

Expert en ingénierie du logiciel

BLOC 1 GESTION DE LA RELATION CLIENT

BLOC 1 GESTION DE LA RELATION CLIENT			
REFERENTIEL D'ACTIVITES <i>décrit les situations de travail et les activités exercées, les métiers ou emplois visés</i>	REFERENTIEL DE COMPETENCES <i>identifie les compétences et les connaissances, y compris transversales, qui découlent du référentiel d'activités</i>	REFERENTIEL D'EVALUATION <i>définit les critères et les modalités d'évaluation des acquis</i>	
		MODALITES D'ÉVALUATION	CRITÈRES D'ÉVALUATION
A1 - Analyse du besoin <ul style="list-style-type: none"> - Organisation de la réunion - Recueil et analyse du besoin - Communication sur la solution / produit / service 	C.1. Organiser et conduire une réunion en pratiquant l'écoute active et en identifiant la typologie du client pour affiner sa demande et construire un argumentaire adapté à ses besoins	Mise en situation professionnelle simulée : À partir d'une situation vécue en entreprise, le candidat organise et anime une réunion de ciblage du besoin en français et/ou en anglais	Mise en situation professionnelle simulée : Identification du besoin <ul style="list-style-type: none"> - Les objectifs de la réunion sont clairement déterminés - Les prises de parole sont pertinentes et opportunes - L'écoute active est pratiquée - Les questions posées permettent d'identifier la typologie du client et de recueillir le besoin - La demande du client est reformulée avec clarté, bien analysée et identifiée avec précision - Les besoins de l'entreprise sont compris - Les arguments utilisés sont en adéquation avec la demande du client - Les critères d'achat du client sont identifiés - Les premiers éléments communiqués sur la solution / produit / service sont précis et adaptés à la situation du client

<ul style="list-style-type: none"> - Diagnostic de l'écosystème du client - Etude de la faisabilité et de l'opportunité du projet - Identification de la problématique - Elaboration d'une proposition 	<p>C.2. Analyser les besoins du client en réalisant un diagnostic de son écosystème interne et en procédant à une étude de faisabilité afin d'identifier les problèmes à l'origine du besoin et faire émerger des pistes de solutions.</p>	<p>Cas pratique : À partir d'un scénario donné, le candidat mène un diagnostic de l'écosystème de l'entreprise, identifie la problématique et propose des solutions en français et/ou en anglais</p>	<p>Cas pratique : Étude de l'écosystème interne et propositions de solutions</p> <ul style="list-style-type: none"> - La méthodologie d'analyse utilisée permet d'évaluer le fonctionnement global de l'écosystème de l'entreprise, tout en prenant en compte ses contraintes - La problématique du client à l'origine du besoin est bien identifiée - L'étude de faisabilité prend en compte les aspects techniques, organisationnels et financiers - Les sources de création de valeur sont identifiées et exploitées - Les solutions proposées répondent aux besoins du client, sont réalisables, chiffrées et argumentées
<p>A.2 - Conseil sur les technologies, outils, services</p> <ul style="list-style-type: none"> - Entretien d'une veille technologique et autoformation continue - Diagnostic des technologies et services - Présélection des solutions les plus pertinentes - Dimensionnement technique de la solution - Conseil sur les choix technologiques (rédaction d'un support) 	<p>C.3. Analyser l'environnement technologique en menant une étude comparative des moyens techniques et services existants de façon à identifier les pistes de solutions et répondre aux besoins du client.</p>	<p>Cas pratique : À partir d'un cas d'entreprise, le candidat mène une étude comparative des technologies et services de l'environnement et rédige un support en français et/ou en anglais</p>	<p>Cas pratique : Analyse et sélection d'une solution, technologie et /ou service</p> <ul style="list-style-type: none"> - La collecte des données permet d'identifier les tendances et d'anticiper les évolutions technologiques - Les choix de solution technologique effectués prennent en compte l'état de l'art ainsi que les besoins du client - L'étude comparative prend en compte les risques, impacts et avantages. - Les préconisations apportées sont explicitées et intègrent le dimensionnement technique de la solution - Capacité à comprendre et argumenter le choix des technologies utilisées (celui-ci afin d'apporter une valeur ajoutée au discours proposé au client) - Pertinence des choix proposés - Les technologies retenues et proposées au client sont pertinentes, comprises et argumentées - Analyser une fiche technique pour voir si le produit est pertinent et répond aux besoins. - La réponse apportée concilie les meilleures pratiques actuelles (état de l'art) et la spécificité du contexte client

