Code Review Guide

Syntax and codestyle

Almost everything regarding syntax and codestyle should be automated using sniffers and lints.

Architecture

- SOLID
- GRASP
- Design Patterns

Random

Check that each function is doing only a single thing.
Unit/Integration test are created/updated
Never allow unit test that are written to show 100% coverage and doesn’t do anything that unit test is supposed to do. Like a unit test code which just call the function to get 100% coverage and doesn’t have any assertions.
Always try to use constants in the left hand side of the comparison. That is instead of doing if ($variable == 2) always use if (2 == $variable) because this will help to identify the errors in the earlier stage of development even if we miss and “=” from that statement
Never ever mix the php code and template (view). In ideal condition a view should not contain any logic.
Never allow bad code with some good comments. Only bad code needs comment
Follow our coding standards
Check Error Handling
Make sure the code has a good performance. Try to avoid too many reading/writing even if it is MemCache. Be gentle with Bob, he is alone, he is weak man
Try to find security holes
Is the code reusable?
Edge cases are considered?
Check the code will have the same behavior on Staging/Live. For example: a file resource is used. Maybe the upload directory is different on sandbox and staging.
Repetitive code has been factored out
Checking undefined variable