Final Report on the CMS Selection for the Web Central Refresh

Web Technologies Team, ITS Applications
University of Texas at Austin
December 2009

Contents

Report Narrative .................................................................................2
Appendix A – Selection Criteria Matrix .............................................7
Appendix B – Requirements, Rationale, and Fit Criteria .................14
Final Report on the CMS Selection for the Web Central Refresh

Executive Summary
As part of the 2009/2010 Web Central Refresh project, the ITS Web Technologies Team (WTT) developed a process to evaluate and select a Content Management System (CMS) for the UT Austin home page and top-level pages. After developing a set of selection criteria, we identified seven candidate packages to evaluate, which was narrowed down to three after the first round of evaluation—ExpressionEngine, Plone, and Drupal.

After further definition of project requirements, rationales, and fit criteria, the three finalists were more closely evaluated. After the second round of evaluation, the package that was determined to most closely match the desired characteristics of the Web Central Refresh CMS was Drupal.

CMS Selection Project Background
The WTT and the UT Austin Office of Public Affairs are currently engaged in a redesign of Web Central, the university’s Web presence at http://www.utexas.edu. One component of this project is the selection of a CMS to manage the content of the university’s home page and top-level pages. Until this point, content on the home page and top-level pages has been updated through manual HTML updates, with no programmatic ability to track or approve changes.

The scope of this project is purposefully limited to selecting and implementing a CMS for the home page and top-level pages. This is not an effort to select a new "enterprise" CMS for the campus, and the selection process did not evaluate candidate packages on that basis.

Phase 1 - Selection Criteria Matrix
The main task during the first phase of evaluation was the development of a Selection Criteria Matrix to be used for rating the initial seven candidates.

During the development of selection criteria, the following high-level categories were established for judging CMS candidates:

• **Availability**: the product should support a very high-traffic site, both in terms of concurrent contributor connections, as well as high-availability caching that does not depend on CMS throughput.
• **Data exportability**: the product should allow content to be published and re-used in multiple formats.

• **Authentication/authorization integration**: the product should have a flexible and modular authentication and authorization system that can support EID-based signon.

• **Ease of use**: the product should be easy for content creators/editors to learn and use on a daily basis.

• **Workflow**: the product should allow for the creation of custom workflows to manage the editing and publishing of content.

• **Product maturity/active user community**: the product should be mature within the CMS market, preferably with significant penetration in the higher-education sector.

• **Ease of UI customization**: the product should allow for easy customization of the content editing interface, including the integration of a WYSIWYG editor.

• **Web standards and accessibility**: the product should support the production of pages that meet the campus standards for accessibility (Section 508) and open Web standards (XHTML/CSS).

• **Alignment with campus-supported technologies**: the product should align with common campus technologies in which we have a broad base of knowledge and expertise, such as PHP, Python, MySQL, and Oracle.

• **Cost-effectiveness and ease of setup/maintenance**: the product should be free or relatively inexpensive, and be able to be set up and configured in less than one day.

• **Application security**: the product should comply with the campus minimum standards for application development and administration, and the developers should demonstrate a history of quick response to identified vulnerabilities.

---

**Phase 2 - First Pass Evaluation**

The seven candidates evaluated against the Selection Criteria Matrix were:

- DjangoCMS
- Drupal
- ExpressionEngine
- Joomla!
- Oracle Universal Content Management System
- Plone
- WordPress

These packages were evaluated based on a combination of product research, first-hand experience, and interviews with other campus units. Most of the initial seven candidates stacked up favorably when rated against the selection criteria (see...
Appendix A for complete results). Decisions about which packages to review as part of the final phase were made using both objective and subjective data gathered through our research.

At the end of this phase, the following packages were identified as finalists:
- Drupal
- ExpressionEngine
- Plone

**Phase 3 - Requirements Definition and Final Evaluation**
Phase 3 began with the development of a list of requirements for the Web Central CMS, based on the work started in the Selection Criteria Matrix. This document included rationale and fit criteria that could be used to directly evaluate the candidate packages against the requirements (see Appendix B).

In addition to the body of research that was built during Phase 2, we installed all three finalist packages locally, in order to get direct experience with them.

**Plone**
After installing and starting to use all three systems, the evaluation team quickly determined that Plone was not a viable candidate for this project, despite the growing importance of the Python language as a development tool at UT.

Plone’s architecture is substantially more complex than most open-source CMS packages. It is a powerful and flexible system with its own server and object database (Zope), which the WTT has no experience setting up or managing.

Plone’s features, including a highly customizable interface and complex workflows, would work well as an enterprise-level solution in a situation with fewer resource and time constraints. Within the context of the Refresh project, Plone’s extensibility and level of configurability were not justified by the added complexity and training time that would have been required.

**ExpressionEngine**
While meeting most of the project’s high-level requirements through either core functionality or available add-ons, ExpressionEngine was eventually deemed to fall short due to the limited nature of the solutions available. In particular, the options available through the single third-party workflow module that we were able to identify (NSM Publish Plus) were extremely limited in comparison to what was found to be available in Drupal through the combination of different contributed modules.
Similarly, the third-party module identified for "pushing" CMS content (Static Page Caching) as static HTML pages met the documented high-level requirements, but did not compare favorably to the options available in the comparable Drupal module.

