

# PS Functional Requirements Checklist

---

This checklist provides a listing of the basic functional requirements of a Performance Support System

## **Administrative Functions**

- ☐ Allows role-based content design and publishing
- ☐ Enables role-based content editing

## **Content Creation Functions**

- ☐ Enforces style guidelines through authoring interface.
- ☐ Provides full-featured, robust, WYSIWIG editing environment
- ☐ Delivers content typing based on PS methodology (e.g. Steps, Details, Additional Resources)
- ☐ Provides image library management
- ☐ Provides inline image editing and hotspotting.
- ☐ Supports creation of graphical workflows
- ☐ Supports creation and manipulation of tables
- ☐ Supports the following style-related requirements:
  - Loose styleguide enforcement (e.g. the system guides users to follow best-practices for creating PS content but allows users to override those best-practices where necessary)
- ☐ Supports all MIME-compliant objects in content, including graphics, video, and other multimedia content.
- ☐ Supports user-defined variables/entities  
*Variables in this case are symbols which are resolved to some value at publication time. Users need to be able to manage (create/view/modify/delete) them with appropriate UIs.*
- ☐ Supports content reuse through content object nesting and across multiple deliverables (e.g. TOCs)
- ☐ Provides a mechanism for annotating content objects.
- ☐ Allows editing by 1 or more editors on a single copy of the content repository.
- ☐ Supports importing (copy/paste) of unstructured content from major formats, including, but not limited to, FrameMaker, Microsoft Word, ASCII Text, RTF, Excel.

- ☐ Supports import of graphics files or copy/paste of graphical images into content objects.

### ***Content Management Functions***

#### **Versioning**

- ☐ Automatically provides check-in/check-out work in progress
- ☐ Supports manual versioning of content objects
- ☐ Restricts changes associated with a version from affecting content associated with any other releases.
- ☐ UI reflects the most recent revision of content or historic versions
- ☐ Selects by default the versions tagged as “current” for the latest release
- ☐ Allows rollback (taking an old revision and making it the current revision)
- ☐ Supports separate status attribute for each object – draft, review, final
- ☐ Supports annotations applied to objects or groups of objects.

#### **Linking**

- ☐ Supports the following types of links:
  - Internal (linking between content objects)
  - Embedded resource (content linked to an embedded document)
  - To External Content (links to content stored outside the repository)
  - From External Content (Links from content stored outside the repository to content within the repository)
  - Mail-to (links generating mail notes to pre-defined recipients)

#### **Content Assembly of Deliverables**

- ☐ Provides a UI for assembling deliverables by dragging and dropping appropriate content objects into a Table of Contents
- ☐ Provides a UI for selecting deliverable types and setting deliverable options

#### **Content Identification Metadata Management**

- ☐ Allows for the configuration of metadata tags and taxonomies

- ☐ Stores standard metadata (author, edit date ,status, etc).
- ☐ Allows for the assignment of hierarchical metadata to content objects. If a leaf node metadata value is assigned to a content object, that object shall behave as if tagged with all of the parents of the leaf node metadata value.
- ☐ Accommodates an arbitrary number of metadata values for each “tag”
- ☐ Provides a graphical UI for the management of metadata on any content object.

#### **Production/Publishing Functions**

- ☐ Delivers HTML (portal view and embedded learning view), SCORM package, Mobile, and PDF targets which are and will be deliverables
- ☐ Provides web-standards driven interface design for each publication channel – enabling custom look-and-feel.
- ☐ Workstation publishing for content review

#### **Content Delivery**

- ☐ Published deliverables (static HTML, SCORM, Mobile, PDF) use standard web technologies for content delivery (IIS/Apache for serving content, IE/Firefox/Safari for rendering)
- ☐ Content provides content sharing (peer-to-peer, author-to-user, application owner-to-user, etc.) mechanism
- ☐ Content provides content object-specific feedback mechanism to content owner.
- ☐ Content can be printed with print-specific stylesheets

#### **Search and Content Access of Deliverables**

- ☐ Searchable full-text index
- ☐ Searchable metadata (keywords)
- ☐ Ranked (ordered) search results

#### **Documentation and Training**

- ☐ Includes complete user documentation
- ☐ Includes complete instructions for installing the applications
- ☐ Includes provisions for training all users in the use of the system