

Grille d'évaluation individuelle RNCP

RNCP 37827 — DÉVELOPPEUR.SE EN INTELLIGENCE ARTIFICIELLE

Date de session :

Nom du candidat :

Prenom du candidat :

Compétences	Attendu	Critère	Acquis / Non acquis
Bloc de compétences 1 : Réaliser la collecte, le stockage et la mise à disposition des données d'un projet en intelligence artificielle			
C1. Automatiser l'extraction de données depuis un service web, une page web (scraping*), un fichier de données, une base de données et un système big data* en, programmant le script* adapté afin de pérenniser la collecte des données nécessaires au projet.	E1. Mise en situation (C1, C2, C3, C4, C5) L'évaluation doit se faire dans un contexte de réalisation d'un service numérique réel ou fictif basé sur l'usage de données, à partir du cadrage pour la réalisation d'un service numérique (spécifications fonctionnelles et techniques par exemple). Le projet évalué a pour but d'optimiser, d'automatiser, de pérenniser et de mettre à disposition les flux de données et les données, utiles et nécessaires à la réalisation du service numérique, par les équipes techniques (par exemple en analyse statistique, en business intelligence, en machine learning ou encore en intelligence artificielle). <u>Livrable</u> : rapport professionnel individuel <u>Évaluation</u> : - Correction du rapport professionnel - Soutenance orale individuelle	- La présentation du projet et de son contexte est complète : acteurs, objectifs fonctionnels et techniques, environnements et contraintes techniques, budget, organisation du travail et planification. - Les spécifications techniques précisent : les technologies et outils, les services externes, les exigences de programmation (langages), l'accessibilité (disponibilité, accès). - Le périmètre des spécifications techniques est complet : il couvre l'ensemble des moyens techniques à mettre en œuvre pour l'extraction et l'agrégation des données en un jeu de données brutes final. - Le script d'extraction des données est fonctionnel : toutes les données visées sont effectivement récupérées à l'issue de l'exécution du script. - Le script comprend un point de lancement, l'initialisation des dépendances et des connexions externes, les règles logiques de traitement, la gestion des erreurs et des exceptions, la fin du traitement et la sauvegarde des résultats. - Le script d'extraction des données est versionné* et accessible depuis	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<p>un dépôt Git*.</p> <ul style="list-style-type: none"> - L'extraction des données est faite depuis un mix entre au moins les sources suivantes : un service web (API REST), un fichier de données, un scraping, une base de données et un système big data. 	
<p>C2. Développer des requêtes de type SQL d'extraction des données depuis un système de gestion de base de données et un système big data en appliquant le langage de requête propre au système afin de préparer la collecte des données nécessaires au projet.</p>		<ul style="list-style-type: none"> - Les requêtes de type SQL pour la collecte de données sont fonctionnelles : les données visées sont effectivement extraites suite à l'exécution des requêtes. - La documentation des requêtes met en lumière choix de sélections, filtres, conditions, jointures, etc., en fonction des objectifs de collecte. - La documentation explicite les optimisations appliquées aux requêtes. 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS
<p>C3. Développer des règles d'agrégation de données issues de différentes sources en programmant, sous forme de script, la suppression des entrées corrompues et en programmant l'homogénéisation des formats des données afin de préparer le stockage du jeu de données final.</p>		<ul style="list-style-type: none"> - Le script d'agrégation des données est fonctionnel : les données sont effectivement agrégées, nettoyées et normalisées en un seul jeu de données à l'issue de l'exécution du script. - Le script d'agrégation des données est versionné et accessible depuis un dépôt Git. - La documentation du script d'agrégation est complète : dépendances, commandes, les 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		enchaînements logiques de l'algorithme, les choix de nettoyage et d'homogénéisation des formats données.	
C4. Créer une base de données dans le respect du RGPD en élaborant les modèles conceptuels et physiques des données à partir des données préparées et en programmant leur import afin de stocker le jeu de données du projet.		<ul style="list-style-type: none"> - Les modélisations des données respectent la méthode et le formalisme Merise. - Le modèle physique des données est fonctionnel : il est intégré avec succès lors de la création de la base de données, sans erreur. - La base de données est choisie au regard de la modélisation des données et des contraintes du projet. - La reproduction des procédures d'installation décrites (base de données et API) a pour résultat un système conforme aux objets techniques attendus.. - Le script d'import fourni est fonctionnel : il permet l'insertion des données dans le système mis en place. - La documentation technique du script d'import est versionné à la racine du même dépôt Git que celui utilisé pour le script d'import. <ul style="list-style-type: none"> - Les documentations techniques des scripts couvrent les parties suivantes : <ul style="list-style-type: none"> - les dépendances nécessaires pour la réutilisation des scripts (langages, dépendances externes, etc) - les commandes pour 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<p>l'exécution des scripts.</p> <ul style="list-style-type: none"> - Le registre des traitements de données personnelles intègre l'ensemble des traitements de données personnelles impliqués dans la base de données. - Les procédures de tri des données personnelles pour la mise en conformité de la base de données avec le RGPD sont rédigées. - Les procédures de tri détaillent les traitements de conformité (automatisés ou non) à appliquer ainsi que leur fréquence d'exécution. 	
<p>C5. Développer une API mettant à disposition le jeu de données en utilisant l'architecture REST afin de permettre l'exploitation du jeu de données par les autres composants du projet.</p>		<ul style="list-style-type: none"> - La documentation technique de l'API (REST) couvre tous les points de terminaisons. - La documentation technique couvre les règles d'authentification et/ou d'autorisation de l'API. - La documentation technique respecte les standards du modèle choisi (par exemple OpenAPI*). - L'API REST est fonctionnelle pour l'accès aux données du projet : elle restreint par une autorisation (ou authentification) l'accès aux données, - L'API REST est fonctionnelle pour la mise à disposition : elle permet la récupération de l'ensemble des données nécessaires au projet, comme prévu selon les spécifications données. 	<p><input type="checkbox"/> ACQUIS</p> <p><input type="checkbox"/> NON ACQUIS</p>

