

Drupal Code Review Procedure

Drupal Code Review Procedure

Basic requirements

1. There should be NO custom coding to Drupal Core or to any contributed components (modules, themes, libraries).
2. Modifications to contributed modules or themes should be implemented as custom modules or subthemes.
3. Disable, uninstall, and remove all contributed modules, libraries, and themes NOT IN USE.
4. Documentation (README.txt) for custom modules is required.
5. Custom modules reside in a **sites/all/modules/custom** folder.
6. Custom themes (subthemes and contributed base themes) reside in a **sites/all/themes** folder.
7. Remove any debugging and/or commented out code.
8. Remove development modules.
9. Remove any source and version control files and folders (ie. .svn .dstore)
10. Stable releases of any contributed modules are preferred; flag any dev versions and provide explanation for why it was included.
11. Do not use the Features module in your application. We do not support Features because it interferes with our Drupal update process. Our developers are happy to discuss alternatives to using the Features module in your application.

Vendor deliverables

1. Output of **phpinfo** command run at vendor site.
2. List of all modules and libraries in use.
3. Documentation on any customized components
 - a. Include README.txt for custom modules
 - b. This documentation should include at a minimum the purpose of the module and installation instructions.
4. A single tarball of entire site (core and all)
 - a. **sites/all/modules/custom** folder should include all custom modules
 - b. **sites/all/themes** folder should include all custom subthemes and contributed base themes
 - c. The site should include a database folder containing the output of the mysql database unload.
 - d. The site should include a documentation folder containing the phpinfo command output.
 - e. Any version control files (.svn etc) should be excluded.

Internal review process

1. Deploy supplied site in our development virtual environment.
2. Examine phpinfo output for conflicts with Dev environment.
3. Review that the supplied list of modules is accurate.
4. Verify custom modules and themes are in proper folders (**sites/all/modules/custom** **sites/all/themes**)
5. Check for modules in need of updating.
6. Check for disabled modules.
7. Verify release levels of any contributed modules.
8. Install and run 'Hacked' which checks whether core or contributed modules have been modified.
9. Install and run 'Coder' which checks for compliance with [Drupal coding standards](#).
10. Examine README.txt supplied with custom modules.

If you wish to discuss a Service Level Agreement, [contact ops-help](#).