**Drupal**

Drupal was found to meet nearly all of the project's documented high-level requirements, either through core functionality or available contributed modules. Two requirements identified early on as critical to the success of this project were configurable workflows and scalability through high-availability caching. We determined that Drupal fulfills both of these requirements using off-the-shelf free and open source modules.

In the specific case of high-availability caching, there is a third-party module called Boost that allows specified pages on a Drupal site to be stored as HTML and/or compressed (gzip) HTML, eliminating the overhead of PHP processing and the database backend as a dependencies for content. For static Web Central content, this will allow us to have the benefits of a CMS without the risk of overloading the system with high traffic.

Since the Drupal project is deeply committed to the open source philosophy, there is a very large community of developers who are continually contributing new functionality back to the Drupal ecosystem either through contributed modules, or patches to the core product itself. Additionally, there are a growing number of companies offering commercial consulting, training, and support services for Drupal.

Based on our research, Drupal may have the widest adoption throughout the higher-education community of any of our initial seven candidate systems, perhaps challenged only by WordPress for its ubiquity and momentum. In addition to many schools (including UT Austin) that have been using Drupal at the departmental level, prominent institutions such as Duke and Rutgers have recently moved their main pages to Drupal.

The WTT already has extensive experience with Drupal, both in terms of the underlying technologies (PHP, MySQL) and the package itself. Since 2008, we have deployed two departmental site redesigns using Drupal, and have consulted with several other departments as they have begun using Drupal for their own sites. We have found it to work well for content contributors who are not technical experts, and most new users have required a minimal amount of training in order to start managing their own content.
Conclusions

Drupal and ExpressionEngine were both identified as systems that met most of the requirements and criteria defined during the selection process. Both systems are already in use on campus, and have proven sufficient for the scale of department or unit-level Web publishing. We believe that Drupal is the best choice for the Web Central CMS due to its scalability, flexibility, and growing popularity, especially within the higher-education community.

A common criticism of Drupal heard during the evaluation process is that it is too complex, offering more options than are necessary for building a relatively simple site. Our experience and evaluation have found that Drupal strikes a balance between ease-of-use and complexity, allowing site administrators to disable or hide almost any feature from users as desired. And its highly extensible architecture and large community practically guarantee that any future requirements can be met by identifying an existing add-on, or in the extreme case, developing one on campus.
Appendix A – Selection Criteria Matrix
<table>
<thead>
<tr>
<th>Requirement</th>
<th>DjangoCMS</th>
<th>Drupal</th>
<th>ExpressionEngine</th>
<th>Joomla!</th>
<th>OU CMS</th>
<th>Plone</th>
<th>Wordpress</th>
</tr>
</thead>
<tbody>
<tr>
<td>The product shall handle 20 concurrent contributor connections.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall allow high-traffic content to be served without dependence on CMS throughput.</td>
<td>?*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>The product shall run on Web Central or push content to Web Central</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall mirror the availability of the Web Central environment.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>?</td>
<td>Y</td>
</tr>
<tr>
<td>Data Exportability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The product shall publish content data to RSS.</td>
<td>N*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall publish content data to PDF.</td>
<td>N</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y*</td>
</tr>
<tr>
<td>The product shall export CMS data via XML for CMS migration support.</td>
<td>Y*</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall support the integration of device-specific CSS for CMS content. (i.e. printer, mobile).</td>
<td>N*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Authentication Integration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The product shall have a pluggable/modular authentication system.</td>
<td>Y*</td>
<td>Y</td>
<td>?*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
</tr>
<tr>
<td>The product shall have the capability to use LDAP to authenticate users upon login.</td>
<td>Y*</td>
<td>Y</td>
<td>?*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
</tr>
<tr>
<td>The product shall allow the integration of UT EID-based sign-on.</td>
<td>Y*</td>
<td>Y</td>
<td>?*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
</tr>
<tr>
<td>Atomicity of Roles and Authorizations</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Requirement</td>
<td>DjangoCMS</td>
<td>Drupal</td>
<td>ExpressionEngine</td>
<td>Joomla!</td>
<td>OUCMS</td>
<td>plone</td>
<td>Wordpress</td>
</tr>
<tr>
<td>----------------------------------------------------------------------------</td>
<td>-----------</td>
<td>--------</td>
<td>------------------</td>
<td>---------</td>
<td>-------</td>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>The product shall allow granular permission settings for users on discrete content units.</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall support role-based task permission inheritance.</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall allow individual users to be assigned roles.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Nice to have: The product shall have content object-level permissions (potentially sub-page) by user.</td>
<td>N</td>
<td>N</td>
<td>?</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>Nice to have: The product shall have content object-level permissions (potentially sub-page) by role.</td>
<td>N</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>Nice to have: The product shall support dynamic role associations based on group affiliations.</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>The product shall support content versioning</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>Workflow</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The product shall allow moderation queues for CMS content.</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>The product shall allow moderation queue notifications.</td>
<td>?</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>The product shall allow production workflows.</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>The product shall allow workflows to be attached to content based on content type.</td>
<td>?</td>
<td>Y</td>
<td>Y*</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>Product Maturity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The product shall be a mature software offering.</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Nice to have: The product shall have market penetration in the higher education or education industry.</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Active User Community</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The product shall have a large and active user community.</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Requirement</td>
<td>DjangoCMS</td>
<td>Drupal</td>
<td>ExpressionEngine</td>
<td>Joomla!</td>
<td>OUCMS</td>
<td>Plone</td>
<td>Wordpress</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
<td>--------</td>
<td>------------------</td>
<td>---------</td>
<td>-------</td>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>The product's last major application patch shall have been applied within the last 12 months.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall have active bug and issue resolution hot-fixes and patches.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall have a significant number of community add-ons or a significant amount of community-driven development.</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall have a significant amount of industry review and commentary on the system.</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td><strong>Ease of Use</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The product shall allow a non-technical user to contribute with minimal training.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall be configurable to include a browser-based editor for rich text content (e.g. TinyMCE).</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td><strong>Ease of UI Customization</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The product shall have straightforward template creation.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall enable the customization of spaces with individual templates.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall cleanly separate presentation from content for easy updates of style and branding.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td><strong>Web Standards and Accessibility</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The product shall include UTF-8 Support (for foreign language display).</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall produce valid XHTML 1.0 according to W3C standards.</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
<td>Y*</td>
<td>Y</td>
</tr>
<tr>
<td>Feature</td>
<td>DjangoCMS</td>
<td>Drupal</td>
<td>ExpressionEngine</td>
<td>Joomla!</td>
<td>OUCMS</td>
<td>Plone</td>
<td>Wordpress</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>-----------</td>
<td>--------</td>
<td>------------------</td>
<td>---------</td>
<td>-------</td>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>The product shall produce valid CSS 1.0 according to the W3C standard.</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall produce web pages that are compliant with Section 508 of the Federal Rehabilitation Act.</td>
<td>?*</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>?</td>
</tr>
</tbody>
</table>