Compétences	Attendu	Critère	Acquis / Non acquis
Bloc de compétences 2 : Intégrer des modèles et des services d'intelligence artificielle			
C6. Organiser et réaliser une veille technique et réglementaire en animant le travail collectif de sélection des sources, de collecte, de traitement et de partage des informations afin de formuler des recommandations pour le projet toujours en phase avec l'état de l'art.	E2. Cas pratique (C6, C7, C8) L'évaluation doit se faire à partir de l'expression d'un besoin réel ou fictif de fonctionnalités d'intelligence artificielle. Ce besoin peut résulter d'une commande client comme d'une sollicitation interne d'un collaborateur data scientist par exemple. Le cas pratique évalué a pour but l'installation et la configuration du service d'intelligence artificielle préconisé. <u>Livrable</u> : rapport professionnel individuel <u>Évaluation</u> : - Correction du rapport professionnel - Soutenance orale individuelle	<ul style="list-style-type: none"> - La thématique de veille choisie porte sur un outil et/ou une réglementation mobilisée dans la mise en situation. - Les temps de veille sont planifiés régulièrement (à minima une récurrence d'une heure hebdomadaire). - Le choix des outils d'agrégation est cohérent avec les sources d'informations visées et le budget disponible (flux RSS, flux réseaux sociaux, agrégation newsletter, etc) - Les synthèses sont communiquées aux parties prenantes dans un format qui respecte les recommandations d'accessibilité (par exemples celles de l'association Valentin Haüy¹ ou de Atalan - AcceDe²). - Les informations partagées dans la synthèse répondent à la thématique de veille choisie. 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

¹ <https://www.avh.asso.fr/fr/favoriser-laccessibilite/accessibilite-numerique/accessibilite-des-documents-et-des-courriels-0>

² <https://www.accede-web.com/wp-content/uploads/accessibility-guidelines-editors-template.pdf>