<p>A.3 - Proposition d'une solution</p> <ul style="list-style-type: none"> - Conception du modèle d'architecture de la solution - Rédaction des spécifications fonctionnelles et techniques - Conception de maquettes 	<p>C.4. Concevoir une solution à partir des besoins exprimés par la modélisation d'une architecture, afin de la présenter au client et de l'assister sur sa prise de décision</p>	<p>Cas pratique : À partir d'un cas d'entreprise, le candidat réalise des schémas d'architecture et rédige une réponse détaillée au cahier des charges en français et/ou en anglais</p>	<p>Cas pratique : conception et modélisation de la solution</p> <ul style="list-style-type: none"> - Les besoins fonctionnels et non fonctionnels de la solution sont identifiés - Les besoins techniques et attributs de qualité de la solution sont identifiés - Les besoins fonctionnels utilisateurs sont compris et traduits en besoins techniques - Les besoins et / ou expressions des besoins sont formalisés (cahier des charges ou spécifications fonctionnelles avec le formalisme nécessaire) - Les documents produits : expression de besoins, cahier des charges, spécifications fonctionnelles sont jugés satisfaisants - Les choix de technologies mis en œuvre dans le prototype ou preuve de concept sont pertinents et permettent de lever les interrogations techniques ou fonctionnelles.
<p>A.4 - Analyse du retour sur investissement</p> <ul style="list-style-type: none"> - Evaluation des coûts de la solution - Evaluation des gains apportés par la solution - Evaluation des coûts de l'existant 	<p>C.5. Réaliser une estimation des coûts de la solution en prenant en compte les dépenses d'exploitation et les gains non quantifiables, afin d'évaluer le retour sur investissement du (des) projet(s) et déterminer la rentabilité de la solution envisagée.</p>	<p>Cas pratique : le candidat étudie l'opportunité de faisabilité du projet (estimation des coûts, rentabilité) en français et/ou en anglais</p>	<p>Cas pratique : Évaluation de l'opportunité de faisabilité</p> <ul style="list-style-type: none"> - Les éléments avancés sont argumentés et bien mis en évidence - L'estimation des coûts de la solution est juste (dépenses d'exploitation...) - Le chiffrage des gains apportés par l'implémentation prend en compte les dépenses d'exploitation et d'investissement et les gains non quantifiables - Les choix de solution technologique effectués apportent des bénéfices substantiels au client

