



2012 Guide for Testing Your Software

Security and Software Assessment Services (SSAS)

Usability Testing Sections

- EPRI Corporate Requirements
- Installation / Un-Installation
- Software Documentation
- Test Cases (or Tutorial)
- Graphical User Interface
- Stress Testing
- Security Vulnerability Testing (Web Applications)

EPRI Corporate Requirements

Software Development Requirements

You Are Here: [Home](#) > [Software Requirements](#) > [EPRI Corporate](#)

Development Process:

1. [Concept Development](#)
2. [Request for Proposal](#)
3. [Contract Package](#)
4. [Implementation](#)
5. [Preproduction Test](#)
6. [Final Acceptance Test](#)
7. [Support & Maintenance](#)

Software Requirements:

- [EPRI Corporate](#)
- [Installation/Un-Installation](#)
- [Software Documentation](#)
- [Test Cases \(or Tutorial\)](#)
- [Graphical User Interface](#)
- [Spreadsheet Application](#)
- [Web Application](#)

[Software Type Matrix](#)

EPRI Corporate

- [Export Control](#)
- [Internationalization](#)
- Localization (*Optional*): Process of adapting internationalized software for a specific region or language by adding locale-specific components and translating text
- [Operating Systems Supported](#)
- [Software Encryption](#)
- [Software Integration in Enterprise Systems](#)
- [Software Security Policy](#)
- System response time meets industry performance standards
- [Third Party and Open Source Software](#)

Note: All EPRI Corporate requirements are tested on your software by SSAS.

Installation / Un-Installation

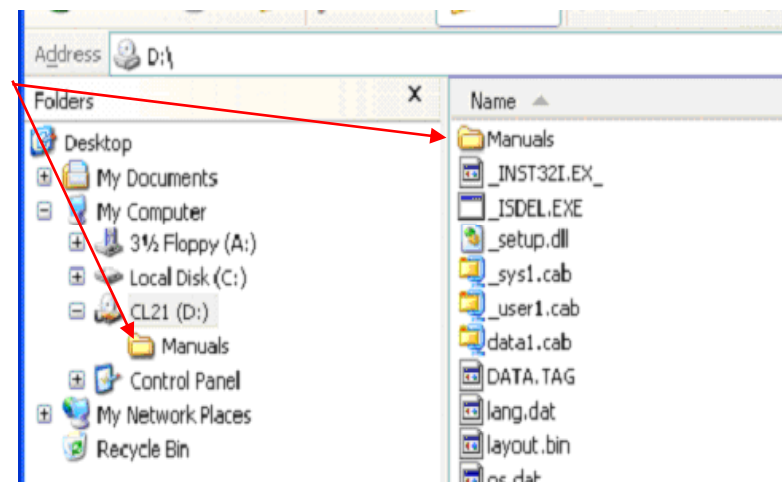
1. Run a Virus Scan



2. Verify:

- User Manual
- Instructions for installation

3. If applicable, provide network installation instructions



4. Default setting installation

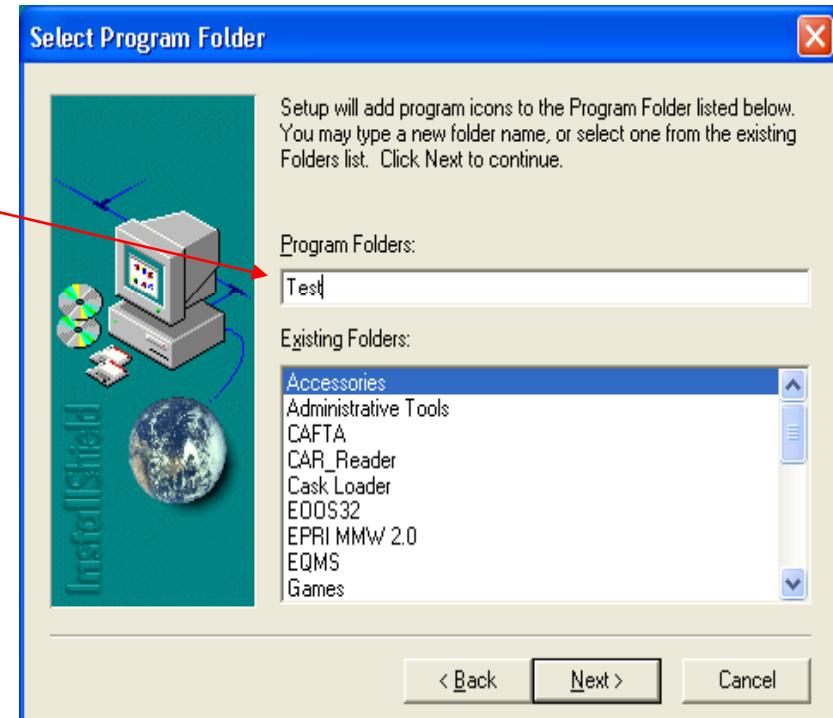
Installation / Un-Installation (cont'd)

