

Référentiel de compétences de la certification "Automatiser des processus dans les applications Microsoft Office avec VBA" (Tosa)

Modalités d'évaluation

Test numérique adaptatif Tosa.

- Typologies de questions : activités interactives (relier, glisser-déposer, cliquer sur zone, listes déroulantes, etc.), QCM et exercices de mises en situation rencontrées en entreprise via des exercices de manipulation d'un ou plusieurs logiciels dans leur environnement
- Format du test : 25 questions – 60 minutes , Questions réparties sur l'ensemble des 4 domaines.
- Algorithme : adaptatif (Le niveau des questions s'adapte au niveau du candidat)
- Scoring mathématique IRT (Item Response Theory) : score sur 1000.
- Conditions de passage de la certification : en ligne, surveillance en présentiel par un examinateur ou à distance par proctoring

Critères d'évaluation

Les compétences "Automatiser des processus dans les applications Microsoft Office avec VBA" (Tosa) sont divisées en 5 niveaux de maîtrise. Chaque niveau de maîtrise est déterminé en fonction de la complexité de la tâche et du degré d'autonomie de l'individu qui réalise cette tâche. Les niveaux sont catégorisés d'Initial (Niveau 1) à Expert (Niveau 5).

Important : Pour les candidats visant la reconnaissance d'un socle de compétences professionnelles et l'obtention de la certification, le seuil minimal de l'équivalent du niveau 3 (Opérationnel Tosa) est requis.

		Niveau Opérationnel (551-725)	Niveau Avancé (726-875)	Niveau Expert (876-1000)
Domaines	Compétences	<p>L'individu utilise VBA pour créer des procédures sophistiquées qui interagissent avec différentes applications Office. Il maîtrise les boucles, les conditions et est capable de manipuler les objets de l'application hôte avec aisance. Sa compréhension lui permet de travailler de manière autonome sur la majorité des projets VBA courants, et il peut débiter l'intégration de formulaires et de contrôles pour améliorer l'interface utilisateur</p>	<p>L'individu a une connaissance approfondie de VBA, lui permettant de développer des applications complexes intégrées dans l'environnement Microsoft. Il est compétent dans l'usage des API Windows pour étendre les fonctionnalités des applications et peut optimiser le code pour améliorer la performance et la sécurité des scripts. Il est également à l'aise pour créer et gérer des interfaces utilisateur complexes avec des formulaires personnalisés</p>	<p>L'individu déploie des compétences expertes en VBA, concevant des solutions automatisées complexes et innovantes adaptées à des besoins spécifiques de haut niveau. Il maîtrise le débogage avancé, la gestion des erreurs et la sécurisation des applications VBA. Il est capable de créer des add-ins personnalisés pour Microsoft, d'intégrer des applications avec d'autres services via des API et de transmettre ses connaissances avancées à d'autres développeurs</p>

