-production), un mode normal (utilisé entre autre en production) et un mode silencieux (facultatif).

2.5.3 - Hébergement

Le produit développé pourra être hébergé sur l'un des principaux types d'hébergement disponibles :

- par le service central Vigicrues, dans le cadre de l'offre d'« hébergement centralisé »
- par un hébergeur ministériel ATE pour les applications PHP
- par certaines DREAL

Actuellement le service central Vigicrues et les différentes DREAL utilisent un environnement virtualisé de type VMware en haute disponibilité.

D'autres DREAL utilisent un environnement virtualisé de type Proxmox en haute disponibilité.

Une machine virtuelle Telemac-Mascaret est régulièrement mise à jour par le maître d'ouvrage et permet l'utilisation dans différents cadre :

- pilotage des modèles de prévision en temps réel par la POM (cf. 1.2.2)
- études (évaluation des modèles, cartographie des inondations...)

Une assistance à la mise à jour de cette machine virtuelle pourra être commandée.

2.5.4 - Graphique

Le service central Vigicrues devra disposer de tous les fichiers sources des écrans validés, dans leur format natif. Les calques et formes vectorielles seront préservés dans les fichiers livrés, y compris les fichiers encapsulés (objets dynamiques).

2.5.5 - Administration et supervision informatique

Ce chapitre ne concerne a priori pas les prestations demandées dans le présent accord cadre mais est donné pour indication du contexte plus global du SI du SCV.

Les développements devront viser à proposer des interfaces d'administration et de supervision informatique à l'état de l'art.

La supervision technique utilisée par le réseau Vigicrues et opérée par le « Pôle de Supervision Informatique National » (PSIN) et par le service central Vigicrues. Ces supervisions sont basées sur le logiciel Centreon. La solution devra s'intégrer dans cet environnement en proposant des sondes systèmes et applicatives permettant la supervision et l'alerte en cas de défaillance logicielle.

Les sondes devront permettre la détection :

- d'incidents liés à l'environnement système (manque de ressources mémoire ou cpu par exemple)
- d'incidents applicatifs en proposant des vérifications de bonne santé applicative à différents niveaux (authentification, page de consultation cohérente, etc)

Pour l'ensemble de ces vérifications des modes opératoires seront rédigés pour corriger les erreurs détectées.

A titre d'information, nous rappelons ci-dessous quelques cas d'anomalies auxquels le service central Vigicrues a été confronté avec la solution existante :

- Microcoupure ou coupure du réseau RIE ou du réseau local
- Indisponibilité du service d'authentification SSO

Le service central Vigicrues sera particulièrement sensible aux mesures prises pour prévenir ou traiter ces anomalies :

- Mécanisme d'alerte
- Mécanisme de trace
- Mécanisme de ré-exécution (manuel ou automatique)
- Mode hors-ligne ou stockage local temporaire

2.5.6 - Réseaux

Ce chapitre ne concerne a priori pas les prestations demandées dans le présent accord cadre mais est donné pour indication du contexte plus global du Si du service central Vigicrues

La production repose sur les infrastructures réseaux locales du service central Vigicrues/DREAL/SPC, les infrastructures ministérielles (Cerbère, authentification SSO cf note technique d'utilisation du SSO ministériel <https://numerique.metier.e2.rie.gouv.fr/produit-securite-cerbere-authentification-r192.html> et https://intra.schapi.e2.rie.gouv.fr/cadrage-technique-de-cerberisation-cas-d-une-a3449.html?id_rub=533) et le RIE. Pour les usages en astreinte, un réseau « données mobiles » peut être utilisé par les prévisionnistes pour se connecter au RIE. Les règles de

sécurisation des accès ainsi que les mécanismes existants (VPN...) devront pouvoir être utilisés par la solution.

2.5.7 - Exigences de performance

Le temps maximal d'affichage d'une page ne devra pas être supérieur à 3 secondes dans 80% des cas (impératif et idéalement inférieur pour la page d'accueil dont le poids devra être optimisé).

Ce temps de chargement s'entend depuis la requête de l'utilisateur (Chargement d'une page dans le navigateur, clic sur un lien, validation d'une action, navigation dans un contenu...) jusqu'à la mise à disposition complète du résultat.

Si le produit nécessite des métriques relatives aux fonctions développées, les tests unitaires devront prendre en charge des aspects de temps de traitement avec une restitution à chaque livraison dans le cadre des tests d'intégration continue.