BLOC 2 - PILOTAGE DE PROJET D'INGÉNIERIE LOGICIEL

REFERENTIEL D'ACTIVITES <i>décrit les situations de travail et les activités exercées, les métiers ou emplois visés</i>	REFERENTIEL DE COMPETENCES <i>identifie les compétences et les connaissances, y compris transversales, qui découlent du référentiel d'activités</i>	REFERENTIEL D'ÉVALUATION <i>définit les critères et les modalités d'évaluation des acquis</i>	
		MODALITÉS D'ÉVALUATION	CRITÈRES D'ÉVALUATION
A.1 - Priorisation des fonctionnalités à livrer <ul style="list-style-type: none"> - Formalisation des fonctionnalités adaptées à l'organisation de travail du client - Hiérarchisation des fonctionnalités - Définition des critères d'acceptance et modalités de recettes - Estimation des charges, délais et du budget - Identification et définition des principes de contractualisation. 	C.6. Etablir une liste priorisée des fonctionnalités à livrer dans un cadre contractuel en tenant compte des exigences du projet afin de fournir le produit (ou le service) qui permet d'apporter une valeur ajoutée aux besoins du client.	Cas pratique : Le candidat tire au hasard un scénario de projet et élabore les spécifications / liste des fonctionnalités ¹ du projet en français et/ou en anglais et analyse les éventuelles failles contractuelles	<ul style="list-style-type: none"> - Les fonctionnalités sont décrites précisément et exhaustivement. - Les fonctionnalités sont testables - Les fonctionnalités sont priorisées - Les estimations de charges, délais, budget sont identifiées et partagées par les parties prenantes - Les différentes modalités contractuelles sont assimilées
A.2 - Planification de l'exécution <ul style="list-style-type: none"> - Identification des dates et des jalons (tests, recettes, livraisons, mise en production, évolutions, maintenance) - Identification des parties prenantes, des rôles et responsabilités 	C.7. Planifier l'exécution du projet en déterminant le cadre méthodologique, les rôles des parties prenantes et le planning prévisionnel permettant de converger vers l'atteinte des objectifs du projet.	Mise en situation professionnelle simulée : Le candidat tire au sort un type de réunion ou d'atelier lancement, et anime cette réunion	<ul style="list-style-type: none"> - Aucun jalon n'est omis - Toutes les parties prenantes sont identifiées - Les indicateurs de qualité sont précis et mesurables - L'ordre du jour est bien défini - Le compte rendu rédigé est complet et structuré - La démarche d'amélioration s'inscrit dans la durée

¹ * product backlog ou cahier des charges selon l'approche organisationnelle

<ul style="list-style-type: none"> - Mise en place de la qualité et d'une démarche d'amélioration continue - Etablissement du planning prévisionnel ajustable - Estimation des charges et des délais 			<ul style="list-style-type: none"> - Les modalités de restitution sont adaptées à la situation et au client - Le planning prévisionnel tient compte de toutes les contraintes et de toutes les évolutions possibles (aléas, charges, délais, disponibilités, etc.). Il comprend plusieurs modalités d'adaptation. - Une prochaine réunion de vérification et de recadrage de ce planning est planifiée
<p>A.3 - Suivi de l'avancement</p> <ul style="list-style-type: none"> - Identification des indicateurs - Suivi des indicateurs - Présentation de l'avancée du projet et des indicateurs - Gestion des risques opérationnels et juridiques - Mesure de la productivité - Elaboration de support de présentation 	<p>C.8. Assurer le suivi du projet en mesurant régulièrement l'avancée du projet avec les outils et méthodes appropriés afin d'appliquer des mesures adaptatives et garantir la satisfaction du client et des utilisateurs.</p>	<p>Cas pratique : A partir d'un recueil de documents, le candidat définit les indicateurs, les modalités de suivi de ceux-ci, et identifie les risques opérationnels et juridiques</p>	<ul style="list-style-type: none"> - Les indicateurs d'avancement sont spécifiques, mesurables, atteignables, réalistes et définis dans le temps - Les indicateurs sont utilisables et le candidat sait les lire et les corrélérer les uns avec les autres. - La démarche de suivi de l'avancement du projet est structurée et permet de mesurer objectivement son avancée - Les risques sont identifiés et catégorisés et sont adressés avec un plan de mitigation - Les données du système sont gérées de bout en bout (récolte, exploitation, archive, destruction, RGPD)
<p>A.4 - Animation de l'équipe</p> <ul style="list-style-type: none"> - Constitution de l'équipe - Développement de la collaboration entre les parties prenantes. - Création d'un environnement de travail efficace - Gestion d'équipes distantes ou multiples - Gestion du temps en situation de projet 	<p>C.9. Faciliter l'organisation et l'environnement de travail de son espace et/ou celui de l'équipe en mettant en œuvre des actions d'amélioration, de communication et/ou de responsabilisation afin d'impliquer toutes les parties prenantes dans l'atteinte des objectifs du projet.</p>	<p>Etude de cas : A partir d'un cas fictif présentant des défauts ou manquements dans le positionnement des membres de l'équipe, le candidat établit des préconisations.</p>	<ul style="list-style-type: none"> - Les rôles et responsabilités sont connus et partagés - Les forces des membres de l'équipe sont identifiées et utilisées au mieux - La collaboration est rendue fluide par l'utilisation d'outils adéquats (le candidat est force de proposition pour les outils) - Le rythme/ les objectifs des réunions sont judicieux et permettent un travail efficace - Les éventuels conflits (réels ou potentiels) sont identifiés et traités afin de ne pas entraver l'avancée du travail...