5. Uninstall, then reinstall in non-default directory/drive
6. Click on *Cancel* button(s) during installation process
7. If serial numbers or security keys are required, enter invalid entries to make sure the security works



Installation / Un-Installation (cont'd)

8. Change the *Program Folder* where the shortcut in the *Windows Start* menu is located.
9. Applications that do not require installation, such as Spreadsheets and Web Applications, still require documentation.

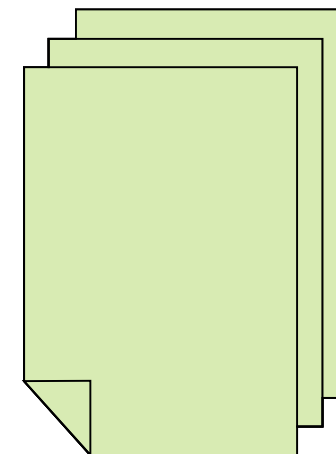


Software Documentation

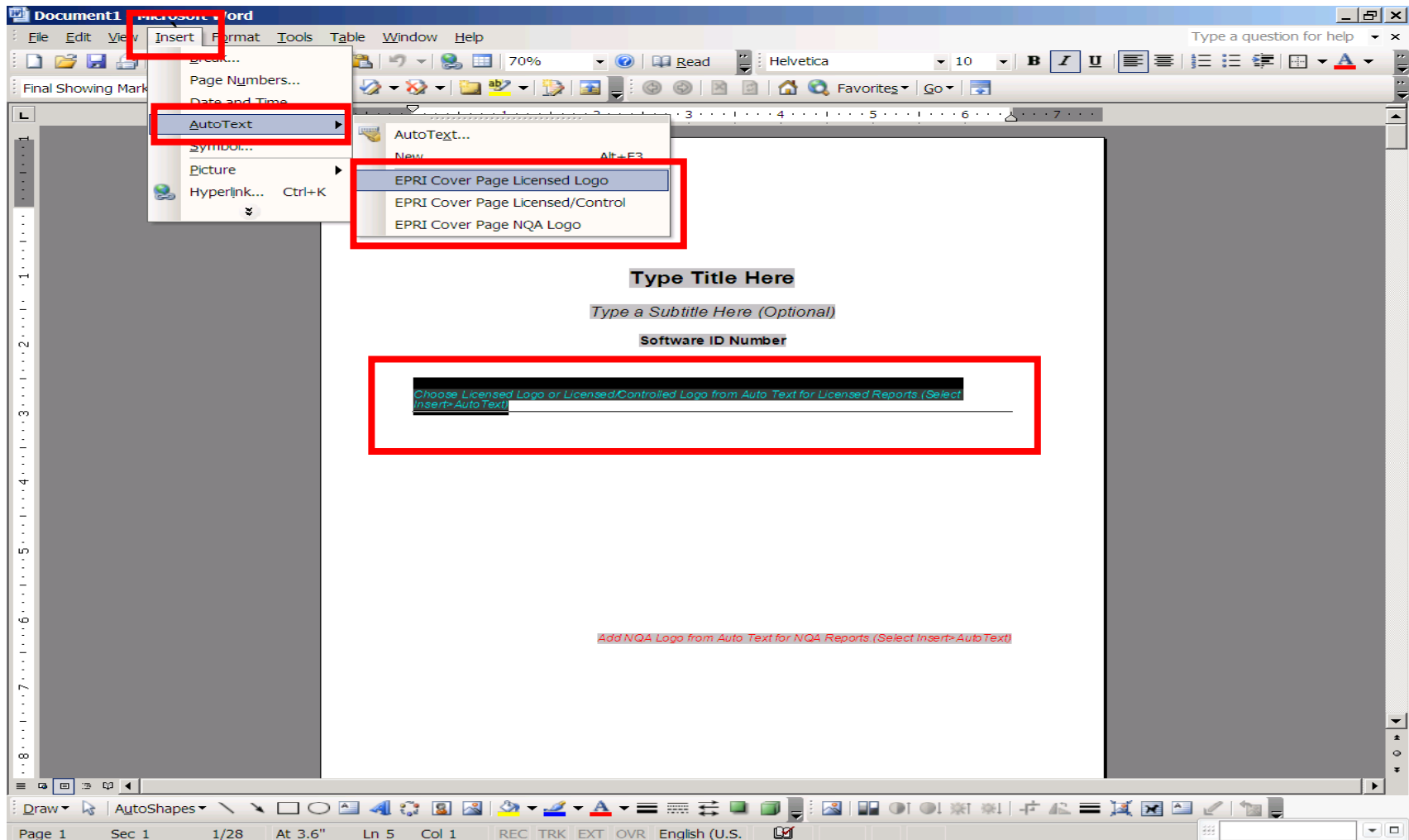
1. Check that EPRI Technical Publications User Manual template was used (or followed)

