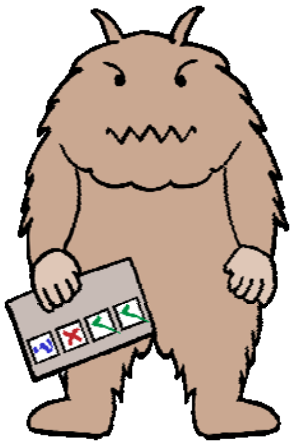
TAME

Test Authoring Made Easy

2-day classroom training • 3-day virtual training



TAME the testing beast!

Learn a systematic procedure for constructing software test cases.

There never seems to be enough time (or money or people) to adequately test systems. Rapid cycle times brought on by the adoption of agile approaches only seem to make things worse. Even with automated systems for *running* tests, you're still left with the time-consuming problem of designing many test cases.

This class will introduce you to a simple yet powerful technique for modeling functional requirements and a tool-supported technique that uses those models to generate effective test cases. This approach can be used for unit testing, user testing, and more. Tests can be developed from early specifications (use cases and user stories) and in response to bug reports.

You'll see how to easily produce a good set of test cases – exhaustive enough to give your software a good workout, but intelligent enough not to waste time and effort.

How to TAME

TAME is Test Authoring Made Easy

- Fill out a simple table and TAME produces dozens of useful test cases.
- Create normal, alternate, and exception scenarios from one simple spec.
- Enhance your software product, then update your test base in minutes.
- Easily maintain a consistent, reliable suite of regression tests.

Here's how it works…

- First, define your different testable functions. These can be business processes, Java functions, use cases, UI screens—any kind of unit with distinct inputs and results.
- Now take one of those functions. Identify its inputs and define different choices of values that will produce different outcomes. Consider normal values as well as those that produce errors.
- List those different results across the top. Then fill in the table by choosing combinations of inputs that lead to different results. It's as simple as "X marks the spot."
- Sometimes the same inputs can lead to different results depending upon certain environment conditions. So also identify any necessary preconditions, such as object states and user roles.
- Specify actual values for each input and the conditions that lead to those values.
- Finally choose or customize document and code templates.
- Now TAME can generate a step by step protocol for manual testing plus code for automated tests.

Course Outline

Start with techniques for organizing systems into processes and activities. Learn how to define the behavior of these processes and activities in process models, use cases, and navigation maps. See how to define distinct scenarios, to partition inputs into different characteristic values, and to relate those scenarios and inputs to expected results. Finally, construct executable test cases from those scenarios, inputs, and results.

Modeling Techniques (Review)

- Structuring and Organizing Systems
- Process Models
- Activity Flow Models
- Data Views for Messages, Displays, and Forms
- Consolidated Information Models
- Object Lifecycles
- State Models

Model Down, Build Up

- Analysis Approaches for Different Problems
- Typical Software Architectures

Testing UI Pages and Simple Functions

- Definition of a Basic Function:
  input – transform – output
- Identifying Inputs
- Partitioning Inputs into Characteristic Values
- Forming Simple Combinations
- Identifying Results
- Linking Inputs to Results

Applying Preconditions

- Preconditions from Process Models
- Preconditions from Information Models
- Linking Inputs and Preconditions to Results
- Limiting Combinations

Defining Values and Test Protocols

- Simple Value Mappings
- Complex Value Mappings
- Protocol Templates and Actions
- Conditional Actions
- Validation ("check that") Actions

Linking Functions

- Results Become Preconditions
- Following the Navigation Map
- Coverage Options: Node, Link, and Path
- Complete Tests for a Single Activity

Testing Process Models

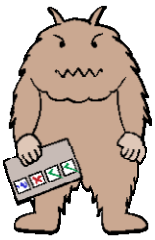
- Linking Activities by Messages
- Decisions by Actors

Optional: Ranorex Integration

Ranorex offers easy-to-use tools for developing and managing test automation projects on a wide variety of desktop, web, and mobile platforms. Use TAME to design the test cases, then implement automated executable tests with Ranorex.

Registration Information

For pricing and registration contact sales@tametest.com or phone (415) 931-3132.



TAME
Test Authoring Made Easy

from Model Compilers