BLOC 3 – INTÉGRATION, INDUSTRIALISATION ET DÉPLOIEMENT DU LOGICIEL

REFERENTIEL D'ACTIVITES <i>décrit les situations de travail et les activités exercées, les métiers ou emplois visés</i>	REFERENTIEL DE COMPETENCES <i>identifie les compétences et les connaissances, y compris transversales, qui découlent du référentiel d'activités</i>	REFERENTIEL D'ÉVALUATION <i>définit les critères et les modalités d'évaluation des acquis</i>	
		MODALITÉS D'ÉVALUATION	CRITÈRES D'ÉVALUATION
<p>A1. Développement du logiciel</p> <ul style="list-style-type: none"> - Gestion et contrôle des versions - Intégration de composants externes - Écriture du code source - Rédaction de la documentation technique <ul style="list-style-type: none"> - Suivi des demandes utilisateurs et des clients - Exécution de la maintenance corrective et évolutive de l'application - Communication des modifications aux parties prenantes 	<p>C.10. Développer les fonctionnalités du logiciel en modélisant un domaine métier, et en intégrant des composants externes afin d'améliorer la qualité du code et faciliter les développements futurs.</p> <p>C.11. Assurer le suivi des demandes utilisateurs en exploitant des outils de suivi permettant de recenser les anomalies afin de procéder aux mesures correctives, évolutives et adaptatives de l'application.</p>	<p>Cas pratique : À partir des fonctionnalités données et de bibliothèques logicielles préexistantes, le candidat présente une documentation technique et argumente la qualité du code.</p> <p>Étude de cas : Dans le cadre d'un travail de groupe et à partir d'une situation donnée, le candidat propose les outils de suivi appropriés et formule des préconisations adaptées aux demandes des utilisateurs</p>	<ul style="list-style-type: none"> - Les bibliothèques, bibliothèques et Framework sont utilisés à bon escient - Les API (interfaces de programmation) sélectionnées sont adaptées au regard des fonctionnalités développées. - Le logiciel est livré sous forme d'un dépôt de code source versionné. - Les commentaires de documentation sont saisis aux bons endroits dans le code source. - La documentation technique est structurée, exhaustive et compréhensible - Le logiciel est intuitif (Facilité d'usage, ergonomie...) <ul style="list-style-type: none"> - Toutes les fonctionnalités usuelles de l'outil proposé sont utilisées et maîtrisées. - Les points forts et les limites de l'outil sont présentés. - Les évolutions et la maintenance sont conformes aux demandes utilisateurs. - Les préconisations formulées répondent aux problématiques des utilisateurs