		<ul style="list-style-type: none"> - Les sources et flux identifiés répondent aux critères de fiabilité : <ul style="list-style-type: none"> - L'auteur de la page est identifié - Des informations sur l'auteur sont disponibles et confirment ses compétences, sa notoriété et l'absence d'intérêts personnels - l'analyse du contenu est valable (date de publication récente, sources de l'information indiquées, niveau de langue correct), - la source (site) ou le document est structuré - les sources (sites) ou documents respectant les normes d'accessibilités sont privilégiés. - l'information peut être confirmée par d'autres sites de confiance 	
<p>C7. Identifier des services d'intelligence artificielle préexistants à partir de l'expression de besoin en fonctionnalités d'intelligence artificielle, en réalisant un <i>benchmark</i> de services existants et en analysant leurs caractéristiques pour formaliser une ou plusieurs recommandations de services adaptés au besoin.</p>		<ul style="list-style-type: none"> - L'expression de besoin est reformulée et présente les objectifs et les contraintes du projet d'intégration d'une solution d'intelligence artificielle. - Le <i>benchmark</i> liste les services étudiés et les services non étudiés. - Les raisons pour écarter un service sont explicitées. 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<ul style="list-style-type: none"> - Le <i>benchmark</i> détaille le niveau d'adéquation du service étudié pour chaque ensemble fonctionnel souhaité par le commanditaire. - Le <i>benchmark</i> détaille le niveau de la démarche éco-responsable du service étudié, en fonction des informations disponibles. - Le <i>benchmark</i> détaille les principales contraintes techniques et les pré-requis pour chaque solution. - Les conclusions délimitent clairement les services répondant aux besoins, avec leurs avantages et leurs inconvénients, des services ne couvrant pas les besoins du commanditaire. 	
C8. Paramétrer un service d'intelligence artificielle en suivant sa documentation technique et en respectant les spécifications du projet, afin de permettre l'intégration des connecteurs du service dans le système d'information.		<ul style="list-style-type: none"> - Le service installé est accessible, avec une éventuelle authentification. - Le service est configuré correctement, il répond aux besoins fonctionnels et aux contraintes techniques du projet. - Le monitoring disponible du service est opérationnel. - La documentation couvre la gestion des accès à la solution, les 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<p>procédures d'installation et de test, les éventuelles dépendances et interconnexions avec d'autres solutions, les données impliquées dans l'utilisation de la solution.</p> <p>- La documentation est communiquée aux parties prenantes dans un format qui respecte les recommandations d'accessibilité (par exemples celles de l'association Valentin Haüy³ ou de Atalan - AcceDe⁴).</p>	
<p>C9. Développer une API exposant un modèle d'intelligence artificielle en utilisant l'architecture REST pour permettre l'interaction entre le modèle et les autres composants du projet.</p>	<p>E3. Mise en situation (C9, C10, C11, C12, C13) L'évaluation doit se faire dans un contexte réel ou fictif de réalisation d'un service d'intelligence artificielle à partir d'un modèle fourni. Le projet évalué a pour but la mise en service (packaging*, monitoring, test...) du modèle fourni, et son intégration dans une application existante. <u>Livrable</u> : rapport professionnel individuel <u>Évaluation</u> :</p> <ul style="list-style-type: none"> - Correction du rapport professionnel - Soutenance orale individuelle intégrant une démonstration du projet. 	<ul style="list-style-type: none"> - L'API restreint l'accès au modèle d'intelligence artificielle avec un moyen d'authentification. - L'API permet l'accès aux fonctions du modèle, comme attendu selon les spécifications. - Les recommandations de sécurisation d'une API du top 10 OWASP sont intégrées quand nécessaires. - Les sources sont versionnées et accessibles depuis un dépôt Git distant. - Les tests couvrent tous les points de terminaison dans le respect des 	<p><input type="checkbox"/> ACQUIS</p> <p><input type="checkbox"/> NON ACQUIS</p>