1. Objets	<p>1.1 Manipuler les objets Excel</p> <p>Automatiser les processus dans Excel en utilisant et en gérant les principaux objets tels que Workbooks (cahiers de travail), Worksheets (feuilles de calcul), Ranges (plages de cellules), et Charts (graphiques), afin de simplifier les tâches spécifiques et optimiser la gestion des données.</p>	<p>Automatise des tâches fréquentes, Manipule des ranges dynamiquement en fonction des données entrées.</p> <p>Utilise des collections pour exécuter des actions complexes sur plusieurs objets.</p> <p>Écrit des scripts qui interagissent avec divers objets pour automatiser des rapports et des analyses.</p>	<p>Intègre des objets Excel avec d'autres applications Microsoft 365 via VBA pour automatiser l'envoi de rapports générés dans Excel par email via Outlook, améliorant ainsi l'efficacité de la communication des données clés aux parties prenantes.</p> <p>Développe des solutions personnalisées qui utilisent une large gamme d'objets Excel.</p>	<p>Conçoit des systèmes complexes qui manipulent intensivement des objets Excel pour automatiser des processus métiers.</p> <p>Est capable de diriger des projets d'automatisation intégrant Excel à d'autres technologies.</p>
	<p>1.2 Gérer les collections d'objets</p> <p>Gérer les groupes d'objets similaires dans Excel, comme les collections Worksheets ou Cells, en parcourant, ajoutant ou supprimant des éléments de la collection, et en utilisant des boucles afin d'automatiser les opérations répétitives sur chaque élément de la collection.</p>	<p>Utilise des collections avancées pour créer des interfaces utilisateur dynamiques et des rapports via VBA pour offrir des expériences interactives et personnalisées aux utilisateurs, facilitant la navigation et l'analyse des données complexes grâce à des formulaires et des tableaux de bord adaptatifs.</p>	<p>Optimise les performances et la gestion des collections pour des applications de grande envergure pour assurer un traitement rapide et efficace des données volumineuses, minimisant ainsi les temps de calcul et améliorant la réactivité des applications complexes.</p>	<p>Développe des solutions personnalisées qui utilisent des collections pour interagir avec des bases de données et des services web pour automatiser la récupération et l'intégration des données externes, facilitant ainsi la mise à jour et l'analyse des informations en temps réel.</p>
	<p>1.3 Interfacer avec d'autres applications</p> <p>Utiliser VBA pour interagir avec d'autres programmes Microsoft 365 tels que Word, Access et Outlook, afin d'automatiser des tâches inter-applications, comme l'exportation de données depuis Excel vers Word ou l'envoi d'emails via Outlook, en exploitant les objets COM (Component Object Model) pour fluidifier les processus.</p>	<p>Automatise des processus inter-applications via VBA pour améliorer l'efficacité opérationnelle en coordonnant les flux de travail entre différents logiciels Microsoft 365, réduisant ainsi les tâches manuelles répétitives et les erreurs associées</p> <p>Intègre des fonctionnalités complexes entre applications.</p>	<p>Développe des solutions intégrées personnalisées pour optimiser les flux de travail inter-applications pour harmoniser les processus métier, augmentant ainsi la productivité et la cohérence des données entre différentes plateformes Microsoft 365.</p>	<p>Crée des solutions d'automatisation qui utilisent VBA pour piloter des processus métiers à travers plusieurs applications Microsoft pour rationaliser les opérations, réduire les erreurs humaines et augmenter l'efficacité globale des tâches complexes impliquant Excel, Word, Outlook et Access.</p>
	<p>2.1 Écrire des procédures</p> <p>Créer des sous-routines et des fonctions en VBA pour exécuter des tâches spécifiques, en utilisant des arguments et des paramètres pour passer des données entre les procédures, afin d'améliorer la modularité et la réutilisabilité du code.</p>	<p>Crée des procédures VBA personnalisées pour automatiser des tâches répétitives permettant une personnalisation fine des actions automatisées pour répondre aux besoins spécifiques de l'utilisateur, améliorant ainsi l'efficacité et la productivité au sein de l'environnement de travail.</p>	<p>Utilise les collections pour manipuler des ensembles d'objets.</p> <p>Développe des fonctions complexes qui interagissent avec l'utilisateur et d'autres applications.</p> <p>Conçoit des bibliothèques de fonctions pour une utilisation globale dans des projets multiples.</p>	<p>Est capable d'innover avec des procédures qui automatisent et intègrent des processus métiers complexes, renforçant ainsi l'efficacité opérationnelle et la prise de décision stratégique.</p> <p>Peut former à l'écriture de procédures.</p>

