

## Concepteur Développeur d'Applications Numériques

Certification visée	CDAN fiche RNCP N° (36463)	Niveau	6
---------------------	----------------------------	--------	---

Sous la responsabilité d'un chef de projets, d'un DSI, d'un cadre dirigeant de PME, le Concepteur Développeur d'Applications Numériques conçoit, réalise et délivre les applications numériques attendues avec une autonomie variable suivant les contextes d'exercice. Il intègre son action au sein d'un projet informatique en tant que maîtrise d'œuvre. Son périmètre s'étend depuis l'étude de l'existant jusqu'à la mise en exploitation, en passant par la programmation de logiciels.

Il assure la conception de la solution logicielle en cohérence avec l'urbanisation du système d'information de l'entreprise. La conception de l'architecture des applications logicielles livrées respecte les contraintes de sécurité et garantit une robustesse maximale aux attaques externes. Il protège l'intégrité des données opérationnelles en réalisant des tests unitaires, et le recettage dans un environnement de test qu'il a réalisé.

Il garantit le respect de l'ergonomie et des normes d'accessibilité dans la réalisation d'interfaces homme/machine (IHM) et le respect des attentes des utilisateurs. C'est un communicant qui organise et anime des réunions, élabore et rédige des documents techniques, commerciaux ou internes à destination, des utilisateurs, des clients ou des collaborateurs, ...

### BC01 : Concevoir des applications numériques en intégrant les recommandations de sécurité.

Compétences attestées	Contextes et critères d'évaluation
Formaliser les procédures des services utilisateurs pour recenser les résultats attendus.	Le cahier des charges issu de l'étude de l'existant, conforme aux besoins du client, est validé par ce dernier. Le cahier des spécifications fonctionnelles contient la liste de contrôle des attendus fonctionnels, tels que notifiés par le client.
Prendre en compte les impératifs utilisateurs en respectant les contraintes des recommandations qualité de la norme en vigueur pour l'architecture des logiciels.	Le plan d'assurance qualité est observé et conforme aux normes choisies par le comité de pilotage du projet. L'application est organisée en couches indépendantes.
Concevoir l'architecture d'une solution fiable en identifiant les spécificités d'une activité pour produire du logiciel générique réutilisable. Concevoir des services d'accès aux données indépendants du mode de stockage en garantissant la sécurité des données pour produire du logiciel partageable.	Les règles métier sont encapsulées dans des services logiciels ou des web-services. L'accès aux données est réalisé par des services logiciels indépendants du mode de stockage.
Envisager toutes les possibilités, même les plus improbables pour livrer un logiciel déverminé en recherchant systématiquement l'erreur ou le dysfonctionnement.	Des outils de contrôle automatique du code sont utilisés. Aucun défaut visible ne persiste.
Estimer la charge de traitement et la puissance de calculs nécessaire proportionnellement aux nombre d'utilisateurs simultanés en vue d'anticiper les évolutions.	L'exécution de l'application est répartie entre un nombre d'ordinateurs adapté au contexte d'exploitation.
Respecter une norme de présentation des écrans et documents de sortie en utilisant les outils de maquettage appropriés en vue de permettre l'adaptabilité des sorties garantissant leur accessibilité à différents niveaux de handicap.	Une norme de présentation des données et de législation sont respectées tel que le RGAA. Les interfaces Homme/Machine sont validées par le comité de pilotage.
Identifier les risques et leur niveau de criticité pour permettre leur prévention.	Un formulaire d'estimation des risques est rempli permettant d'anticiper les conséquences sur l'activité de l'organisation.

Orienter son style de programmation en vue de produire du code lisible, maintenable, robuste, fiable, efficace par une approche méthodologique objets.	<p>Une programmation orientée objets est utilisée.</p> <p>Le taux de réutilisation du code utile est &gt; 80 %.</p> <p>Des gabarits sont utilisés.</p> <p>Une charte de nommage est utilisée.</p> <p>Le taux de documentation interne du code est &gt; 8 % et &lt; 15 %.</p>
Garantir un accès sécurisé aux données en évitant toute corruption de la base de données, par l'usage de contraintes d'intégrité et de déclencheurs.	<p>Les anomalies d'accès aux données ne génèrent pas d'interruption de l'exécution et sont répertoriées ou alertées selon leur niveau de gravité.</p> <p>Une méthode de conception est utilisée.</p>

