Job and Task Analysis for:
Certified TestStand Developer

Certification ID	Certification Title	Job Description:
CTD	Certified TestStand Developer	Given a set of conditions, the Certified TestStand Developer will be able to develop testing solutions using TestStand software and either LabVIEW or LabWindows/CVI

Task ID	Task	Objective ID	Objective
CTD-IND-01	Navigate and exploit the TestStand environment		
		CTD-IND-01-01	Define the role of a test executive, including the inherent operations ▪ Operator interface ▪ User management ▪ Unit under test (UUT) tracking ▪ Test flow control ▪ Collecting results ▪ Test reporting
		CTD-IND-01-02	List and describe the main components of TestStand ▪ Sequence editor ▪ Operator interface ▪ TestStand engine ▪ Module adapters ▪ Process Model

		CTD-IND-01-03	Define the uses and access priority of the \User and \NI folders in TestStand
		CTD-IND-01-04	Compare and contrast the following TestStand components ▪ Sequence Editor ▪ Operator Interface
		CTD-IND-01-05	Explain the relationship between the following TestStand components ▪ Sequence editor ▪ Operator interface ▪ TestStand engine ▪ Module adapters ▪ Process Model
		CTD-IND-01-06	Run a sequence file from the Sequence Editor
		CTD-IND-01-07	Run a sequence file from a default TestStand operator interface
		CTD-IND-01-08	Define the following terms: ▪ Sequence ▪ Sequence step ▪ Step group ▪ Sequence file
		CTD-IND-01-09	Describe the debugging tools available in TestStand
		CTD-IND-01-10	Define a process model

		CTD-IND-01-11	List and describe the three available process models and identify the default model: ▪ SequentialModel.seq ▪ BatchModel.seq ▪ ParallelModel.seq
		CTD-IND-01-12	List and describe the two modes of operation for each process model: ▪ Test UUTs ▪ Single Pass
		CTD-IND-01-13	Define the following TestStand terms: ▪ Workspaces ▪ Projects ▪ Source control ▪ Differ
		CTD-IND-01-14	Describe the following terms associated with sequence files: ▪ Subsequence ▪ Setup Step Group ▪ Main Step Group ▪ Cleanup Step Group ▪ Execution Flow
		CTD-IND-01-15	Explain the use of the Watch Window
CTD-IND-02	Create test sequences in TestStand		

		CTD-IND-02-01	Describe the process for creating a test sequence ▪ Specify the adapter ▪ Select the step type ▪ Specify the test module ▪ Configure the step properties
		CTD-IND-02-02	List and describe the function of the following step properties: ▪ Precondition ▪ Run option ▪ Post action ▪ Loop option ▪ Synchronization ▪ Expression
		CTD-IND-02-03	Create steps in the Sequence Editor
		CTD-IND-02-04	Configure a Sequence with loop options
		CTD-IND-02-05	Create a Sequence with different step types
CTD-IND-03	Create and incorporate custom step types in a TestStand sequence		

		CTD-IND-03-01	Describe and compare the standard step types included in TestStand

		CTD-IND-03-02	Describe the method used to customize a step type
			▪ Add/Remove step properties
			▪ Add/Remove step sub-steps
			▪ Configure step options
			  ▪ Version
			  ▪ Struct passing
			  ▪ Menu
			  ▪ Default synchronization
			  ▪ Default run options
			  ▪ Default expressions
			  ▪ Disable properties
			  ▪ Code templates

CTD-IND-04	Incorporate Parameters, Variables, and Expressions in TestStand tests		
		CTD-IND-04-01	List and describe the methods for passing data within TestStand ▪ Parameters ▪ Variables ▪ Expressions
		CTD-IND-04-02	List and describe the variable types in TestStand ▪ Parameters ▪ Local variables ▪ Sequence file global variables ▪ Station globals

		CTD-IND-04-03	For a given situation, identify the correct variable type to be used ▪ Parameters ▪ Local variables ▪ Sequence file global variables ▪ Station globals ▪ File Globals
		CTD-IND-04-04	List the steps for creating variables ▪ Local ▪ File global ▪ Station global ▪ Temporary global
		CTD-IND-04-05	Describe the RunState property
		CTD-IND-04-06	Explain the function of the RunState subproperties ▪ Engine ▪ Root ▪ Main ▪ ThisContext ▪ Caller ▪ Report ▪ Execution ▪ Sequence ▪ Thread