2.5.8 - Niveaux de service de la solution

L'architecture et la solution d'hébergement mise en œuvre doit être la plus à même de répondre aux niveaux de service importants nécessaires à l'activité opérationnelle de prévision des crues.

2.5.9 - Migrations

Chaque livraison doit prévoir un script automatisé d'installation et de migration depuis la version précédente. Une procédure de retour arrière doit être décrite.

3 - Prestations

Les frais de maintenance des environnements de développement et de démonstration et les outils (coûts des licences notamment) nécessaires à la réalisation des prestations sont à la charge du prestataire.

3.1 - Développement d'évolution (DEV)

Les étapes de développement et de maintenance évolutive détaillées dans les sous chapitres suivants s'effectuent dans l'ordre suivant :

- Analyse du besoin (qui peut éventuellement être sautée si elle est triviale) : possible sous forme d'ateliers
- Spécifications générales
- Spécifications fonctionnelles détaillées, avec relecture et validation
- Développements
 - livraison avec documentation (cf. 3.1.4.2)
 - installation/mise à jour et recette (VA puis VSR)
 - garantie d'un an puis MCO

Pour les prestations de développement d'évolutions, afin d'éviter l'effet « tunnel », il est demandé au prestataire de proposer un découpage adapté à la commande sous forme de « sprints » (cycles de spécifications/développements).

3.1.1 - Analyse du besoin (ANA)

3.1.1.1 - Objectifs

La prestation d'analyse du besoin permet de partir des besoins fonctionnels de utilisateur et de proposer plusieurs solutions techniques pour couvrir ces besoins avec une description des avantages et des inconvénients de ces solutions (complexité technique/coût de développement, contraintes fonctionnelles, ergonomie...) afin que le maître d'ouvrage puisse retenir une solution technique à spécifier.

3.1.1.2 - Déroulement

L'analyse par le titulaire comporte des phases suivantes :

- Compréhension des besoins fonctionnels (qui peut nécessiter des enquêtes auprès des SPC et/ou des ateliers)
- Analyse technique, en tenant compte du cadrage technique du présent CCTP
- Proposition des scénarios avec leurs avantages et inconvénients et une estimation approximative des coûts de développement de ces scénarios
- Chiffrage du scénario retenu par le maître d'ouvrage (soit uniquement pour les spécifications, soit également pour le développement si c'est déjà possible à cette étape)

C'est un travail itératif. cf. 3.1.5 Relecture et validation

3.1.1.3 - Livrables

- Comptes-rendus des ateliers
- Document d'Analyse Fonctionnelle (DAF)
- Devis :
 - pour les spécifications générales
 - pour les spécifications techniques détaillées
 - pour les développements (facultatif, éventuellement commun au devis précédent)

NB : Les DAF n'ont pas besoin d'être mis à jour par la suite. cf. 3.1.4.2 Documentation

3.1.2 - Spécifications (SPEC)

Selon la complexité et l'ampleur des évolutions, la phase d'analyse (ANA) pourra être suffisamment détaillée pour servir de spécifications (générales et détaillées) et on pourra enchaîner directement sur la phase de développements (DEV).

3.1.2.1 - Spécifications générales

3.1.2.1.1 - Objectifs

Les objectifs attendus pour cette phase de spécifications fonctionnelles et techniques générales sont les suivants :

- Traduire les besoins du Service Central Vigicrues en exigences fonctionnelles et techniques
- Concevoir une architecture générale fonctionnelle et technique de la solution couvrant ces besoins.
- Initier la stratégie de bascule et/ou de mise en œuvre

C'est un travail itératif. cf. 3.1.5 Relecture et validation

3.1.2.1.2 - Déroulement

Le titulaire préparera et conduira les études et les ateliers de travail afin de réaliser cette phase.

3.1.2.1.3 - Livrables

Le livrable attendu pour cette phase de spécifications fonctionnelles et techniques générales est le dossier de spécifications fonctionnelles générales.

Afin de permettre une bonne articulation avec les autres développements menés dans le cadre du Consortium Open Telemac, il conviendra de créer un ticket dans le gitlab opentelemac (en anglais) pour permettre de tracer l'évolution prévue.