## BC02 : Piloter un projet DevOps de développement d'application numérique.

Compétences attestées	Contextes et critères d'évaluation
Formaliser les procédures des services utilisateurs en contrôlant le respect du management des processus de l'entreprise.	<p>Le niveau de conformité de la procédure du service utilisateur aux règles du système de management des services de l'entreprise est établi.</p> <p>La décision de validation en l'état ou de réingénierie du processus est prise par le comité de pilotage.</p>
Réaliser une réingénierie d'un processus de l'entreprise en tenant compte des règles de management de l'entreprise dans un but d'amélioration des résultats et/ou des conditions de travail.	La proposition de reconstruction de la procédure est documentée et validée par le responsable fonctionnel.
Formaliser la circulation des documents générés en identifiant les acteurs concernés et leur rôle ainsi que les rubriques utilisées et leur provenance.	<p>La circulation du document résultat du traitement prévu est matérialisée dans un diagramme de workflow permettant d'identifier l'initialisation, les postes de modifications, les points de validation, le ou les destinataires et les critères et l'emplacement de stockage.</p> <p>Le dictionnaire des données est mis à jour respectant la norme choisie par le directeur du projet.</p>
Modéliser une base de données adaptée aux attentes en formalisant les règles de gestion et d'organisation de l'entreprise des processus concernés et en tenant compte d'un existant possible.	La base de données est modélisée, si besoin, en harmonie avec un existant, dans le respect de la méthode choisie par le directeur de projet..
S'insérer dans l'urbanisation présente et future du système d'informations en concevant des éléments logiciels réutilisables et structurés en couches.	<p>Une méthode de conception par objets est utilisée.</p> <p>Une structure en couches est produite dans le respect de la conception de l'architecture d'applications logicielles du projet..</p>
Choisir le degré de réutilisation à utiliser selon le cas de figure en décidant collectivement en équipe d'une réutilisation totale, partielle ou une écriture neuve et des briques à réutiliser en tentant de se rapprocher du niveau maximum de satisfaction CMMI.	La forme de construction de la solution logicielle est arrêtée et consignée dans le livre de bord du projet, tel que le sprint Backlog préconisé par la méthode SCRUM.
Utiliser l'expérience vécue pour anticiper l'avenir en vue d'estimer des délais de réalisation compte tenu du taux réel de disponibilité du réalisateur et des contraintes date départ/date livraison.	Les étapes du projet sont planifiées avec un outil permettant la synchronisation des équipes et des tâches affectées à chaque acteur du projet.
Coordonner un projet de développement en utilisant les outils et méthodologies de gestion de projet Agile afin de respecter les contraintes définies (coûts, délais, qualité), tout en minimisant les risques.	<p>Les obstacles entre le client et l'équipe développement sont levés.</p> <p>Le backlog est mis à jour à chaque sprint.</p> <p>L'information circule entre les membres de l'équipe développement.</p>
Clôturer une mission de développement en faisant valider le livrable par les parties concernées et en respectant les	Le PV de réception attestant la conformité du logiciel aux attendus a été validé par le client permettant la

préconisations CFTL.	mise en pré-production.
Adapter son discours à l'auditoire en appuyant ses manipulations et explications fonctionnelles à partir des contraintes exprimées tout au long du projet pour obtenir une bonne adhésion des décideurs ou de leurs représentants.	La présentation est appréciée par le client final. Le recettage est validé par le comité de pilotage.
Réaliser la procédure d'intégrabilité d'un logiciel ou d'un correctif dans l'environnement de tests selon les bonnes pratiques ITIL en vérifiant que l'intégralité des points de contrôles sont positifs.	Le bon à intégrer est signé du responsable d'exploitation. Le S.L.A. (service level agreement) ou son équivalent est respecté.
Interagir efficacement dans un environnement de travail collaboratif en reformulant la demande et en adaptant son discours à l'auditoire pour obtenir un niveau de compréhension des demandes optimum en tenant compte des collaborateurs en situation de handicap.  User d'une communication professionnelle tant en français qu'en anglais en structurant des informations sur une thématique donnée afin de les partager au sein de la structure ou à l'externe.	Le compte-rendu de la réunion est validé et est diffusé aux acteurs du projet.  Le document collectant l'expression des besoins des utilisateurs est validé par le comité de pilotage ou le Product Owner dans le cas d'utilisation de la méthode Scrum. Les documents produits sont explicites, concis, utiles aux destinataires.  Le niveau de conformité de la procédure du service utilisateur aux règles du système de management des services de l'entreprise est établi. La décision de validation en l'état ou de réingénierie du processus est prise par le comité de pilotage.