**Web Standards and Accessibility (cont.)**

<table>
<thead>
<tr>
<th>Feature</th>
<th>DjangoCMS</th>
<th>Drupal</th>
<th>ExpressionEngine</th>
<th>Joomla!</th>
<th>OUCMS</th>
<th>Plone</th>
<th>Wordpress</th>
</tr>
</thead>
<tbody>
<tr>
<td>The product’s interface for content contributors shall comply with Section 508 of the Federal Rehabilitation Act.</td>
<td>?*</td>
<td>Y</td>
<td>?</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>The product’s interface for content contributors shall make it easy for contributors to produce content that meets Section 508 of the Federal Rehabilitation Act.</td>
<td>?*</td>
<td>Y</td>
<td>?</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>The product’s interface for administrators shall comply with Section 508 of the Federal Rehabilitation Act.</td>
<td>?*</td>
<td>Y</td>
<td>?</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
</tr>
</tbody>
</table>

**Rich Media Support**

<table>
<thead>
<tr>
<th>Feature</th>
<th>DjangoCMS</th>
<th>Drupal</th>
<th>ExpressionEngine</th>
<th>Joomla!</th>
<th>OUCMS</th>
<th>Plone</th>
<th>Wordpress</th>
</tr>
</thead>
<tbody>
<tr>
<td>The product shall provide efficient, web interface-driven handling of embedded media, including images and Flash video.</td>
<td>Y*</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

**Technology Alignment**

<table>
<thead>
<tr>
<th>Feature</th>
<th>DjangoCMS</th>
<th>Drupal</th>
<th>ExpressionEngine</th>
<th>Joomla!</th>
<th>OUCMS</th>
<th>Plone</th>
<th>Wordpress</th>
</tr>
</thead>
<tbody>
<tr>
<td>The product shall align with campus supported technologies such as: Python, PHP, MySQL, and Oracle.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

**Modularity/ Ease of Software Customization**

<table>
<thead>
<tr>
<th>Feature</th>
<th>DjangoCMS</th>
<th>Drupal</th>
<th>ExpressionEngine</th>
<th>Joomla!</th>
<th>OUCMS</th>
<th>Plone</th>
<th>Wordpress</th>
</tr>
</thead>
<tbody>
<tr>
<td>The product shall have a modular architecture so that functionality can be extended without extensive refactoring.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall support the definition of block-level, structured, and richly-functioned content types.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Cost-effectiveness</td>
<td>DjangoCMS</td>
<td>Drupal</td>
<td>ExpressionEngine</td>
<td>Joomla!</td>
<td>OUCMS</td>
<td>Plone</td>
<td>Wordpress</td>
</tr>
<tr>
<td>--------------------------------------------------------</td>
<td>-----------</td>
<td>--------</td>
<td>------------------</td>
<td>---------</td>
<td>-------</td>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>The product shall have a reasonable setup cost.</td>
<td>N*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall have a reasonable maintenance cost.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>?</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall have a reasonable licensing cost.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ease of Installation and Setup</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The product shall have an installation time of less than 1 work day.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall be hosted on a standard Linux/Apache environment.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N*</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ease of Maintenance</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The product shall have quick responses to security problems</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall have support for backward compatibility.</td>
<td>?*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

| Nice to have: The product shall notify the administrator of new versions. | N         | Y      | N                | N       | N     | ?     | N         |
| Nice to have: The product shall provide upgrades via the web interface. | N         | N      | N                | N       | N     | N     | N         |
| The product shall have accessible application source code. | Y         | Y      | Y                | Y       | N     | Y     | Y         |