3.1.2.2 - Spécifications détaillées

3.1.2.2.1 - Objectifs

Cette prestation comporte les phases suivantes :

- Affiner les besoins fonctionnels du maître d'ouvrage
- Concevoir une architecture fonctionnelle et technique de la solution

C'est un travail itératif. cf. 3.1.5 Relecture et validation

3.1.2.2.2 - Déroulement

Cette phase de spécifications fonctionnelles et techniques détaillées comprend au minimum les étapes suivantes :

- Ateliers de recueil des besoins pour le paramétrage/développement de la solution adaptée au contexte et besoins du Service Central Vigicrues
 - Le titulaire animera les ateliers conformément à la démarche et la méthodologie détaillées dans son offre technique.
- Rédaction des spécifications fonctionnelles détaillées :
 - Une description détaillée des fonctionnalités avec cas d'usage.
 - Une description de l'architecture technique cible et des objectifs et contraintes techniques à prendre en compte.
 - La modélisation détaillée des données.
 - Définition du modèle de données envisagé et description de leurs propriétés.
 - Description du contexte de reprise des données (source, tables destinataires, volume, règles de mapping, fréquence, responsable).
- Présentation de maquettes des IHM
- Rédaction des spécifications fonctionnelles détaillées des interfaces :
 - Une description détaillée des données.
 - Une description détaillée des règles de gestion.
- Description détaillée de l'architecture technique, avec intégration dans le SI existant.
- Le schéma documenté de la base de données.
- La stratégie de bascule.
- Toutes les réunions nécessaires

Les étapes liées à la création et la présentation des maquettes sont un moment important du projet. Il est essentiel que ces maquettes décrivent précisément les futures IHM.

Les objets utilisés, les listes déroulantes, les couleurs utilisées, les enchaînements des écrans pour décrire les cas d'utilisation, seront clairement présentés dans les maquettes.

3.1.2.2.3 - Livrables

- Dossier de spécifications fonctionnelles et techniques détaillées :
 - Le modèle de données détaillé.
 - Les définitions des règles de gestion, calculs et contrôles.
 - La description de l'environnement technique.

- Les cas d'usage formalisés.
- Les comptes rendus des ateliers.
- Les maquettes des IHM.
- Le plan des tests :
 - La méthodologie.
 - Le déroulement des tests.
 - Le cahier des tests de la VA qui permet de tester les cas d'usage.
- Le document d'architecture technique (DAT) :
 - Vue fonctionnelle.
 - Vue logique.
 - Vue physique.
 - Vue opérationnelle.

3.1.3 - Développements (DEV)

3.1.3.1 - Objectifs

Les objectifs attendus pour cette phase de paramétrage et de développement sont les suivants :

- Développer et paramétrer les systèmes en répondant à l'ensemble des exigences.
- Compléter le plan de tests si nécessaire.

3.1.3.2 - Déroulement

Afin que le maître d'ouvrage garde la visibilité sur les développements en cours, il est demandé lors des cycles de développement (issue de chaque sprint) de présenter :

- le pourcentage global d'avancement
- des démonstrations des nouvelles fonctionnalités

Pour réduire l'effet tunnel et permettre de réduire le risque de problèmes lors de la validation, le prestataire intégrera les retours du maître d'ouvrage au cours des différents sprints (cycles de spécifications/développements).

Pour éviter les régressions fonctionnelles ou de performance, il est demandé d'effectuer un cycle d'intégration continue automatisé et des tests de performance.

3.1.3.3 - Livrables

Afin d'être recevables, les livraisons devront contenir l'ensemble des fourniture de la version :

- Les codes des applications et des tests unitaires à déposer sur le dépôt défini en phase d'initialisation (voir chapitre Erreur : source de la référence non trouvée)
- Le dossier de validation interne du titulaire (recette usine) :
 - Journal des corrections et des évolutions (notes de version)

- Compatibilité navigateurs (cf. 2.5.1.3 Cadre de cohérence technique spécifique du ministère)
- Le cahier des tests :
 - Le détail, les modalités d'exécution de chaque type de test.
 - Les jeux de tests de chaque type de test.
 - Liste et résultats des tests unitaires de performance, tests fonctionnels.
 - Tests de charge, uniquement pour la livraison finale.
 - Tests de sécurité.

Fournir des métriques de qualité et de sécurité du code permettant de certifier l'absence de vulnérabilité.

- Une archive de l'application prête à installer avec son script d'automatisation (forme à adapter selon les produits), livrée par un cycle d'intégration continue automatisé
- Des procédures de mises à jour automatiques cf. 2.5.9 Migrations. Une documentation décrira les manipulations à effectuer.

Le service central Vigicrues se réserve le droit d'auditer par un cabinet extérieur la qualité, la performance et la sécurité de l'application.