2. Procédures	<p>2.2 Traiter les erreurs et les exceptions</p> <p>Mettre en place des mécanismes de gestion des erreurs en utilisant des structures et des blocs conditionnels pour anticiper et traiter les erreurs, et en développant des procédures de nettoyage et de récupération après une erreur, afin de rendre le code plus robuste et moins sujet aux plantages.</p>	<p>Crée des mécanismes de récupération d'erreur pour gérer les exceptions sans interruption de service.</p>	<p>Analyse et optimise les scripts pour prévenir les erreurs dans des environnements critiques en utilisant des techniques avancées de programmation VBA, telles que la gestion efficace des erreurs, la validation des données et l'optimisation des performances, garantissant ainsi la fiabilité et la stabilité des applications utilisées.</p>	<p>Développe des systèmes de logging et de reporting d'erreurs pour améliorer la maintenance et le débogage de solutions d'entreprise.</p>
	<p>2.3 Améliorer la performance du code</p> <p>Optimiser les routines VBA en minimisant les accès aux objets Excel, en utilisant des types de données appropriés, en optimisant les boucles et en gérant la mémoire de manière efficace, afin qu'elles s'exécutent plus rapidement et de manière plus performante.</p>	<p>Utilise des techniques d'optimisation avancées pour telles que la réduction du temps d'exécution des boucles, la minimisation des accès aux objets externes, et l'utilisation efficace des structures de données, assurant ainsi des performances optimales et une utilisation efficace des ressources dans les applications VBA.</p>	<p>Réécrit des procédures existantes pour optimiser la performance globale des applications en identifiant et en mettant en œuvre des améliorations telles que la réduction du nombre d'appels aux objets, l'optimisation des algorithmes et des structures de données, et la suppression de toute redondance inutile, assurant ainsi une exécution plus rapide et plus efficace des tâches dans les applications VBA.</p>	<p>Conçoit des algorithmes hautement optimisés et des structures de données pour des applications critiques en utilisant des techniques avancées de programmation et de gestion des données, telles que la mise en cache, la parallélisation et l'optimisation des accès aux données, garantissant ainsi des performances optimales et une utilisation efficace des ressources dans des environnements où la fiabilité et la rapidité sont primordiales.</p>
	<p>3.1 Concevoir des formulaires</p> <p>Créer des UserForms (formulaires utilisateur) en VBA en configurant l'interface utilisateur, en ajoutant des contrôles tels que des boutons, des étiquettes et des zones de texte, et en gérant leur disposition ainsi que leurs propriétés, afin de faciliter l'interaction avec l'utilisateur.</p>	<p>Intègre des UserForms dans des workflows complexes pour piloter des applications en créant des interfaces utilisateur conviviales et interactives qui permettent aux utilisateurs de saisir, de visualiser et de manipuler les données de manière efficace, facilitant ainsi la navigation et la gestion des processus métier au sein de l'environnement Microsoft 365.</p>	<p>Conçoit des interfaces utilisateurs interactives et professionnelles avec des UserForms avancés en intégrant des contrôles personnalisés, des mises en page sophistiquées et des fonctionnalités avancées telles que la validation des données et les interactions dynamiques, offrant ainsi une expérience utilisateur améliorée et une navigation intuitive dans les applications Microsoft 365.</p>	<p>Développe des solutions de formulaires personnalisés qui intègrent des technologies web et des bases de données via VBA, en utilisant des UserForms avancés pour collecter des données auprès des utilisateurs et les intégrer de manière transparente dans des systèmes en ligne ou des bases de données externes, permettant ainsi une communication fluide et une gestion efficace des informations à travers différents environnements.</p>