<table>
<thead>
<tr>
<th>Application Security</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The product shall comply with minimum security standards for application development and administration:</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall validate input properly and restrictively.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall encode output for appropriate destination</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

| Application Security (cont.)                           |           |        |                  |         |       |       |           |

Refresh CMS Evaluation Final Report
ITS Applications
December 2009
<table>
<thead>
<tr>
<th>Requirement</th>
<th>DjangoCMS</th>
<th>Drupal</th>
<th>ExpressionEngine</th>
<th>Joomla!</th>
<th>OUCMS</th>
<th>Plone</th>
<th>Wordpress</th>
</tr>
</thead>
<tbody>
<tr>
<td>The product shall execute proper error handling and avoid information leak</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall authenticate users through central authentication system</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall allow for establishment of role or group based authentications</td>
<td>N*</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>The product shall pass review of reported vulnerabilities on NIST and Secunia.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y*</td>
<td>Y</td>
</tr>
</tbody>
</table>
Appendix B – Requirements, Rationale, and Fit Criteria
Availability

Requirement: The product shall handle at least 20 concurrent contributor connections.

Rationale: The product needs to support a fair number of concurrent connections to allow for distributed and synchronous work efforts.

Fit Criteria: The product allows at least 20 concurrent contributor connections.

- Number of concurrent connections allowed?

Requirement: The product shall allow high-traffic content to be served without dependence on CMS throughput.

Rationale: High traffic pages must be highly available and need to mirror current web central performance.

Fit Criteria: The product shall support the pushing of published static content.

Fit Criteria: The product shall support high-availability caching

- Describe the page pushing/publishing process?
- What is the typical performance of pushing (time, pages, etc)?
- List performance metrics on high availability caching
- Describe any metrics around this CMS's throughput

Requirement: The product shall run on Web Central or push content to Web Central

Rationale: Web Central is our systems supported infrastructure for www.utexas.edu and our content must be served from there. The product must be able to push content to Web Central either by running there and having file/folder level access or through SFTP or shared key.

Fit Criteria: The product shall run on *NIX-based systems

Fit Criteria: Publishing/Pushing mechanism shall access Web Central publishing directories through currently existing authentication and authorization strategies.

- What Operating Systems are supported?
- Which Operating System version has the highest level of installs?
• What authorization and authentication methods are supported for the pushing/publishing process?

**Requirement:** The product shall mirror the availability of the Web Central environment. (web central web server environment)

**Rationale:** Must duplicate the availability of publishing processes that we currently support.

**Fit Criteria:** The product shall have high availability features (99.95% uptime)

• What is a typical level of downtime for support on this product?
• What high availability features or configurations are supported?

**Requirement:** The product shall mirror the availability of the Web Central publishing environment. (Administrative or publishing function)

**Rationale:** Must provide reasonably high availability of administrative interfaces (99.5%).

**Fit Criteria:** The product shall have high availability features (99.5% uptime)

• What is a typical level of downtime for support on this product?
• What high availability features or configurations are supported?

**Versioning**

**Requirement:** The product shall support content versioning

**Rationale:** Versioning of content can preserve authoring trails and allow for multi-level undo and content archaeology. Retaining versions of content will allow administrators and business owners to protect their content investment.

**Fit Criteria:** The product shall version CMS content.

**Fit Criteria:** The product shall support CMS content versioning with meta-data including author and editing date.
Data Exportability

**Requirement:** The product shall publish content data to syndicated formats.

**Rationale:** Increasing our data portability is one of the goals of the refresh project. Being able to leverage publishing new and existing content to standards-based syndication formats will help accomplish that goal.

**Fit Criteria:** The product shall support publishing content to syndicated formats.

- List some URLs of this feature in use
- What syndication features are available for this CMS?
- What different syndication formats are available for this product?
- Are there any limitations or caveats for syndication use and this CMS?

**Requirement:** The product shall publish content data to PDF.

**Rationale:** Increasing our data portability is one of the goals of the refresh project. Being able to leverage publishing new and existing content to widely available reading formats will help accomplish that goal.

**Fit Criteria:** The product shall support publishing content to PDF

- List some URLs of this feature in use
- How is PDF publishing supported (is it out of the box, an add-on)?
- What PDF features are available for this CMS?
- Are PDF forms supported?
- Are encrypted or secure PDFs supported?
- Are there any limitations or caveats for PDFs use and this CMS?