<p>A2. Intégration continue de la solution logicielle</p> <ul style="list-style-type: none"> - Définition des indicateurs de qualité du logiciel - Mise en place des contrôles automatisés de la qualité du code - Automatisation des chaînes de build² - Exécution automatisée des tests 	<p>C.12. Automatiser la construction de la solution logicielle en configurant les chaînes de build et l'exécution des tests unitaires, fonctionnels et d'intégration afin de préparer le déploiement continu du logiciel.</p>	<p>Cas pratique (C.12 - C.13): À partir d'une plateforme imposée et d'un dépôt de code source réel ou fictif, le candidat :</p> <ul style="list-style-type: none"> - Configure une solution d'intégration continue et expose les métriques générées ; 	<ul style="list-style-type: none"> - Les indicateurs de build, tel que la couverture de code, sont présents et reflètent les résultats des tentatives de construction (automatisation des tests et contrôles) - Les étapes de construction sont configurées et leurs rapports de dépendances permettent l'exécution conditionnelle des scénarios d'intégration continue.
<p>A3. Déploiement continu de la solution logicielle</p> <ul style="list-style-type: none"> - Paramétrage / Configuration de l'outil d'intégration continue - Empaquetage de la solution à destination de plusieurs plateformes - Conteneurisation de la solution - Mise en œuvre des environnements d'exécution (développement, tests, production, ...) - Automatisation du processus de déploiement 	<p>C.13. Industrialiser le développement du logiciel⁴ à l'aide d'outils d'automatisation et le documenter en décrivant le processus de déploiement de manière à faire évoluer les logiciels développés et minimiser les erreurs de manipulation par les tiers.</p>	<ul style="list-style-type: none"> - Configure et automatise le déploiement d'un serveur web (réel ou fictif) sur trois environnements d'exécution avec validation manuelle et l'empaquetage d'une librairie³ (réelle ou fictive). 	<ul style="list-style-type: none"> - La procédure de déploiement est automatiquement déclenchée lorsqu'une contribution (commit) est détectée. - L'application et/ou la librairie n'est pas redéployée si l'une des étapes échoue. - L'application est déployée de manière progressive dans trois environnements d'exécution - La librairie est rendue disponible sur internet ou via un gestionnaire de paquets logiciel.

² chaîne de build : processus de génération du logiciel préalablement à son déploiement.

³ librairie : brique logicielle répondant à un besoin technique spécifique à destination d'autres développeurs.

⁴ software factory

BLOC 4 – CONCEPTION AVANCÉE DE L'ARCHITECTURE DU LOGICIEL

REFERENTIEL D'ACTIVITES <i>décrit les situations de travail et les activités exercées, les métiers ou emplois visés</i>	REFERENTIEL DE COMPETENCES <i>identifie les compétences et les connaissances, y compris transversales, qui découlent du référentiel d'activités</i>	REFERENTIEL D'ÉVALUATION <i>définit les critères et les modalités d'évaluation des acquis</i>	
		MODALITÉS D'ÉVALUATION	CRITÈRES D'ÉVALUATION
<p>A1. Conception de l'architecture de la solution logicielle</p> <ul style="list-style-type: none"> - Modélisation du domaine métier - Étude et sélection du ou des modèles d'architectures de la solution - Découplage du code métier et des dépendances logicielles - Réalisation de diagrammes d'architecture - Rédaction d'un dossier d'architecture - Utilisation de protocoles et de services d'authentification tiers 	<p>C.14. Concevoir une architecture adéquate, à partir des exigences et attributs de qualité en réalisant des diagrammes d'architecture et en les formalisant dans un support technique à destination de l'équipe de développement afin de faciliter son usage, son adoption, sa robustesse et son évolutivité.</p>	<p>Cas pratique : À partir d'un cas donné (réel ou simulé) présentant les exigences et attributs de qualité d'une solution logicielle, le candidat produit un support d'architecture adéquat et complet.</p>	<ul style="list-style-type: none"> - Les processus et le domaine métier sont identifiés et reproduits sous forme de diagrammes. - Le choix du modèle d'architecture (monolithique, micro-service, distribué, centralisé, fédéré) est argumenté. - Les interactions avec les systèmes informatiques externes sont intégrées à l'architecture. - Les dépendances logicielles sont découplées grâce aux principes SOLID. - Le choix de deux propriétés sur trois du théorème CAP⁵ est justifié. - Un diagramme de classe prenant en compte des design pattern appropriés est inclus dans le dossier. - L'authentification des utilisateurs est possible via des services d'authentification tiers (OAuth, OpenID). - Les recommandations sécurité web (de l'OWASP⁶) sont appliquées. - Les failles de sécurité sont corrigées par la mise à jour des dépendances logicielles.