3. Boîtes de dialogue, formulaires et contrôles ActiveX	3.2 Implémenter des contrôles ActiveX Ajouter et gérer des contrôles ActiveX dans les formulaires ou directement dans les feuilles de calcul Excel, en insérant des contrôles tels que des boutons, des cases à cocher et des listes déroulantes, afin d'améliorer l'interactivité et la fonctionnalité des applications.	Développe des interfaces personnalisés en utilisant des contrôles ActiveX avancés en intégrant des fonctionnalités avancées telles que des graphiques interactifs, des tableaux dynamiques et des contrôles de navigation sophistiqués, offrant ainsi aux utilisateurs une expérience utilisateur riche et interactive au sein des applications Microsoft 365.	Intègre des contrôles ActiveX complexes pour interagir avec d'autres applications et services en utilisant des fonctionnalités avancées telles que la communication avec des bases de données distantes et la gestion de flux de travail entre différentes plateformes, offrant ainsi une intégration transparente et efficace des données et des fonctionnalités entre divers environnements logiciels.	Crée des contrôles ActiveX sur mesure pour des fonctionnalités spécifiques non disponibles commercialement en développant des composants personnalisés répondant aux besoins uniques d'une application, offrant ainsi des solutions sur mesure pour des tâches complexes ou des processus métiers spécialisés, améliorant ainsi l'efficacité et la flexibilité des applications Microsoft 365.
	3.3 Gérer les événements de formulaires Écrire du code pour répondre aux actions de l'utilisateur sur les formulaires, en utilisant les événements associés aux contrôles et aux formulaires, afin de déclencher des actions spécifiques et d'améliorer l'interactivité des applications.	Conçoit des flux de travail entièrement pilotés par des événements dans des formulaires en utilisant des mécanismes de gestion d'événements avancés pour déclencher des actions spécifiques en réponse aux interactions utilisateur, permettant ainsi la création d'interfaces utilisateur réactives et dynamiques qui guident les utilisateurs à travers des processus métier complexes de manière fluide et intuitive.	Utilise des événements complexes pour intégrer des formulaires à des processus métier complexes n développant des mécanismes de déclenchement d'événements sophistiqués qui permettent une interaction dynamique entre les utilisateurs et les systèmes, facilitant ainsi l'intégration harmonieuse des formulaires dans des workflows métier étendus et diversifiés.	Développe des solutions innovantes qui utilisent les événements des formulaires pour automatiser et optimiser des processus d'entreprise critiques.
	4.1 Utiliser les outils de débogage Connaître et utiliser les outils intégrés dans l'IDE VBA, tels que les points d'arrêt, la fenêtre d'exécution, la surveillance des variables et l'exécution pas à pas du code, afin d'identifier et de corriger les erreurs efficacement.	Utilise des techniques de débogage avancées pour optimiser et sécuriser des applications VBA en identifiant et en corrigeant les erreurs de manière efficace, en utilisant des outils tels que les points d'arrêt, les fenêtres de surveillance des variables et les messages de débogage, afin d'assurer la fiabilité et la robustesse des applications dans des environnements de production critiques.	Crée des outils personnalisés pour étendre les capacités de débogage de VBA dans des environnements complexes en développant des scripts et des macros supplémentaires qui améliorent les fonctionnalités de débogage de l'IDE VBA, fournissant ainsi des solutions sur mesure pour diagnostiquer et résoudre efficacement les problèmes dans des scénarios de développement et de déploiement variés.	Est capable de former d'autres développeurs sur les meilleures pratiques de débogage et de développement sécurisé.

4. Environnement et outils de débogage	<p>4.2 Analyser et suivre le code</p> <p>Utiliser les fonctionnalités de traçage et de journalisation en employant des instructions Debug.Print pour afficher des messages dans la fenêtre Exécution et en analysant les flux d'exécution, afin de comprendre le comportement du code, identifier les problèmes potentiels et optimiser les routines.</p>	<p>Applique des techniques de profiling pour analyser la performance et l'efficacité du code en utilisant des outils spécialisés ou des scripts personnalisés pour mesurer et évaluer les temps d'exécution, l'utilisation des ressources et d'autres métriques clés, permettant ainsi d'identifier les zones de code nécessitant une optimisation pour améliorer les performances globales de l'application VBA.</p>	<p>Analyse de grandes bases de code pour identifier et résoudre les problèmes de performance et de sécurité en utilisant des techniques avancées telles que l'inspection manuelle et l'automatisation à l'aide de scripts personnalisés, permettant ainsi de détecter et de corriger les vulnérabilités potentielles, les erreurs de programmation et les goulots d'étranglement qui pourraient compromettre la stabilité ou la sécurité de l'application VBA dans des environnements critiques.</p>	<p>Peut diriger des projets de refonte de code pour améliorer la maintenabilité, la performance, et la sécurité.</p>
	<p>4.3 Optimiser l'environnement de développement</p> <p>Configurer l'IDE VBA en personnalisant l'éditeur, en gérant les références de projet, en utilisant des modèles de code et en mettant en place un environnement structuré, afin de maximiser l'efficacité du développement et faciliter la gestion des projets VBA.</p>	<p>Personnalise l'IDE pour accélérer le développement et le débogage Implémente des scripts et des outils pour automatiser des tâches de développement répétitives.</p>	<p>Intègre des systèmes de contrôle de version et des outils de collaboration pour les équipes de développement en mettant en place des processus et des flux de travail qui permettent une gestion efficace du code source, une collaboration transparente entre les membres de l'équipe, et une coordination cohérente des modifications apportées aux applications VBA, assurant ainsi la cohésion et la qualité du développement logiciel dans un environnement de travail collaboratif.</p>	<p>Crée et maintient un environnement de développement optimisé pour des projets de grande envergure, incluant des formations et des guides pour les développeurs.</p>