**Requirement:** The product shall export CMS data via XML for CMS migration support.

**Rationale:** As technologies and support levels change we need to be able to migrate our content easily from product to product. Standards based exporting can assist migration efforts and reduce overall level of effort.

**Fit Criteria:** The product shall support XML export of all CMS content

- What other export/migration options are supported?
- Describe any issues around product upgrades and content migration?
- Describe any issues around migrating content to and from this CMS?
- Links to discussions and comments around migrations
**Requirement**: The product shall support the integration of device-specific CSS for CMS content. (i.e. printer, mobile).

**Rationale**: Publishing material with either device specific CSS or device specific styling will ease our content portability efforts.

**Fit Criteria**: The product shall allow multiple style sheets to be applied to published content

- Does CMS allow content to be published in multiple styles?
- Is there any mobile platform publishing feature? If so, describe.
- Is there any printer-friendly publishing feature? If so, describe.

### Authentication Integration

**Requirement**: The product shall allow the integration of UT EID-based sign-on.

**Rationale**: Leveraging EID sign on will mirror the standard log-in mechanism for most of campus’ secure service offerings.

**Fit Criteria**: The product shall integrate with the EID through LDAP or LDAP module or mod_auth_eid headers.

- What is the likely method of integration? (i.e. use mod_auth_eid headers like trac?)
- Is CMS authentication system is pluggable and modular?
- Provide links to issues/discussion surrounding integrating existing authentication systems and this CMS.
- List most common authentication integration issues.
- Is LDAP integration for authentication supported?
- Describe any issues in the latest version around LDAP authentication integration.
- Detail LDAP support
Atomicity of Roles and Authorizations

Requirement: The product shall allow granular permission settings on discrete content types.

Rationale: Atomic permissions on contributor content will increase delegation and task distribution. It will also increase oversight and improve security.

Fit Criteria: The product shall have authorizations for individual content types. What limitations exist around atomic permission assignments?

Requirement: The product shall support role-based task permission inheritance.

Rationale: Groups of permissions make user management tasks and support much easier.

Fit Criteria: A privileged user shall be able to assign an individual to a role/group, thereby entitling the assigned individual to the permissions given to that role/group. (groups of permissions assignable by role/group)

• What limitations exist around role-based permission inheritance?
• Are role permissions customizable?
• Are additional roles able to be created?
• What roles are available if new roles cannot be added?
• Do the roles map to contributor/moderator/administrator?
• What level of atomicity is assignable by role?

Requirement: The product shall allow individual users to be assigned roles/groups or a combination of the two.

Rationale: Assigning individuals to roles with sets of permissions makes user management tasks and support much easier.

Fit Criteria: An administrator shall be able to assign and de-assign an individual to a role or group.

• Are there any limits on number of users?
• Are there any limits on number of users assigned a role?
• Are groups and roles both offered (permission groups and organizational groups)?
• How are the concepts of groups and roles handled in this CMS product?
**Requirement:** The product shall have content object-level permissions (potentially sub-page) by role.

**Rationale:** Mapping content object-level permissions to roles eases user administration tasks and increases delegation and distribution of task effort, which keeps content up-to-date and supports workflows.

**Fit Criteria:** A privileged user shall be able to grant/restrict a role/group to a sub page content item.

- What is the smallest level of permissions available on content items (page, content type, chunk, etc) to a role?
- Are there limits to the number of role permissions at this level?

**Requirement:** Nice to have: The product shall have content object-level permissions (potentially sub-page) by user.

**Rationale:** Increasing the granularity of content permissions will increase the delegation and distribution of task effort, which may keep content more up-to-date and support complex work flows.

**Fit Criteria:** A privileged user shall be able to grant/restrict permission for an individual to a sub page content item.

- What is the smallest level of permissions available on content items (page, content type, chunk, etc) to a user?
- Are there limits to the number of user permissions at this level?

**Requirement:** Nice to have: The product shall support dynamic role associations based on external data (external group) about users.

**Rationale:** Leveraging existing group memberships or affiliations to automatically create role-based permissions could ease support efforts and allow for some automation to the same end.

**Fit Criteria:** An authorized user shall be able to utilize existing user group memberships from an external source to assign role/group permissions in the product.

- What mechanisms exist to automate user creation?
- What mechanisms exist to automate group assignments?
**Workflow**

**Requirement**: The product shall allow multi-step workflows.

**Rationale**: Workflow processes allow for pre-publication review of content and separation of concerns between content authoring/editing and approval.

**Fit Criteria**: A privileged user shall be able to create a workflow process of more than one step for a piece of content.

- Describe workflow process support.
- How many workflow steps are allowed?

**Requirement**: The product shall allow notifications at each step of the workflow.

**Rationale**: Moderation notifications assist the moderation process and help content stay fresh by keeping moderators and contributors in the loop about the status of their content.

**Fit Criteria**: An indicated user shall be able to be notified via email or other channel as part of the workflow process. What types of notifications are supported?

- Are supporting technologies required to facilitate messaging?
- Describe the complexity of enabling notifications for workflows for this product.