		CTD-IND-04-07	Describe the function of the Expression Browser
		CTD-IND-04-08	Describe, explain the uses of, and differentiate between the following objects ▪ SequenceContext ▪ ThisContext
		CTD-IND-04-09	Describe and explain the uses of the RunState property
		CTD-IND-04-10	Add a local variable to a sequence
		CTD-IND-04-11	Create a file global variable
		CTD-IND-04-12	Create a station global variable
		CTD-IND-04-13	Create an expression with the Expression Browser
		CTD-IND-04-14	In a Test Sequence, modify the limit properties of a Numeric Limit Step
		CTD-IND-04-15	Create a test that dynamically selects certain sequences for execution
		CTD-IND-04-16	Change a step property using an expression

CTD-IND-05	Create a Test Module in an External Environment		
		CTD-IND-05-01	Explain the role of the Module Adapters in the TestStand architecture
		CTD-IND-05-02	Describe ActiveX Automation as it relates to TestStand
		CTD-IND-05-03	List and describe the three ways to transfer data in TestStand ▪ Predefined structures ▪ Arbitrary numbers of parameters ▪ ActiveX Automation API
		CTD-IND-05-04	Describe the advantages of using TestStand Code Templates
		CTD-IND-05-05	Create a pass/fail test in either LabVIEW or LabWindows/CVI and pass the results to a TestStand step
		CTD-IND-05-06	Create a Numeric Limit Test in either LabVIEW or LabWindows/CVI
		CTD-IND-05-07	Create a Numeric Limit Test for the DLL Flexible Prototype Adapter
		CTD-IND-05-08	Debug a test using either LabVIEW or LabWindows/CVI
		CTD-IND-05-09	Incorporate the ActiveX API in a Code Module using either LabVIEW or LabWindows/CVI

		CTD-IND-05-10	Incorporate the ActiveX API in a Code Module using the DLL Flexible Prototype Adapter
		CTD-IND-05-11	Create a Simple Test using either the LabVIEW or LabWindows/CVI Standard Prototype Adapter
		CTD-IND-05-12	Pass parameters from TestStand to either LabVIEW or LabWindows/CVI using the TestStand API
		CTD-IND-05-13	Pass data between TestStand and a Test Module using the DLL Flexible Prototype Adapter
CTD-IND-06	Configure and control User Management in TestStand		
		CTD-IND-06-01	Create a new user in TestStand
		CTD-IND-06-02	List and describe the user profiles and privileges in TestStand
		CTD-IND-06-03	Describe the process for the following new user actions in TestStand: ▪ Create a new user ▪ Set login access ▪ Set privileges ▪
		CTD-IND-06-04	Configure TestStand to automatically login the Windows system user

CTD-IND-07	Collect Results and Generate Reports in TestStand		
		CTD-IND-07-01	Explain the composition and use of the Locals.ResultList
		CTD-IND-07-02	List the data that is stored in Locals.ResultList
		CTD-IND-07-03	Describe the TestReport sequence
		CTD-IND-07-04	Describe the Report Configuration options: ▪ Report format ▪ Inclusion of certain result elements ▪ Insertion of arrays ▪ Result filtering ▪ Numeric formatting ▪ Report color schemes ▪ Report file pathnames ▪ Separate reports per UUT ▪ Create Unique Report for each Device
		CTD-IND-07-05	Given a set of conditions and desired results, generate a report in TestStand
CTD-IND-08	Integrate database functionality with TestStand		

		CTD-IND-08-01	Describe the following database interface technologies: ▪ ActiveX Data Object ▪ Object Linking and Embedding – Database ▪ Open Database Connectivity ▪ Data Link
		CTD-IND-08-02	Describe the following database configuration options: ▪ Logging options ▪ Data link ▪ Schemas ▪ Statements ▪ Parameters
		CTD-IND-08-03	Describe the process for creating default database tables
		CTD-IND-08-04	Explain the use of the database viewer, including: ▪ Data link window ▪ Execute SQL window ▪ Data view window
		CTD-IND-08-05	List and explain the actions of each of the following database step types: ▪ Open database ▪ Open SQL statement ▪ Data operation statement ▪ Close SQL statement ▪ Close database