3.1.4 - Livraisons

3.1.4.1 - Procédures de livraison

Les modalités techniques de livraison seront définies lors de la réunion de lancement. La documentation détaillée et contextualisées à l'environnement d'hébergement ainsi qu'une automatisation de tout ce qui peut l'être sera exigée.

Bordereau de livraison :

Un bordereau de livraison accompagnera toutes les livraisons (logiciels et livrables documentaire). Il comportera :

- la liste des livrables
- la liste des évolutions, adaptations et corrections prises en compte (en revoyant éventuellement vers le bug tracker pour une description plus complète)
- le compte rendu des tests (recette usine) effectués avant la livraison d'un logiciel (cf. 3.1.3.3 Livrables)
- les signatures de validation des parties prenantes garantissant un respect du plan qualité (recette usine, circuit de relecture interne...)

Démonstrations :

Une démonstration devra être effectuée (éventuellement en partage d'écran à distance) avant les premières livraisons et lors des phases d'itération en cours de développement.

Livraisons majeures :

Pour les livraisons majeures (par exemple avec mise à jour de l'OS de machines virtuelles ou de certains composants), le déplacement du prestataire sur site (au service central Vigicrues) pourra être exigé par le service central Vigicrues afin de réaliser, avec les équipes du service central Vigicrues,

l'installation de l'application au service central Vigicrues.

Livraisons mineures :

Les installations mineures (sans mise à jour de composants majeurs) pourront être réalisées par le service central Vigicrues et les SPC suite à la livraison de tous les éléments dont les documentations d'installation et d'exploitation précises et contextualisées à l'environnement d'hébergement (service central Vigicrues, Dreal). Le prestataire doit prévoir une assistance (disponibilité) à l'installation de l'application par téléphone, mail ou présence sur site.

3.1.4.2 - Documentation ⁴

Mise à jour de toute documentation en phase avec le contenu de livraison ;

- Manuel d'installation
- Manuel utilisateur
- Spécifications générales selon la complexité et l'ampleur des évolutions
- Spécifications détaillées selon la complexité et l'ampleur des évolutions
- Dossier d'architecture technique (DAT) selon la complexité et l'ampleur des évolutions
- Documentation du code source cf. 2.5.2.10 Qualité de documentation et journaux

Les documents seront fournis en format modifiable de préférence aux formats libre office (odt) et acrobat reader (pdf) et le cas échéant sur la version en ligne intégré à l'application.

Attention à bien distinguer la documentation des versions de production (stables) et de test qui seront à maintenir en parallèle.

3.1.5 - Relecture et validation

Pour les prestations de type ANA et SPEC, il convient de mettre en place une relecture itérative des documents, avec des livraisons intermédiaires quand cela est possible. Des réunions (éventuellement téléphonique ou visio) seront également parfois nécessaires pour éviter de perdre du temps dans de trop nombreux aller-retours.

La validation finale par le maître d'ouvrage permet de disposer de documents de référence pour la suite. Les documents cités dans le paragraphe « 3.1.4.2 Documentation » devront être mis à jour au fur et à mesure des livraisons.

3.1.6 - Recette

Pour les prestations de Développements (DEV) (cf. 3.1.3),

Le délai d'exécution de chaque étape (VA/VSR) sera précisé dans les bons de commande. Par défaut il est de 2 mois pour la VA et 3 mois pour la VSR.

Une mise à jour de la documentation (cf. 3.1.4.2) devra être faite en fin de VA et en fin de VSR.

Les prestations de maintenance évolutive sont soumises à des vérifications qualitatives qui ont pour but de contrôler la conformité des prestations exécutées aux spécifications et stipulations de l'accord-cadre et leur bon fonctionnement en exploitation. Ces vérifications concernent donc le logiciel, mais également la documentation. Les opérations de vérifications qualitatives des prestations de maintenance évolutive comprennent trois étapes :

- Phase de réception et d'installation

⁴ #Exigence de niveau A (indispensable)

- Vérification d'Aptitude (VA)
- Vérification de Services Réguliers (VSR)

Une anomalie est considérée comme :

- *Bloquante si elle empêche une utilisation de l'application ou d'une partie essentielle de l'application ;*
- *Majeure si certaines fonctions importantes ne sont plus accessibles ou si l'utilisation de l'application est rendue difficile (par exemple : problèmes de performance, etc.) ;*
- *Mineure si l'application peut fonctionner, malgré une fonction mineure impactée, ou si le problème peut être contourné.*

En cas de désaccord, la personne publique décide au final de la nature de l'anomalie.

