**RandomPageRedirect Design Manual**

**Problem definition**

Many larger wiki systems on the internet provide the option to traverse to a "random" page upon clicking a button. This can allow users to explore pages they may normally not see or know about. A feature like this is important for wiki systems with thousands of pages since users may not always traverse through the entire list of available pages in the sidebar due to sheer length.

**Requirements (User Stories)**

A navbar button to go to a random page on the wiki will be available in the top bar. A user will click this button and be directed to a "random" page. A user can also type /random/ as the url to call this function

**Design/Implementation plan**

First, for every page on the wiki, I will load the page into a list. The pseudo random number generator from the random library will be used to generate a number between 0 and the number of pages - 1. The RNG will be seeded with the current system datetime. Then the page that matches that number in the array will be traversed to.

**Test plan**

I will first ensure that every page is being accurately read into the array, including new ones. The random module in Python uses the Mersenne Twister PRNG algorithm to generate its numbers. While these results may not be truly random in the sense of the word, they should be random enough for the purposes of this feature. I will test what redirects the random page brings me to. I will also see what integer value is generated through multiple test runs and see if the distribution is what would be expected.

I designed my feature by first understanding how the wiki system works and what modifications needed to be done. I added an API folder for RandomPageDirect. This is where the core implementation goes that the existing wiki system files can call upon to do the redirects. Test cases are also included in the RandomPageDirect API folder but are in their own subfolder. I created a test content folder to use for having a consistent set of pages to access.