³ <https://www.avh.asso.fr/fr/favoriser-laccessibilite/accessibilite-numerique/accessibilite-des-documents-et-des-courriels-0>

⁴ <https://www.accede-web.com/wp-content/uploads/accessibility-guidelines-editors-template.pdf>

		spécifications. - Les tests s'exécutent sans bug. - Les résultats des tests sont correctement interprétés. - La documentation couvre l'architecture et tous les points de terminaisons de l'API. - La documentation couvre les règles d'authentification et/ou d'autorisation d'accès à l'API. - La documentation et l'API respectent les standards d'un modèle choisi (par exemple Open API*). - La documentation est communiquée dans un format qui respecte les recommandations d'accessibilité (par exemple celles de l'association Valentin Haüy ou de Microsoft).	
C10. Intégrer l'API d'un modèle ou d'un service d'intelligence artificielle dans une application, en respectant les spécifications du projet et les normes d'accessibilité en vigueur, à l'aide de la documentation technique de l'API, afin de créer les fonctionnalités d'intelligence artificielle de		- L'application de départ est installée et fonctionnelle en environnement de développement. - La communication avec l'API depuis l'application fonctionne.	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

l'application.		<ul style="list-style-type: none"> - Les éventuelles étapes d'authentification et de renouvellement de l'authentification (expiration des jetons par exemple) sont intégrées correctement en suivant la documentation de l'API. - Tous les points de terminaison de l'API concernés par le projet sont intégrés à l'application selon les spécifications fonctionnelles et techniques. - Les adaptations d'interfaces nécessaires et en accord avec les spécifications sont intégrées à l'application. - Les tests d'intégration couvrent tous les points de terminaison exploités. - Les tests s'exécutent en totalité : il n'y a pas de bug dans les programmes des tests en eux-mêmes. - Les résultats des tests sont correctement interprétés. - Les sources sont versionnées et accessibles depuis le dépôt Git de l'application. 	
C11. Monitorer un modèle d'intelligence artificielle à partir des métriques courantes et spécifiques au projet, en intégrant les outils de collecte, d'alerte et de restitution des données du monitoring pour permettre l'amélioration du modèle de façon itérative.		<ul style="list-style-type: none"> - Les métriques faisant l'objet du monitoring du modèle sont expliquées sans erreur d'interprétation. 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<ul style="list-style-type: none">- Le ou les outils pour l'intégration du monitoring du modèle sont adaptés au contexte et aux contraintes techniques du projet.- Au moins un vecteur de restitution des métriques évaluées, en temps réel, est proposé (dashboard, feuille de calcul, etc).- Les enjeux d'accessibilité, pour toutes les parties prenantes du projet, sont pris en compte lors de la sélection de l'outil de restitution- La chaîne de monitoring est d'abord testée dans un bac à sable ou environnement de test dédié.- La chaîne de monitoring est en état de marche. Les métriques visées sont effectivement évaluées et restituées.- Les sources sont versionnées et accessibles depuis un dépôt Git distant- La documentation technique de la chaîne de monitoring couvre la procédure d'installation de la chaîne, de configurations, et d'utilisation du monitoring à destination des équipes techniques- La documentation est communiquée dans un format qui respecte les recommandations d'accessibilité (par exemple celles	
--	--	---	--

		de l'association Valentin Haüy ou de Microsoft).	
<p>C12. Programmer les tests automatisés d'un modèle d'intelligence artificielle en définissant les règles de validation des jeux de données, des étapes de préparation des données, d'entraînement, d'évaluation et de validation du modèle pour permettre son intégration en continu et garantir un niveau de qualité élevé.</p>		<ul style="list-style-type: none"> - L'ensemble des cas à tester sont listés et définis : la partie du modèle visée par le test, le périmètre du test et la stratégie de test. - Les outils de test (framework, bibliothèque, etc.) choisis sont cohérents avec l'environnement technique du projet. - Les tests sont intégrés et respectent la couverture souhaitée établie. - Les tests s'exécutent sans problème technique en environnement de test. - Les sources sont versionnées et accessibles depuis un dépôt Git distant (DVC, Gitlab...). - La documentation couvre la procédure d'installation de l'environnement de test, les dépendances installées, la procédure d'exécution des tests et de calcul de la couverture. - La documentation est communiquée dans un format qui respecte les recommandations d'accessibilité (par exemple celles de l'association Valentin Haüy ou de Microsoft). 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