**Requirement**: The product shall allow custom workflows.

**Rationale**: Customizable workflow processes allow for flexibility in the separation of concerns between content authoring/editing and approval in a large, distributed environment.

**Fit Criteria**: A privileged user shall be able to create multiple workflows for the product and the creation of one will not invalidate or destroy the other.

- Describe customizable workflow process support.
- How many steps are allowed per customizable workflow?

**Requirement**: The product shall allow customizable workflows based on content type.

**Rationale**: Different kinds of content can require different types of workflows involving different business units and owners.
**Fit Criteria:** A privileged user shall be able to create multiple workflows that are assigned and enacted on a piece of content based on its type.

- Are there any limits to the number of workflows which can be created?
- Are there any limits to assigning workflows to types of content?
- Are there any functions of the product which cannot be given workflows? Which functions?

**Product Maturity**

**Requirement:** The product shall be mature.

**Rationale:** Mature software has well-established support and development cycles.

**Fit Criteria:** The product shall have been in production with active support and development cycles for at least the past 3 years.

**Fit Criteria:** The product shall have demonstrated development cycles including feature elicitation, development, testing, release and support.

- Links to FAQs on development philosophy
- How many releases has the product had?
- How long has the product been on the market?
- List any significant changes to the CMS in terms of product direction, project ownership, conversion from private to open-source.
- How long is a typical development cycle for a major version for this CMS?

**Requirement:** Nice to have: The product shall have market penetration in the higher education or education industry.

**Rationale:** The higher education industry is unique in its decentralized distribution which lacks a corporate mirror. Proven implementations at other large universities would be helpful in demonstrating a product’s applicability to our industry.

**Fit Criteria:** The product shall be in use by other higher education institutions.

**Fit Criteria:** The product shall be in use by other education-based organizations.

- List links to white papers or examinations by other higher education institutions.
- Any public lessons learned on implementing this product at a higher education institution available (provide links).
- List some uWebD comments and questions on this product.
**Active User Community**

**Requirement**: The product shall have a large and active user community.

**Rationale**: Active and diverse user communities signify a product's health and interest in the product. They also guarantee dispersed expertise in a variety of industries and a variety of environments.

**Fit Criteria**: Official forums shall have memberships in the thousands

**Fit Criteria**: Unofficial and official forums shall have total posts numbering in the thousands

**Fit Criteria**: There shall have been more than 200 posts in the past 10 days

Fit Criteria: The product shall be represented by its own "tag" or topic on support/discussion sites such as stackoverflow.com.

- How many members belong to the official forums
- How many posts have been made in the past 10 days on the official forums?
- Do most posts on the official forums receive at least one response?
- Are there at least 2 or 3 independent forums or support sites dedicated to this product?

**Requirement**: The product's last major application patch shall have been released within the last 12 months.

**Rationale**: Frequency of application patching can indicate a product's life cycle health.

**Fit Criteria**: The product's last major application patch shall have been released within the last 12 months.

- When was the date of the last major application patch?
- When was the date of the last major application patch before this one?

**Requirement**: The product shall have active bug and issue resolution hot-fixes and patches.
Rationale: Active bug patching and issue resolution hot-fixes indicate a product which responds to community and internally discovered errors, bugs, and issues.

Fit Criteria: Major releases shall be followed with issue resolution or bug-patching fixes.

Fit Criteria: Project issue status and patch information shall be available and prominent on the product web site.

- How many hot-fixes are there for the most recent release?
- How frequently were issue patches released for the previous release?
- For high impact issues, what was the support cycle in terms of days?

Requirement: The product shall have a significant number of community add-ons or a significant amount of community-driven development.

Rationale: Significant community-driven development indicates a level of adoption and support for the product that will result in strong support from the core developers.

Fit Criteria: Product web site shall host a list of community add-ons.

- Is there a centralized issue tracking system for community add-ons?
- Are community developers working to make new versions available in sync with major releases of the core product?

Fit Criteria: Product shall have a well documented API enabling community developers to easily extend core functionality.

Requirement: The product shall have a significant amount of industry review and commentary on the system.

Rationale: The product shall be well-known enough to have been a focus of critiques and evaluations by trusted technology sites or companies.

Fit Criteria: The product shall be reviewed by a major technology website or organization.

- List links to such critiques or reviews.
### Ease of Use

**Requirement:** The product shall allow a non-technical user to contribute with minimal training.

**Rationale:** A substantial number of users for this product will not have a technical background, and need to be able to understand and use the product without extensive training or other costs.

**Fit Criteria:** A content creator or editor shall be able to edit and publish without any knowledge of HTML, CSS and other web formats.

**Requirement:** The product shall be configurable to include a browser-based WYSIWYG editor for rich text content (e.g. TinyMCE).

**Rationale:** A WYSIWYG rich text editor has become a de-facto standard method for web content editing, and should be implemented either as a core feature, or an easily installable extension/module.

**Fit Criteria:** The product shall include a browser-based WYSIWYG editor, or an extension/module should exist to easily integrate one of the many open-source WYSIWYG editor projects.