### BC03 : Développer des applications numériques.

Compétences attestées	Contextes et critères d'évaluation
Utiliser les ressources à sa disposition ou faire appel si besoin à un expert externe pour contrôler l'identification et la teneur du résultat attendu ou approfondir un point technique.	Les solutions trouvées ont permis de lever les doutes et les inconnus. L'objectif final et les moyens d'y parvenir doivent être clairement identifiés.
Décomposer un problème complexe en sous- problèmes en faisant des analogies et des différenciations tout en changeant d'approche, de point de vue, face à un obstacle en vue de résoudre un problème algorithmique.	Les résultats intermédiaires et leur mode d'obtention sont définis. La totalité des spécifications fonctionnelles ont été transcrits en algorithmes afin de rédiger le code source.
Traduire une solution algorithmique dans un langage de codage informatique avec l'utilitaire approprié.	La totalité des algorithmes ont été transcrits en code source afin d'effectuer les tests techniques.
Modifier un algorithme sans générer de dysfonctionnements en comprenant et s'adaptant si besoin au mode de pensée de son auteur.	Le code source modifié est prêt pour les tests techniques, le compilateur ou son équivalent ne détecte plus aucune erreur.
Remédier aux erreurs de codage ou de logique en comprenant ou interprétant les messages d'erreur du compilateur ou du système d'exploitation pour mettre au point un élément logiciel opérationnel.	Les tests techniques prouvent que le logiciel produit le résultat fonctionnel attendu. Les tests techniques prouvent que la modification n'entraîne pas de régression fonctionnelle.
Intégrer des éléments logiciels hétérogènes spécifiques et en réutilisant des services logiciels externes, en local ou à distance pour produire des exécutable livrables en conformité avec la politique RSE.	L'ensemble des éléments logiciels sont assemblés ou intégrés, l'unité de traitement est prête pour passer les tests unitaires.

Préparer des jeux d'essai en envisageant toutes les possibilités dans le but de livrer un logiciel déverminé exempt d'anomalies logiques et fonctionnelles.	Des outils de contrôle automatique du code sont utilisés. Les jeux de tests utilisés ne révèlent plus aucun défaut d'exécution.
Estimer son taux de disponibilité réel et rendre compte de son travail en renseignant l'outil de suivi permettant de constater l'avance de la tâche en cours et sa répercussion sur l'ensemble du projet.	Le compte-rendu d'activité est renseigné. Les écarts sont constatés. Le taux de disponibilité individuel est actualisé.

### BC04 : Réaliser une interface d'échange de données informatisées.

Compétences attestées	Contextes et critères d'évaluation
Procéder à une analyse organique d'un logiciel existant par l'étude du code des programmes et des données qui sont accédées dans le but de disposer d'une documentation technique du logiciel jusque-là inexistante ou indisponible.	La rétro-documentation est disponible, fiable, utilisable et permet la mesure des impacts de modifications des logiciels et des bases de données.
Faire des analogies et des différenciations entre les données à échanger entre logiciels à l'aide des dictionnaires de données disponibles ou reconstitués en vue de permettre l'échange de données entre les logiciels identifiés.	Les tables de correspondance des données sont opérationnelles, aucune référence n'est orpheline.
Produire des données indisponibles en agrégeant, consolidant ou calculant automatiquement ces nouvelles données à partir de celles existantes dans le but de favoriser les échanges de données entre logiciels dans le respect de la RGPD.	Les données agrégées sont consolidées, disponibles et conformes à la réglementation en vigueur.
Permettre l'exportation et l'importation de données entre logiciels en utilisant des formats compatibles entre les systèmes émetteurs et récepteurs grâce à des flux synchrones ou asynchrones.	Les flux de données entre logiciels sont effectifs et mesurables.
Écrire des scripts système en langage de commande ou en shell système pour automatiser l'installation, la configuration de systèmes d'exploitation et de middleware permettant la création, la configuration de machines virtuelles, de serveurs d'applications, Web et bases de données dans le but d'adapter et simuler en réel l'environnement d'exécution du logiciel à tester.	L'environnement de tests est opérationnel et répond aux besoins techniques et fonctionnels du cahier des charges.