<p>C13. Créer une chaîne de livraison continue d'un modèle d'intelligence artificielle en installant les outils et en appliquant les configuration souhaitées, dans le respect du cadre imposé par le projet et dans une approche MLOps*, pour automatiser les étapes de validation, de test, de <i>packaging</i>* et de déploiement du modèle.</p>		<ul style="list-style-type: none"> - La documentation pour l'utilisation de la chaîne couvre toutes les étapes, les tâches et tous les déclencheurs disponibles. - Les déclencheurs sont intégrés comme préalablement définis. - Le ou les fichiers de configuration de la chaîne sont correctement reconnus et exécutés par le système selon les déclencheurs configurés. - L'étape de test des données est intégrée à la chaîne et s'exécute sans erreur. - La ou les étapes de test, d'entraînement et de validation du modèle sont intégrées à la chaîne et s'exécutent sans erreur. - Les sources de la chaîne sont versionnées et accessibles depuis le dépôt Git distant du projet - La documentation de la chaîne de livraison continue couvre la procédure d'installation, de configuration et de test de la chaîne - La documentation est communiquée dans un format qui respecte les recommandations d'accessibilité (par exemple celles de l'association Valentin Haüy ou de Microsoft). 	<p><input type="checkbox"/> ACQUIS</p> <p><input type="checkbox"/> NON ACQUIS</p>
--	--	--	---

Compétences	Attendu	Critère	Acquis / Non acquis
Bloc de compétences 3 : Réaliser une application intégrant un service d'intelligence artificielle			
C14. Analyser le besoin d'application d'un commanditaire intégrant un service d'intelligence artificielle , en rédigeant les spécifications fonctionnelles et en le modélisant, dans le respect des standards d'utilisabilité et d'accessibilité, afin d'établir avec précision les objectifs de développement correspondant au besoin et à la faisabilité technique.	E4. Mise en situation (C14, C15, C16, C17, C18, C19) L'évaluation doit se faire dans un contexte réel ou fictif de développement d'une application intégrant un service d'intelligence artificielle. Le projet évalué a pour but d'analyser un besoin en développement d'application d'intelligence artificielle, de concevoir, développer, tester et enfin livrer l'application. <u>Livrable</u> : rapport professionnel individuel <u>Évaluation</u> : - Correction du rapport professionnel - Soutenance orale individuelle intégrant une démonstration du projet.	<ul style="list-style-type: none"> - La modélisation des données respecte un formalisme : Merise, entités-relations, etc. - La modélisation des parcours utilisateurs respecte un formalisme : schéma fonctionnel, wireframes, etc. - Chaque spécification fonctionnelle couvre le contexte, les scénarios d'utilisation et les critères de validation. - Les objectifs d'accessibilités sont directement intégrés aux critères d'acceptation des <i>user stories</i>. - Les objectifs d'accessibilité sont formulés en s'appuyant sur un des standards d'accessibilité : WCAG, RG2AA, etc. 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS
C15. Concevoir le cadre technique d'une application intégrant un service d'intelligence artificielle , à partir de l'analyse du besoin, en spécifiant l'architecture technique et applicative et en préconisant les outils et méthodes de développement, pour permettre le développement du projet.	L'évaluation doit se faire à partir d'une application existante présentant au moins une erreur technique, en contexte réel ou fictif. Le cas pratique évalué a pour but la mise en place du monitoring applicatif et de la résolution d'un incident technique dans l'application. <u>Livrable</u> : documentation technique du	<ul style="list-style-type: none"> - Les spécifications techniques rédigées couvrent l'architecture de l'application, ses dépendances et son environnement d'exécution (langage de programmation, framework, outils, etc). 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