		CTD-IND-08-06	Create a Microsoft Access Database session using built-in TestStand database step types
CTD-IND-09	Perform TestStand Operations from the Operator Interface		
		CTD-IND-09-01	List the four operator interfaces included with TestStand ▪ LabVIEW ▪ LabWindows/CVI ▪ Visual Basic ▪ Delphi
		CTD-IND-09-02	Describe the similarities and differences between the Sequence Editor and the default TestStand Operator Interface
		CTD-IND-09-03	List and explain the purpose of the operator interface functions ▪ Login with privilege control ▪ Sequence file display ▪ Execution display ▪ Breakpoints / single-stepping ▪ Interactive Execution
		CTD-IND-09-04	Explain the purpose of a UIMessage

		CTD-IND-09-05	Describe the Import/Export Properties tool and its associated functions: ▪ Source/Destination ▪ Skip Rows That Begin With ▪ First Row of Data Specifies Step Property for Each Column ▪ Import ▪ Export
		CTD-IND-09-06	Explain the purpose of the Property Loader Step
		CTD-IND-09-07	Given a set of conditions and desired results, edit the properties and variables for a sequence
CTD-IND-10	Distribute TestStand and the Operator Interface		
		CTD-IND-10-01	Describe the TestStand run-time engine, and its associated deployment files ▪ Setup TSEngine.exe ▪ TSEngine.cab
		CTD-IND-10-02	Explain the method of distribution for the Operator Interface • LabVIEW • CVI
		CTD-IND-10-03	Explain the method of distribution for Sequence Files

		CTD-IND-10-04	Explain the method of distribution for Code Modules associated with: • LabVIEW • CVI
CTD-CST-01	Utilize the TestStand API		
		CTD-CST-01-01	Describe the TestStand API and list typical clients ▪ Operator Interfaces ▪ Sequence Editor ▪ Process Model ▪ Test Code Modules
		CTD-CST-01-02	Define and explain the use of TestStand objects ▪ Sequence File ▪ Sequence ▪ Step ▪ Execution ▪ Sequence Context ▪ Property object ▪ Report ▪ UI Messages ▪ Thread ▪ Engine
		CTD-CST-01-03	Identify and define API classes
		CTD-CST-01-04	Describe the property_object and its relationship to other TestStand objects

		CTD-CST-01-05	Describe the application of Property Object methods to specific TestStand objects: Methods/Properties of Propety Object: ▪ Name ▪ Exists ▪ GetPropertyObject ▪ SetValBoolean ▪ SetValNumber ▪ SetValString ▪ GetValBoolean ▪ GetValNumber ▪ GetValString ▪ GetNumElements ▪ GetNumSubProperties Objects ▪ Step ▪ Sequence ▪ Sequence File ▪ Property Object
		CTD-CST-01-06	Describe the concept of Containment as it relates to the API Structure
		CTD-CST-01-07	Define the following terms: ▪ Properties ▪ PropertyObjects ▪ Methods

		CTD-CST-01-08	Explain the purpose of the TestStand Adapters ▪ ActiveX Automation Adapter ▪ LabVIEW Standard Prototype Adapter ▪ C/CVI Standard Prototype Adapter ▪ DLL Flexible Prototype Adapter ▪ HT Basic Adapter ▪ None Adapter ▪ Sequence Adapter
CTD-CST-02	Manage Users in TestStand		
		CTD-CST-02-01	List and explain the purpose of the following high-level categories of configurable properties associated with TestStand users ▪ LoginName ▪ FullName ▪ Password ▪ Privileges
		CTD-CST-02-02	List the steps necessary to delete a TestStand user • Through the user manager • Programmatically
		CTD-CST-02-03	Explain the aspects of configuring user information in TestStand
CTD-CST-03	Integrate Multithreading and Multi-UUT Testing in TestStand		
		CTD-CST-03-01	Explain the benefits of using Multithreading in a Test Executive

