Naming Standards

1. Capitalize first letter of each word. (ContributedBy)
    a. Identifies each word easily.
2. No separation between words on multi word names. (ContributedBy)
    a. Keeps names small and simple.
3. Each word spelled out not abbreviated. (ContentTypeId)
    a. Avoids confusion and needing a document identifying the abbreviated words.
4. Universally recognized abbreviations will be accepted.
    a. See attached exceptions list.
5. Each instance of a word needs to be the same.
    a. If we use “Email” in one table it needs to be “Email” in all other tables.

As Is System	To Be System
reviewer_id	ReviewerId
FName	FirstName
Username	Username
CurrEv	CurrentEvent
num_reviews	NumberOfReviews
ContType_id	ContentTypeId

Code Standards

1. Indentation
   a. Four spaces shall be used to indent. Once the programmer chooses the number of spaces to indent by, then it is important that this indentation amount be consistently applied throughout the program. **Tabs shall not be used for indentation purposes.**
2. Variables
   a. Variable shall have mnemonic or meaningful names that convey to a casual observer, the intent of its use. Variables should be documented prior to its first use.
3. Open/Close Tags
   a. Open and close tags should be on separate lines with the same indentation.
      i.
      ```
      <head>
          <title>
              Example Site
          </title>
      </head>
      ```
4. Line Length
   a. Each line shall not contain more than 80 characters.
5. Wrapping Lines
   a. If the line is going to exceed the line length follow these rules:
      i. Break after a comma.
      ii. Break after an operator.
      iii. Align the new line with the beginning of the expression on the previous line.
6. CSS
   a. Add CSS through external files, if possible. It should always be in the HEAD of the document.
   b. Format CSS with selectors on one line and each property on its own line.
      i. The declarations are indented
   c. If you specify multiple selectors start each on a new line.
7. PHP
   a. Always use <?php ?> to delimit PHP code, not the <? ?> shorthand.
   b. Arrange code by putting logic at the top of a file and output at the bottom of the file.

c. Control statements should have one space between the control keyword and opening parenthesis.

d. Always use curly braces even in situations where they are technically optional.

e. Functions should be called with no spaces between the function name, the opening parenthesis, and the first parameter.

    i. Spaces between commas and each parameter, and no space between the last parameter, the closing parenthesis, and the semicolon.

        1. $var = foo($bar, $baz, $quux);

f. There should be one space on either side of an equals sign used to assign the return value of a function to a variable.

    i. Block of related assignments, more spaces may be inserted to promote readability.

```
1. $short         = foo($bar);
2. $long_variable = foo($baz);
```

g. Beginning brace and ending brace should be on their own lines at the same indentations.

    i. function foo_func($arg1, $arg2 = '')

```
function foo_func($arg1, $arg2 = '')
{
  if (condition) {
    statement;
  }
  return $val;
}
```

h. Use of C style comments (/* */) and C++ comments (//) are both fine.

i. Comment sections of code to describe how it works.

8.

Exceptions List

1. Email
    a. Abbreviation for Electronic Mail.
2. Username
    a. Being treated as one word as to conform to “password”.
3. ID
    a. Abbreviation for Identification
4. Anon
    a. Abbreviation for anonymous
5. index.html
    a. index.html is the default document used for the website.
6.