	<p>monitorage. Documentation de la résolution de l'incident technique.</p> <p><u>Évaluation</u> :</p> <ul style="list-style-type: none"> - Correction de la documentation - Soutenance orale individuelle présentant le monitoring de l'application et la solution implémentée en réponse à l'incident technique traité. 	<ul style="list-style-type: none"> - Les éventuels services (PaaS, SaaS, etc) et prestataires ayant une démarche éco-responsable sont favorisés lors des choix techniques. - Les flux de données impliqués dans l'application sont représentés par un diagramme de flux de données. - La preuve de concept est accessible et fonctionnelle en environnement de pré-production. - La conclusion à l'issue de la preuve de concept donne un avis précis permettant une prise de décision sur la poursuite du projet. 	
<p>C16. Coordonner la réalisation technique d'une application d'intelligence artificielle en s'intégrant dans une conduite agile de projet et un contexte MLOps et en facilitant les temps de collaboration dans le but d'atteindre les objectifs de production et de qualité.</p>		<ul style="list-style-type: none"> - Les cycles, les étapes de chaque cycle, les rôles, les rituels et les outils de la méthode agile appliquée sont respectés dans sa mise en place et tout au long du projet. - Les outils de pilotage (tableau kanban, burndown chart, backlog, etc.) sont disponibles dans les conditions prévues par la méthode appliquée. - Les objectifs et les modalités des rituels sont partagés à toutes les parties prenantes et rappeler si besoin. - Les éléments de pilotage sont 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		rendus accessibles à toutes les parties du projet et ce tout au long du projet, en accord avec les recommandations de la méthode de gestion de projet appliquée.	
C17. Développer les composants techniques et les interfaces d'une application en utilisant les outils et langages de programmation adaptés et en respectant les spécifications fonctionnelles et techniques, les standards et normes d'accessibilité, de sécurité et de gestion des données en vigueur dans le but de répondre aux besoins fonctionnels identifiés.		<ul style="list-style-type: none"> - L'environnement de développement installé respecte les spécifications techniques du projet. - Les interfaces sont intégrées et respectent les maquettes. - Les comportements des composants d'interface (validation formulaire, animations, etc.) et la navigation respectent les spécifications fonctionnelles. - Les composants métier sont développés et fonctionnent comme prévu par les spécifications techniques et fonctionnelles. - La gestion des droits d'accès à l'application ou à certains espaces de l'application est développée et respecte les spécifications fonctionnelles. - Les flux de données sont intégrés dans le respect des spécifications techniques et fonctionnelles. - Les développements sont réalisés dans le respect des bonnes pratiques d'éco-conception d'une application (Les recommandations d'éco-index ou Green IT par exemple) - Les préconisations du top 10 d'OWASP sont implémentées dans 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<p>l'application quand nécessaire.</p> <ul style="list-style-type: none"> - Des tests d'intégration ou unitaires couvrent au moins les composants métier et la gestion des accès. - Les sources sont versionnées et accessibles depuis un dépôt Git distant. - La documentation technique couvre l'installation de l'environnement de développement, l'architecture applicative, les dépendances, l'exécution des tests. - La documentation est communiquée dans un format qui respecte les recommandations d'accessibilité (par exemple celles de l'association Valentin Haüy ou de Microsoft). 	
<p>C18. Automatiser les phases de tests du code source lors du versionnement des sources à l'aide d'un outil d'intégration continue* de manière à garantir la qualité technique des réalisations.</p>		<ul style="list-style-type: none"> - La documentation pour l'utilisation de la chaîne couvre les outils, toutes les étapes, les tâches et tous les déclencheurs de la chaîne. - Un outil de configuration et d'exécution d'une chaîne d'intégration continue est sélectionné de façon cohérente avec l'environnement technique du projet. - La chaîne intègre toutes les étapes nécessaires et préalables à l'exécution des tests de l'application (build, configurations...). 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<ul style="list-style-type: none"> - La chaîne exécute les tests de l'application disponibles lors de son déclenchement. - Les configuration sont versionnées avec les sources du projet d'application, sur un dépôt Git distant. - La documentation de la chaîne d'intégration continue couvre la procédure d'installation, de configuration et de test de la chaîne. - La documentation est communiquée dans un format qui respecte les recommandations d'accessibilité (par exemple celles de l'association Valentin Haüy ou de Microsoft). 	