Note: This ensures title page, disclaimer page, contacts page, copyright and ordering information are all current and that EPRI style guides are used

2. Check headers and footers
3. Check for system requirements:
 - a. Hardware and Software specifications
 - b. Permissions such as Administrator rights
4. Check application feature descriptions
5. Check spelling and grammar



Software Documentation (cont.)

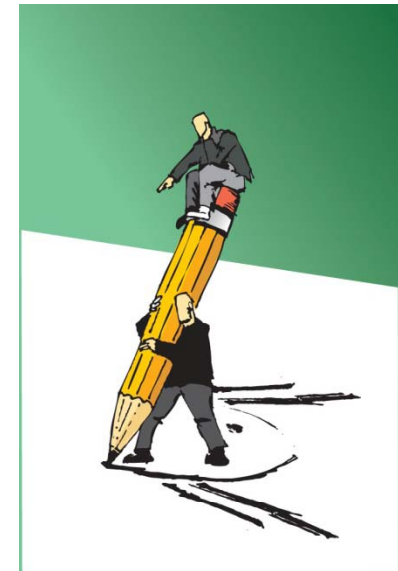


Test Cases (or Tutorial)

Reminder: Three solved example problems (or one tutorial) are required

- Run solved example problems (or tutorial) to make sure all inputs and results (i.e., calculations, graphs, screen captures, etc.) in the application match exactly with the inputs and results in the user documentation

Note: If any inputs or results do not match, the software can not be approved to send to customers



Graphical User Interface

1. Check for the Preproduction Splash screen (if preproduction stage)
2. Windows fit in the main application screen and nothing is cut-off if windows are resized
3. Make sure all data/information is accessible
4. Internationalization, check multiple regions
5. Change appearance settings
6. Controls on pages must respond properly to Tab order and hot-keys (alt-keys)
7. Check online Help feature, including buttons to open the Help feature



Stress Testing

1. Range checking – Look for input fields and enter invalid values
2. Make sure that numeric-only fields accept only numeric values
3. Follow the solved example problems, but then skip a step or do them in a different sequence



Stress Testing (cont'd)

4. Check print feature
5. Try different login combinations
6. Check error messages for clarity. Error messages should appear when the error occurs.
7. Check for spelling within the application




The login information is invalid. Please check and try again.

Login:	<input type="text"/>
Password:	<input type="password"/>



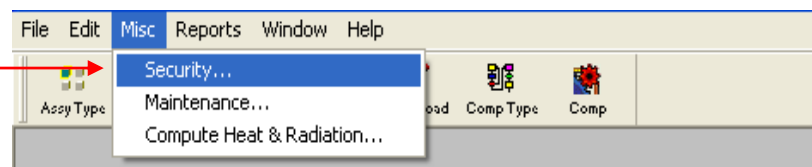
Stress Testing (cont'd)