3.1.6.1 - Phase de réception et d'installation

Objectifs

Les principaux objectifs de cette phase de réception et d'installation sont d'installer le système et toutes les interfaces associées.

Déroulement

La phase de réception et d'installation se déroule après la livraison de la phase de développement.

L'installation des applicatifs est réalisée sur l'environnement de recette du service central Vigicrues.

Le titulaire doit assurer le support pendant ces phases d'installation et si besoin se déplacera sur site.

Livrables attendus

Le bordereau de livraison et d'installation (Mise en ordre de marche) émis par le titulaire.

Le service central Vigicrues vérifie la présence de tous les livrables de la phase de développement (cf. 3.1.3.3) et l'accès aux outils développés, en particulier les IHM et valide la fin de cette phase via un PV qui permet le démarrage de la phase de VA.

3.1.6.2 - Vérification d'Aptitude (VA)

Les modalités des opérations de vérification sont décrites à l'article 8 « Vérifications » du CCAP.

3.1.6.3 - Vérification de Services Réguliers (VSR)

Les modalités des opérations de vérification sont décrites à l'article 8 « Vérifications » du CCAP.

3.2 - Accompagnement (ACC)

Les prestations d'accompagnement permettront de couvrir les besoins suivants des utilisateurs, voire des développeurs, de Mascaret et Telemac 2D : formations, assistances, réunions, diagnostics, installations, mises à jour, transfert de connaissance, aide à l'autonomie. Elles pourront nécessiter une présence au service central Vigicrues ou dans les SPC (voire CVH).

Exemples (non exhaustif) :

- aide aux spécifications du PIM et du PIT

- aide à l'installation et/ou l'exploitation de Mascaret et Telemac
- mise à jour de la VM Telemac-Mascaret et documentation associée
- aide à la compréhension de cas de plantage
- aide à la conception de modèle Mascaret ou Telemac
- adaptation des tests de supervision, de performance
- diagnostic de fonctionnement et d'amélioration de performance

En plus des frais d'accompagnement dont le tarif est fixé par le bordereau des prix, les frais complémentaires de déplacements et d'hébergement pourront être intégrés aux devis des prestations d'accompagnement.

4 - Gestion de projet ⁵

Dans le cadre de la prestation, le titulaire du marché désignera un référent unique, **chef de projet**. En cas de changement de référent en cours d'exécution de la prestation, le titulaire désignera, après l'accord du service central Vigicrues, un nouveau responsable pour le représenter.

Un **directeur de projet**, côté prestataire, s'assurera de la bonne exécution des prestations et assurera l'affectation des ressources compétentes nécessaires au respect des plannings.

Le principe général qui guidera les décisions sera un **pilotage par les délais et la valeur**, à savoir qu'en cas de difficulté à tenir le planning on agira plutôt sur une réduction du périmètre des livraisons que sur un allongement du planning.

Toutes les réunions nécessaires à la gestion de projet seront à la charge du titulaire.

Ainsi, lors de la définition des périmètres des prestations qui feront l'objet de devis puis de bons de commandes, il sera important de distinguer les niveaux d'exigence suivants :

- A indispensable
- B importante
- C souhaitable

4.1 - Comité de suivi (Cosui)

Un comité de suivi sera effectué régulièrement (a priori par visio ou téléphone). La fréquence de cette réunion sera hebdomadaire et pourra être adaptée selon les besoins.

Le titulaire assurera la présence des personnes compétentes à la réunion (chef de projet, et si besoin responsable qualité, responsables des principales tâches, et le cas échéant de ses co-traitants et de ses sous-traitants) afin de garantir un bon déroulement et un fonctionnement efficace de celle-ci.

Il se déroulera en deux phases :

- Point de gestion de projet

Le prestataire présentera via un « tableau de bord » la mise à jour des éléments suivants associés aux différentes commandes :

- Pour les différents livrables : les étapes en cours (analyse, spécification, développement, recette interne), le pourcentage d'avancement associé, les dates de livraisons (intermédiaires avec description de leur périmètre ; complète avec la documentation), les personnes en charge des actions, etc.
- MCO, VA, VSR : groupement des corrections et dates de livraison prévues (intermédiaires avec description de leur périmètre ; complète avec la documentation)
- Procès verbaux de validation des différents livrables
- Paiements
- Identifications des difficultés rencontrées et action à entreprendre
- etc.
- Point de suivi des tickets

Le titulaire fournit :

- les indicateurs de résolution

⁵ #Exigence de niveau A (indispensable)

- le suivi des tickets en cours sera effectué en s'appuyant sur l'espace de gestion des tickets cf.chapitre 2.5.2.4.