C19. Créer un processus de livraison continue d'une application en s'appuyant sur une chaîne d'intégration continue et en paramétrant les outils d'automatisation et les environnements de test afin de permettre une restitution optimale de l'application.		<ul style="list-style-type: none"> - La documentation pour l'utilisation de la chaîne couvre toutes les étapes de la chaîne, les tâches et tous les déclencheurs disponibles. - Le ou les fichiers de configuration de la chaîne sont correctement reconnus et exécutés par le système. - La ou les étapes de <i>packaging</i> (compilation, minification, <i>build</i> de containers, etc.) de l'application sont intégrées à la chaîne et s'exécutent sans erreur. 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<ul style="list-style-type: none"> - L'étape de livraison (<i>pull request</i> par exemple) est intégrée et exécutée une fois la ou les étapes de packaging validées. - Les sources de la chaîne sont versionnées et accessibles depuis le dépôt Git distant du projet d'application. - La documentation de la chaîne de livraison continue couvre la procédure d'installation, de configuration et de test de la chaîne. - La documentation est communiquée dans un format qui respecte les recommandations d'accessibilité (par exemple celles de l'association Valentin Haüy ou de Microsoft). 	
<p>C20. Surveiller une application d'intelligence artificielle, en mobilisant des techniques de monitoring et de journalisation, dans le respect des normes de gestion des données personnelles en vigueur, afin d'alimenter la <i>feedback loop</i>* dans une approche MLOps, et de permettre la détection automatique d'incidents.</p>		<ul style="list-style-type: none"> - La documentation liste les métriques et les seuils et valeurs d'alerte pour chaque métrique à risque. - La documentation explicite les arguments en faveur des choix techniques pour l'outillage du monitoring de l'application. - Les outils (collecteurs, journalisation, agrégateurs, filtres, dashboard, etc.) sont installés et opérationnels à minima en environnement local. 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		<ul style="list-style-type: none"> - Les règles de journalisation sont intégrées aux sources de l'application, en fonction des métriques à surveiller. - Les alertes sont configurées et en état de marche, en fonction des seuils préalablement définis. - La documentation couvre la procédure d'installation et de configuration des dépendances pour l'outillage du monitoring de l'application. - La documentation est communiquée dans un format qui respecte les recommandations d'accessibilité (par exemple celles de l'association Valentin Haüy ou de Microsoft). 	
C21. Résoudre les incidents techniques en apportant les modifications nécessaires au code de l'application et en documentant les solutions pour en garantir le fonctionnement opérationnel.		<ul style="list-style-type: none"> - La ou les causes du problème sont identifiées correctement. - Le problème est reproduit en environnement de développement. - La procédure de débogage du code est documentée depuis l'outil de suivi. - La solution documentée explicite chaque étape de la résolution et de son implémentation. - La solution est versionnée dans le dépôt Git du projet d'application (par 	<input type="checkbox"/> ACQUIS <input type="checkbox"/> NON ACQUIS

		exemple avec une <i>merge request</i>).	
--	--	--	--

Décision du jury (entourer la mention utile)

Admis - Non admis - Admission partielle

Commentaires (optionnel, à renseigner obligatoirement si admission partielle)

.....

.....

.....

.....

.....

Détails des blocs acquis en cas de validation partielle (à renseigner obligatoirement si admission partielle)

Dates et signatures de l'ensemble des membres du jury