8. For databases:
 - a. ensure all connections through the application are valid when accessing data
 - b. ensure single quotes and double quotes are tested to verify they do not corrupt the database
 - c. add duplicate records 
 - d. delete all records to make sure it does not crash the application
9. Modify data files (such as adding an extra comma) to make sure the application gives a correct error message

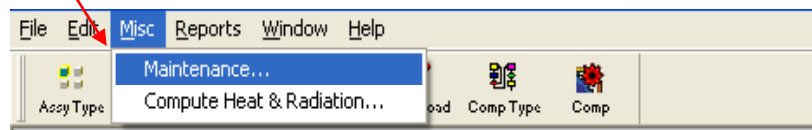
Stress Testing (cont'd)

10. For application administrative features, make sure only administrators of the application may access those features
11. Check for compatibility with Microsoft Office applications if applicable (such as copy and paste features)
12. Click all buttons to make sure they work
13. Check save feature (does not overwrite existing file without permission, saves to correct directory, creates correct extension, etc.)

With administrative feature



Without administrative feature



Stress Testing (cont'd)

14. Check open file feature (correct file extensions, choosing incorrect file type brings up error message, etc.)



15. If there are graphs, check graph features and settings
16. Check options/settings not covered in the sample problems.
17. Check to make sure international units are converted correctly

The International Standard

date notation

DD-MM-YYYY

United States Standard

date notation

MM-DD-YYYY

Stress Testing (cont'd)

18. Maximize, minimize, and resize windows to make sure the application responds correctly.
19. Check keyboard shortcuts
20. Check all menu items, including the pop-up menus that come up when the user right-mouse clicks an item
21. If there are hardware/software keys, check to see if the application responds when executed with the key(s), then without the key(s)



X



C



P

Security Vulnerability Testing

- OWASP Top Ten Web Application Vulnerabilities
 - http://www.owasp.org/index.php/OWASP_Top_Ten_Project



10
TOP

- 1: Injection
- 2: Cross-Site Scripting (XSS)
- 3: Broken Authentication and Session Management
- 4: Insecure Direct Object References
- 5: Cross-Site Request Forgery (CSRF)
- 6: Security Misconfiguration
- 7: Insecure Cryptographic Storage
- 8: Failure to Restrict URL Access
- 9: Insufficient Transport Layer Protection
- 10: Unvalidated Redirects and Forwards

Security Vulnerability Testing

- 2 examples of vulnerabilities SSAS will test for:
 - Structured Query Language (SQL) Injection
 - Cross-Site Scripting
- The developer is expected to address security vulnerabilities when developing an application



Security Vulnerability Testing (cont.)

- SQL Injection – Injection of a SQL Query through input data, such as a querystring or form
- Examples:
 - In the querystring, enter a SQL Statement, such as " ' ; Delete from users --' ", into a querystring variable
 - Enter in " ' OR 1=1 " into a form field or querystring variable
- See the following for more information and testing examples:
http://www.owasp.org/index.php/SQL_Injection



Security Vulnerability Testing (cont.)

- Cross-Site Scripting - Harmful scripts are entered into web sites via querystring or form field
- Example:
 - Enter in "<script type='text/javascript'> alert('hello'); </script>" into a form field to check whether the form field is validated
- Allows the user to execute scripts that are harmful
- See the following for more information:
<http://www.owasp.org/index.php/Cross-site-scripting>

Security Vulnerability Testing (cont.)

- Testing tools:
 - OWASP's Web Scarab (Manual)
 - IBM Rational AppScan (Automated)
 - Rapid 7 (Automated)
- Reference:
 - Open Web Application Security Project (OWASP)
http://www.owasp.org/index.php/Main_Page



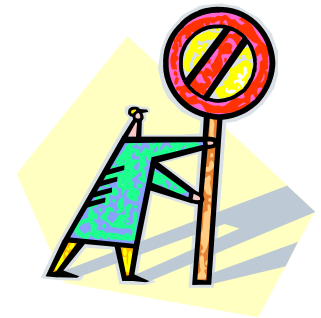
What SSAS Does Not Do

SSAS software usability testing **does not do:**

1. V&V (Verification and Validation) testing
2. test or validate real world data (this should be done by beta testers)
3. exhaustive testing or “white box” (source code) testing

SSAS usability testing will not find all errors and is not intended to

All errors are expected to be found by developers





Together...Shaping the Future of Electricity