

Mis à jour le 24/11/2025

S'inscrire

Formation TestStand Niveau 2

3 jours (21 heures)

Présentation

Maîtrisez l'automatisation de test avancée avec TestStand Niveau 2 : personnalisez vos modèles, orchestrez des séquences parallèles et fiabilisez vos bancs. Accélérez vos campagnes, réduisez les cycles et augmentez la traçabilité grâce à des artefacts réutilisables et normés. Parfait pour industrialiser du prototype à la production, avec intégration LabVIEW/.NET et reporting base de données.

Cette formation consolide votre compréhension de l'architecture TestStand : process models (Batch/Parallel), plug-ins de reporting et base de données, gestion des résultats, propriétés personnalisées et persistance station. Vous apprendrez à créer des types d'étapes, structurer des données, paramétrer les post actions et piloter des séquences robustes multi-UUT.

Notre approche est résolument pratique : ateliers guidés, démos pas-à-pas et refactoring d'une séquence existante vers un modèle réutilisable. Livrables fournis : projets d'exemple complets, templates de step types, snippets de synchronisation (queues, notifiers, locks), checklists de revue, guide de déploiement et fichiers de configuration pour TestStand Deployment Utility.

Objectifs

- Créer des étapes de test personnalisées
- Utiliser l'API TestStand pour automatiser et étendre les fonctionnalités
- Concevoir des interfaces utilisateur adaptées aux opérateurs
- Adapter les modèles de processus à des besoins spécifiques
- Optimiser le déploiement et la maintenance des systèmes de test

Public visé

- Ingénieurs tests et validation
- Développeurs LabVIEW/CVI/.NET impliqués en test
- Architectes de bancs de test

- Responsables techniques QA

Pré-requis

- Pratique de base de TestStand (séquences, steps, variables).
- Notions de LabVIEW ou C/C++ ou C# pour les code modules.
- Compréhension du multithreading et de la synchronisation.
- Connaissances Git et gestion de versions.
- Bases SQL et schéma de logging TestStand.

Pré-requis techniques

- PC 64 bits avec 16 Go de RAM (4 cœurs recommandés).
- Windows 10/11 avec droits administrateur.
- NI TestStand installé, plus LabVIEW et/ou Visual Studio selon besoins.
- Drivers NI nécessaires (ex. NI-DAQmx) et accès à un device simulé ou réel.
- Outils: Git, éditeur de code, base SQL locale (ex. SQL Server Express).

Programme de formation TestStand Niveau 2

[Jour 1 - Matin]

Architecture avancée et rappels essentiels

- Rôle du Process Model et principaux callbacks (PreUUT, PostUUT, TestReport)
- Exécutions, threads et contextes : navigation via PropertyObject
- Expressions avancées et lookup strings (Locals, File/Station Globals)
- Gestion des résultats : ResultList, on-the-fly logging, nettoyage
- Structuration d'un projet : séquences, types, modèles et répertoires
- Atelier pratique : Étendre PreUUT pour lire/valider un numéro de série.

[Jour 1 - Après-midi]

Personnaliser le Process Model et le Result Processing

- Choisir entre Sequential, Batch, Parallel et impacts
- Surcharger des callbacks (PreUUT, PostResultListEntry, ReportOptions)
- Configurer les plugins de Result Processing (rapport, base de données)
- Filtrer/étendre les résultats : Additional Results, Include/Exclude par type de pas
- Personnaliser le format de rapport (XML/HTML/JSON) et les métadonnées UUT
- Atelier pratique : Créer un rapport JSON enrichi avec entête custom.

Exécutions multi-UUT : Batch et Parallel

- Configurer les Test Sockets et options d'exécution
- Synchronisation avec Lock, Rendezvous, Queue, Notifier
- Partage de données entre sockets (File/Station Globals) et règles d'accès
- Déboguer des exécutions concurrentes : traces, breakpoints, vues d'exécution
- Atelier pratique : Séquence parallèle 2 UUT avec ressource partagée protégée.

[Jour 2 - Matin]

Création de Step Types et réutilisabilité

- Anatomie d'un Step Type : propriétés, templates, Substeps (Pre/Post)
- Paramètres, valeurs par défaut, expressions et icône/menu/aide
- Contrôle de version et distribution de type palettes
- Journalisation : Additional Results et mapping vers ResultList
- Bonnes pratiques pour éviter les effets de bord et faciliter la maintenance
- Atelier pratique : Créer un Step Type .NET appelant une API d'instrument.

[Jour 2 - Après-midi]

Intégration avec LabVIEW / .NET / Python

- Adapters : configuration, chargement de VIs, assemblies et scripts
- Passage de données complexes (clusters/objets) et conversions
- Gestion des erreurs et mapping vers TS Status/Error
- Appels asynchrones et UI Messages pour retour d'état
- Atelier pratique : Appeler un script Python et parser ses résultats.

Interfaces opérateur personnalisées (UI)

- ApplicationManager, ExecutionViewManager et cycle de vie
- Événements UIMessages et commandes (CommandKind)
- Chargement de séquences, gestion des exécutions et des résultats
- Branding, droits utilisateurs et verrouillage contextuel des commandes
- Atelier pratique : Ajouter "Re-run failed" dans une UI .NET existante.

[Jour 3 - Matin]

Base de données et traçabilité

- Configurer le Database Logging et choisir un schéma

- Mapping des champs, statements personnalisés et triggers
- Optimiser taille et performance (purge, index, on-the-fly)
- Requêtes de validation et export de données pour analyse
- Atelier pratique : Créer un schéma custom et logger une mesure clé.

[Jour 3 - Après-midi]

Déploiement et gestion de configuration

- TestStand Deployment Utility : workspaces, builds, inclusions
- Prérequis : moteurs d'adapter, drivers, merge modules
- Fichiers de configuration (Station/Model/Type) et ordre de recherche
- Versioning avec Git et post-build smoke tests
- Atelier pratique : Générer un installeur et valider sur poste vierge.

Diagnostic, performance et bonnes pratiques

- Outils de debug avancés : Trace, Watch, Remote Debugging
- Mesurer le temps d'exécution et profiler les pas critiques
- Structurer les projets : naming, dossiers, dépendances, types
- Sécurité et comptes utilisateurs : signatures, mots de passe
- Atelier pratique : Audit d'un projet et plan d'amélioration ciblé.

Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

Positionnement à l'entrée en formation

Le positionnement à l'entrée en formation respecte les critères qualité Qualiopi. Dès son inscription définitive, l'apprenant reçoit un questionnaire d'auto-évaluation nous permettant d'apprécier son niveau estimé sur différents types de technologies, ses attentes et objectifs personnels quant à la formation à venir, dans les limites imposées par le format sélectionné. Ce questionnaire nous permet également d'anticiper certaines difficultés de connexion ou de sécurité interne en entreprise (intraentreprise ou classe virtuelle) qui pourraient être problématiques pour le suivi et le bon déroulement de la session de formation.

Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des

séances de réflexions, et de travail en groupe.

Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.