⁵ CAP : Théorème CAP (Consistency, Availability, Partition tolerance).

⁶ OWASP : Open Web Application Security Project, communauté en ligne travaillant sur la sécurité des applications web.

<p>A2. Reprise de solutions historiques existantes</p> <ul style="list-style-type: none"> - Prise de connaissance des éléments et productions existants (code source, diagrammes) - Identification des choix architecturaux précédents - Reprise des données - Restructuration du code source 	<p>C.15. Développer les fonctionnalités d'un logiciel existant, en s'appropriant l'historique de la solution à faire évoluer, et prenant en compte les évolutions futures dont elle pourra faire l'objet à long terme dans le cadre d'une reprise par un tiers afin de garantir la pérennité du logiciel et s'assurer de son évolutivité dans le temps.</p>	<p>Étude de cas :</p> <p>Dans le cadre d'un travail de groupe, portant sur un cas de restructuration d'un logiciel (réel ou simulé), le candidat formule et argumente les changements à apporter et propose la restructuration du code source.</p>	<ul style="list-style-type: none"> - Les architectures, éléments et productions précédents sont bien identifiés (compréhension et appropriation de la solution existante) - Les éléments et productions précédents sont utilisés pour montrer leur inadéquation au nouveau contexte et pour justifier les changements proposés. - Un plan de migration (avec des jalons) est proposé. - Les différents moyens d'accès aux anciennes données, après remise en état de la solution, sont explicités - La nouvelle fonctionnalité est facilement maintenable et évolutive (diminution de la dette technique). - Le code est davantage lisible, maintenable et évolutif au regard des bonnes pratiques communément admises. - Des conventions de codages sont établies.
<p>A3. Assurance qualité du logiciel</p> <ul style="list-style-type: none"> - Sélection et définition des structures de données - Implémentation des algorithmes - Mise en œuvre des patrons de conception - Gestion des problématiques de concurrence et de parallélisme 	<p>C.16. Implémenter un logiciel de qualité, en choisissant des structures de données adaptées et des algorithmes pertinents afin d'assurer la robustesse du logiciel.</p>	<p>Cas pratique :</p> <p>Le candidat développe un logiciel avec des algorithmes concurrents et parallèles grâce aux structures de données et aux patrons de conceptions qu'il juge appropriés.</p>	<ul style="list-style-type: none"> - Les différents paradigmes de programmation choisis (objet, événementiel, fonctionnel, procédural, déclaratif) sont appropriés pour résoudre les problématiques de développements. - Les structures de données choisies facilitent le traitement algorithmique. - Les principes ACID⁷ sont appliqués - Les problématiques courantes et avancées d'algorithmique sont résolues grâce aux patrons de conception. - Les algorithmes concurrentiels appliqués permettent une exécution non-bloquante du logiciel. - L'exécution parallèle des algorithmes permet un traitement plus rapide des calculs et des interactions utilisateur conformément à l'état de l'art et aux patrons de conception, - Les notions et les problématiques de concurrence et de parallélisme sont pris en compte

⁷ ACID : Atomicité, Cohérence, Isolation, Durabilité. Ensemble de propriétés qui garantissent qu'une transaction informatique est exécutée de façon fiable.