Livrable attendu :

- tableau de bord mis à jour

4.2 - Comité de pilotage élargi

Une réunion de comité de pilotage (COPIL) élargi (avec participation du directeur de projet, chef de projet, responsable qualité, responsables des principales tâches, et le cas échéant de ses co-traitants et de ses sous-traitants) sera effectuée tous les 3 mois.

Le titulaire proposera, au plus tard 3 jours ouvrés avant la réunion, les points particuliers qu'il souhaite intégrer à l'ordre du jour de la réunion, et le support de présentation. Le Service Central Vigicrues sera libre d'ajuster l'ordre du jour et de rajouter les points souhaités.

Ces réunions permettront de faire un point d'avancement⁶ du projet et passeront systématiquement en revue les aspects suivants :

- Compte-rendu d'activité concernant les différents type de prestations (ANA, SPEC, DEV, ACC), respects des plannings, retours générés par l'environnement de développement et de test
- Avancement des actions en cours
- Planning détaillé du projet
- Tâches de la période à venir
- Identification des difficultés rencontrées et actions à entreprendre
- Bilan des tickets d'anomalie créés depuis le dernier Copil
- Risques et le cas échéant stratégie de réduction du risque
- Coordination et gestion des interfaces entre les différentes parties prenantes
- Suivi administratif et contractuel
- Satisfaction client

Ces réunions feront l'objet d'un compte-rendu établi par le titulaire. Dans ce dernier cas, le compte rendu sera transmis pour approbation au Maître d'Ouvrage et son représentant au plus tard quatre jours ouvrés après la réunion.

Une fois par an, une revue de l'environnement de développement permettra d'identifier les mises à jour nécessaires tant au niveau du cadre technique que des produits mis à disposition par le maître d'ouvrage.

⁶ #Exigence de niveau B (important)

5 - Clauses environnementales

Le titulaire s'engage à fournir les livrables suivants dans les délais indiqués :

Le prestataire détaillera dans un document spécifique (Cahier des clauses environnementales) l'ensemble des actions et moyens mis en œuvre pour limiter l'impact environnemental de la prestation, tels que détaillé dans le cadre de réponse technique. Ce document inclura en particulier un chapitre clairement identifié des actions menées par le prestataire dans les domaines suivants et le titulaire devra respecter cet ordre de présentation :

1) La stratégie d'écoconception	- Description des pratiques d'optimisation énergétique de l'application (réduction de la charge serveur, allègement des éléments transférés, etc.) - Principes de codage écoresponsable envisagés pour limiter la consommation des ressources.
2) Les engagements en matière de réduction des Gaz à Effet de Serre	Plan d'actions pour réduire l'empreinte carbone de l'application et des livrables associés, y compris le recours à des infrastructures à faible impact carbone et les objectifs chiffrés de réduction d'émissions.
3) Les solutions d'écomobilité	Détail des moyens mis en place pour réduire l'impact des déplacements, avec une priorisation des interventions à distance et l'utilisation de transports à faible émission pour les déplacements inévitables.
4) Les actions de formation et de sensibilisation des consultants en matière d'impact carbone de leur activité, et d'écoconception des services numériques	Liste des formations

Le suivi des clauses environnementales est à mener tout le long de l'accord cadre. Une livraison des documents suivants devra être faite 6 mois avant la fin de la période ferme et, en cas de reconduction, 6 mois avant la fin de la période de reconduction.

- Évaluation des évolutions du code et tests de performance pour vérifier que l'impact environnemental (consommation des ressources serveurs) du logiciel n'évolue pas défavorablement et de manière non justifiée par des besoins métiers ou des contraintes liées au SI du maître d'ouvrage, auquel cas des travaux d'optimisation seront couverts par la garantie sur les développements récents.
- Bilan des émissions de Gaz à Effet de Serre

Le titulaire est soumis à l'article L.229-25 du code de l'environnement, et doit communiquer son Bilan d'Emission de Gaz à Effet de Serre (BEGES) et le plan de transition associé.

- mesure d'impact environnementale des infrastructures mobilisées, empreinte carbone des trajets domicile / travail...
 - dans le cadre du projet (et si possible de manière plus globale pour l'agence locale)
- Justificatifs des formations environnementales suivies par les membres de l'équipe projet (au

minimum une formation doit être suivie chaque année par un membre de l'équipe).

Les pénalités pour non exécution sont détaillées à l'article 14.3 du CCAP.