### Ease of UI Customization

**Requirement:** The product shall have straightforward template creation.

**Rationale:** Template creation should be possible without needing in-depth knowledge of the software itself, aside from a template language.

**Fit Criteria:** Templates shall be able to be dropped into a directory or created through the CMS’s web interface.

**Fit Criteria:** Documentation shall be available on creating templates, customizing existing templates, and the template language itself.

**Requirement:** The product shall enable the customization of spaces with individual templates.
**Rationale:** Allowing page sections and content types to render from their own templates allows sections and pages of the site to be easily customized.

**Fit Criteria:** Product shall allow definition of a base template or set of base templates for customizing the site's header, footer, and navigation menus.

**Fit Criteria:** Product shall allow definition of templates for each content type.

**Requirement:** The product shall cleanly separate presentation from content for easy updates of style and branding.

**Rationale:** Users need to be able to edit and publish content without having to understand or modify the presentation of that content. Designers need to be able to change the presentation of such content without having to edit the content.

**Fit Criteria:** An authorized user shall be able to edit and publish content without having to edit or understand the underlying presentation format (HTML/CSS, etc.).

**Fit Criteria:** Edits made to design elements shall cascade across the site without having to touch individual content items.

### Web Standards and Accessibility

**Requirement:** The product shall include UTF-8 Support (for foreign language display)

**Rationale:** UTF-8 Support allows foreign language characters to display properly without creating unrecognizable or unintended display characters.

**Fit Criteria:** The product shall have support for the UTF-8 character set.

**Requirement:** The product shall be capable of producing content that renders in valid XHTML 1.0 according to W3C standards

**Rationale:** Standards-based XHTML ensures cleaner rendering, faster page loads, and conforms to the web publishing standards of the University of Texas at Austin.

**Fit Criteria:** The product renders content that shall pass XHTML validation tests.
**Requirement:** The product shall be capable of producing valid CSS 1.0 according to the W3C standard

**Rationale:** Standards based CSS 1.0 style markup and style sheets ensures cleaner rendering, broader support, and more uniform presentation of content and conforms with the web publishing standards of the University of Texas at Austin.

**Fit Criteria:** The product renders style sheets that shall pass CSS 1.0 validation tests.

---

**Requirement:** The product shall be capable of producing web pages that are compliant with Section 508 of the Federal Rehabilitation Act

**Rationale:** Federal law requires web site compliance with section 508. In addition it conforms to the web publishing guidelines of the University of Texas at Austin.

**Fit Criteria:** Rendered pages shall have the capability to pass section 508 automated testing.

**Fit Criteria:** Rendered pages shall have the capability to pass section 508 manual testing.

---

**Requirement:** The product's interface for content contributors shall comply with Section 508 of the Federal Rehabilitation Act.

**Rationale:** Federal law requires compliance with Section 508 of the Federal Rehabilitation Act. In addition it conforms to the web publishing guidelines of the University of Texas at Austin.

**Fit Criteria:** The product’s content contributor section shall pass section 508 automated testing.

**Fit Criteria:** The product’s content contributor section shall pass section 508 manual testing.

---

**Requirement:** The product's interface for content contributors shall make it easy for contributors to produce content that meets Section 508 of the Federal Rehabilitation Act.
**Rationale:** Assisting content contributors in complying with Federal requirements will ensure greater adherence to the law.

**Fit Criteria:** Product shall be able to flag content for common 508 compliance pitfalls like alt text, .acrobat files and other media links, and captioning for images.

**Fit Criteria:** Product shall be customizable to require common 508 accessibility fixes, i.e. requiring alt text.

- Are there any other special considerations this product makes for 508 compliance or accessibility?

**Requirement:** The product's interface for administrators shall comply with Section 508 of the Federal Rehabilitation Act.

**Rationale:** Federal law requires compliance with Section 508 of the Federal Rehabilitation Act. In addition it conforms to the web publishing guidelines of the University of Texas at Austin.

**Fit Criteria:** The product's administrative interface shall pass automated section 508 testing.

**Fit Criteria:** The product's administrative interface shall pass manual section 508 testing.

---

**Rich Media Support**

**Requirement:** The product shall provide efficient, web interface-driven handling of embedded media, including images and Flash video.

**Rationale:** Content contributors should be able to include/embed rich media assets with a minimum of effort.

**Fit Criteria:** An average user of the product shall be able to embed an image into a piece of content via the web interface.

- Does the product support the upload and embedding of images through the web interface?
• Does the product support the embedding of Flash video objects through the web interface?

**Technology Alignment**

**Requirement:** The product shall align with campus supported technologies such as: Python, PHP, MySQL, and Oracle.

**Rationale:** Deployment and maintenance will be quicker, cheaper and easier if the product uses technologies that are already in use at the University.

**Fit Criteria:** The product shall be implementable in either PHP or Python

Fit Criteria: The product shall have a database backend of MySQL or Oracle.

**Modularity/ Ease of Software Customization**

**Requirement:** The product shall have a modular architecture so that functionality can be extended without extensive refactoring.