<ul style="list-style-type: none"> - Catégorisation des tests à développer (régression, performance, fonctionnels) - Utilisation des tests appropriés en fonction de la partie du logiciel testé (unitaires, intégration) - Développement logiciel piloté par les tests 	<p>C.17. Tester le logiciel et l'application à plusieurs niveaux en utilisant les méthodologies de test éprouvées afin de garantir la conformité du logiciel au regard des spécifications et la non-régression des fonctionnalités déjà développées.</p>	<p>Cas pratique : Le candidat écrit les tests qui serviront à vérifier la conformité d'un code en vue de son implémentation future</p>	<ul style="list-style-type: none"> - Les tests unitaires couvrent les détails d'implémentation du modèle objet à développer. - Les tests d'intégration vérifient les propriétés générales du code à développer. - Les tests de performance mesurent l'évolution de la consommation des ressources pour des données d'entrée croissantes (linéaires, exponentielles).
<p>A4. Mise en œuvre de l'Intelligence Artificielle</p> <ul style="list-style-type: none"> - Labelliser des données - Classifier l'information - Générer des recommandations 	<p>C.18. Concevoir une application d'analyse de données massives en intégrant un programme d'apprentissage automatique (machine learning) au développement du logiciel et en utilisant des réseaux de neurones, des algorithmes d'optimisation et de recommandation afin de faire ressortir les tendances utilisateurs.</p>	<p>Cas pratique : Le candidat développe une application de recommandation d'articles à partir de photos des utilisateurs.</p>	<ul style="list-style-type: none"> - Le réseau de neurones permet d'identifier correctement les articles de l'utilisateur. - L'algorithme de recommandation permet de faire ressortir les habitudes d'achat des autres utilisateurs. - L'application recommande des articles selon la similarité de l'utilisateur avec d'autres utilisateurs qui ont les mêmes articles. - Les natures de problèmes ou de questions pouvant se résoudre avec le machine learning sont bien déterminées.
	<p>Examen final : toutes les compétences sont évaluées dans le cadre de la réalisation d'un portefeuille de compétences et une soutenance finale devant un jury professionnel</p>	<p>En lien avec les différents travaux réalisés tout au long du parcours et son expérience professionnelle, le candidat réalise un portefeuille de compétences portant sur l'ensemble des compétences de la certification</p> <p>Soutenance finale : le candidat restitue son travail à l'oral devant un jury</p>	<p>Le candidat apporte la preuve de la validation des 18 compétences du présent référentiel, en entreprise et en formation.</p> <p>Communication orale en mode soutenance :</p> <ul style="list-style-type: none"> - Prise en compte et respect de l'auditoire - Expression de la pensée / cohérence du discours / correction des énoncés - Argumentation / valorisation

			<ul style="list-style-type: none"> - Interaction avec l'auditoire durant la phase de questions-réponses - Gestion de son temps <p>Présentation de soi en tant qu'expert en ingénierie du logiciel :</p> <ul style="list-style-type: none"> - Maîtrise des sujets liés à son activité / à son métier - Prise de recul sur son parcours / autoévaluation - Affirmation de soi / positionnement personnel - Présentation étayée de ses compétences / valorisation de ses contributions - Construction de son évolution personnelle et professionnelle <p>Entretien (Questions/Réponses) avec le jury</p>
--	--	--	---

Conditions de validation de la certification :

Une soutenance finale de certification est organisée à la fin du parcours de formation pour chaque candidat afin de vérifier qu'il a bien validé les 18 compétences du présent référentiel.

Celle-ci repose sur la validation de son portefeuille de compétences comprenant :
le parcours d'alternance en entreprise (portfolio*),
la validation de tous les modules de formation.

Le candidat remet à la fin de son cursus le livret de son parcours en entreprise ainsi que les bulletins attestant de la validation des compétences.
L'accent est mis sur les évaluations formatives et l'auto-évaluation.

L'échelle d'évaluation étant la suivante :

- 3 = Maîtrise : le candidat est référent dans son domaine, capable d'évaluer et de transmettre
- 2 = Acquis : la mise en œuvre est en autonomie
- 1 = En cours d'acquisition : la mise en œuvre n'est pas autonome, le candidat a besoin d'être accompagné
- 0 = Non acquis

Pour obtenir la certification l'apprenant doit obtenir au moins le niveau 2 correspondant à acquis.

**Un portfolio est un dossier personnel dans lequel les acquis de formation et les acquis de l'expérience d'une personne sont définis et démontrés en vue d'une reconnaissance par un établissement d'enseignement ou un employeur.*