		CTD-CST-03-02	Describe the following process models: ▪ Parallel Model ▪ Batch Model
		CTD-CST-03-03	Describe the use of the Synchronization Tab for thread accessibility
		CTD-CST-03-04	Describe the use of the Sequence Call Option for multi-threading purposes
		CTD-CST-03-05	Describe the use of the Synchronization Step Types for synchronizing multiple threads
		CTD-CST-03-06	Explain the concept of remote execution as it relates to TestStand
		CTD-CST-03-07	Describe the process for implementing a remote sequence and the purpose of the Rengine Remote Server
CTD-CST-04	Modify Process Models		
		CTD-CST-04-01	List the customizable aspects of TestStand: ▪ Process models ▪ Result collection ▪ Report generation ▪ Step types ▪ Data types ▪ Operator interfaces ▪ Database logging
		CTD-CST-04-02	Define the purpose of a TestStand Process Model

		CTD-CST-04-03	List common operations of the default TS Process Models: ▪ Identify the UUT ▪ Log results ▪ Notify the operator of pass/fail status ▪ Generate a report
		CTD-CST-04-04	Differentiate between the Sequential Process Model, Parallel Process Model, and the Batch Process Model
		CTD-CST-04-05	List and describe the two methods for modifying a process model: ▪ Overriding Callbacks ▪ Directly Modifying Callbacks
		CTD-CST-04-06	Describe the function and use of Overriding a Callback
		CTD-CST-04-07	Differentiate between a model callback and an engine callback
		CTD-CST-04-08	Define the following sequence types and describe the differences in their use ▪ Execution entry point ▪ Configuration entry point ▪ Callback sequence
CTD-CST-05	Incorporate TestStand Types in test operations		
		CTD-CST-05-01	Define a TestStand Type and the two categories of Types ▪ Step Type ▪ Data Type

		CTD-CST-05-02	Describe the two groups of TestStand Data Types: ▪ Standard Types ▪ Custom Types
		CTD-CST-05-03	List and describe the features of a custom step type that can be modified: ▪ Step properties and values ▪ Editing Dialog ▪ Run-time behavior ▪ Code template ▪ Step sub-steps
		CTD-CST-05-04	Explain the purpose of combining step types
		CTD-CST-05-05	Given a set of conditions and desired results, create a custom step type
CTD-CST-06	Modify a TestStand Operator Interface		
		CTD-CST-06-01	List and describe the three methods for handling UIMessages ▪ Polling Loop ▪ Callback ▪ ActiveX Control Event
		CTD-CST-06-02	Describe the process for Posting a UIMessage
		CTD-CST-06-03	Given a set of conditions and requirements, develop an Operator Interface using either LabVIEW or LabWindows/CVI

CTD-CST-07	Modify Result Collection and report generation in TestStand		
		CTD-CST-07-01	Describe the process by which TestStand collects results Resultlist Result.Container
		CTD-CST-07-02	List and describe the four methods for modifying the Result Collection ▪ Add test to Step.Result.ReportText ▪ AddExtraResult method of the TestStand API ▪ Insert subproperties in Local.ResultList ▪ Add subproperties to Step.Result
		CTD-CST-07-03	Given a set of conditions and desired results, modify a Result Collection in TestStand
		CTD-CST-07-04	List and describe the four methods of customizing Report Generation ▪ Configure Report Options dialog ▪ Use automatic reporting flags ▪ Override test report callbacks ▪ Directly modify test report callbacks and/or report generation files
		CTD-CST-07-05	Given a set of conditions and desired results, modify Report Generations options in TestStand
CTD-CST-08	Create customized database logs in TestStand		

		CTD-CST-08-01	List the steps required to log additional data to a database ▪ Identify additional data to be logged ▪ Pass the information to the Log to Database Sequence ▪ Determines how and where the additional data should be stored in the database ▪ Modify database schema and tables to enable the additional data to be logged
		CTD-CST-08-02	Describe the use and differences between the TestStand database step types
		CTD-CST-08-03	Given a set of conditions and desired results, modify the default Generic Recordset schema to log additional data to a database