**Rationale:** Local modifications and additions are easier and less risky when the product supports modular extension; direct modifications to the main codebase typically introduce upgrade issues.

**Fit Criteria:** The product shall provide an explicitly-supported, well-documented extension/plug-in API that provides enough functionality to support anticipated modifications.

**Requirement:** The product shall support the definition of block-level, structured, and richly functioned content types.

**Rationale:** Custom content types allow storage, functional, and display flexibility for content owners without modifications to the CMS itself.

**Fit Criteria:** Product shall allow for creation of new content types, with custom data storage models, sets of functional methods, and display templates.

**Cost-effectiveness**

**Requirement:** The product shall have a reasonable setup cost.
Rationale: Our resource/staffing availability aligns with a product that does not require extensive developer ramp-up time or extensive hardware/software resources. The product needs to be able to be quickly set up by a skilled developer having little previous experience with the product, using existing or easily acquired hardware and software resources.

Fit Criteria: The product shall be able to be installed in less than one working day by a developer with no previous experience with the product?

Fit Criteria: The product shall be able to run in an existing server environment or one that is easily provisioned (e.g. *NIX-based virtual machine)?

Requirement: The product shall have a reasonable maintenance cost.

Rationale: Our resource/staffing availability aligns with a product that needs minimal staff time to keep running and minimal downtime needed for upgrades and security fixes. The product needs straightforward upgrades and maintenance.

Fit Criteria: The product shall provide automated upgrade functionality

• Does the product require downtime for upgrades, template changes, new content type definitions, etc.?
• Are there any regular clean-up tasks required for efficient operation?

Requirement: The product shall have a reasonable licensing cost.

Rationale: Cost-aware solutions that mirror the models of open source and near-open source software closely align with ITS Applications funding strategies and efficiency seeking measures.

Fit Criteria: The product should be made available for use without extension, including for testing and internal development, for no more than $1,000 a year.

Ease of Maintenance

Requirement: The product’s developers shall have quick responses to security problems

Rationale: The product’s developers need to address vulnerabilities quickly to ensure content and data integrity.
**Fit Criteria:** Product displays a good track record for responding to public vulnerabilities in the eyes of its developers and CMS community.

**Fit Criteria:** The product's developers or community responds to high threat vulnerabilities quickly (1 month or less between identification of issue and patch).

- Are new product versions released after security issues have been addressed?
- How long are the product's versions supported for receiving security updates?

**Requirement:** The product shall have support for backward compatibility.

**Rationale:** Minor releases should maintain backward compatibility so that upgrades for performance improvements, security issues and bug fixes can be routinely deployed without migration issues or code incompatibilities.

**Fit Criteria:** The product's producers shall have a specific policy indicating that backwards compatibility for minor releases included in product documentation or website.

**Fit Criteria:** The product shall have a history of backward compatibility.

**Requirement:** The product shall notify the administrator of new versions.

**Rationale:** Version and patch update notifications can ease administration tasks and can assist support processes.

**Fit Criteria:** The product shall have a new version notification widget in the administrator's interface.

**Fit Criteria:** The product shall notify administrators of new versions via email.

**Fit Criteria:** New versions of the software shall be indicated via syndication or other channels.

**Requirement:** The product shall provide upgrades via the web interface.

**Rationale:** Product upgrades are more likely to be installed quickly and to minimize downtime if the upgrade can be accomplished from the web interface.
Fit Criteria: Administrators shall be able to install upgrades via the web interface.

Requirement: The product shall have accessible application source code.

Rationale: The availability of source code assists development efforts, support, and encourages community innovation.

Fit Criteria: There shall be a publically accessible source code repository for the product.

- How many versions of the source code are available?
- What are the restrictions on source code use?
- Is the source code accessible via SVN, ant or other tools?
- What is the location of that source code repository?

Application Security

Requirement: The product shall comply with the minimum security standards for Application Development and Administration, http://www.utexas.edu/its/policies/opsmanual/appstd.php

Rationale: This Standard provides a list of minimum information security practices that help to reduce risk to campus assets. Compliance with this standard is mandatory.

Fit Criteria: The Standards should be consulted to get a complete list of required practices; however chief among them are the following:

- To minimize the risk of client and server side injection flaws the application should provide proper and restrictive input validation (auditing, filtering) and appropriate output encoding (e.g. data binding, HTML entity encoding)
- To minimize exposure of potentially sensitive configuration information, the product shall execute proper error handling.
- In order to minimize the risk of broken authentication or session management, the product shall authenticate users through central authentication system and ensure that authentication tokens and or session tokens are protected in transit and removed once a session has ended or expired.
• In order to facilitate annual authorization audits and deprovisioning procedures, the product should support role or group based authorizations.

**Requirement:** The product shall pass review of reported vulnerabilities on NIST and Secunia.

**Rationale:** Common security vulnerabilities represent the lowest threshold of computer attacks. Guarding against these techniques and the way in which these techniques are addressed relates a product’s security methodology and its’ resilience to hacking.

**Fit Criteria:** The product shall pass a review of reported vulnerabilities on NIST and Secunia.