COMP 490: Individual Primer Project

### Due Date: Sept 7, 2015 Submission Subject: Primer

## General Submission Criteria:

All laboratory assignments of must be submitted using git. In addition to the submission of the assignment, the following criteria must be met for all assignments.

1. Your laboratory software project must be stored within a publically accessible git repository. Given a URL, a pull request can be made.
2. This git repository must have been updated via a series of “commit”s and “push”es.
3. You must have a README file associated with your software project, and it must provide a log/journal of the activities you performed on your project and it must provide a summary of your project status.
4. An email is sent with the appropriate subject to [steve@my.csun.edu](mailto:steve@my.csun.edu) that
   1. announces that the assignment is complete and ready to be graded,
   2. contains a copy of your README document, and
   3. includes the git URL for download and testing.
5. Your submission must include everything to deploy your project within a “public\_html” directory.
6. I will be testing your application on Campus web environment: http://www.csun.edu/~steve/

## Overview:

In this project, you are to create a simple web application. This web application can be written in any programming language of your choice. However, you must use the simple CGI interface. I.e., you can’t use the vendor supplied libraries to read access the HTTP Header information.

The functionality of your web application is to be defined by you. There is a minimal number of features that are required, you are free to add additional features.

## Features:

Your web application must have the following features:

* It must respond to the GET verb.
* It must have a different response given the URI provided to the application.
* It must return HTML and appropriate links to any associated CCS sheets or Javascript (I.e., do not inline style or dynamic behavior within your structure).
* It must have least one style sheet.
* It must consume information information from the file system.
* It must consume information from a remote web-server  
  (i.e., you are acting like a proxy server.).

## Conditions:

* You may use any programming language, but you must follow the CGI conventions  
  (mkdir public\_html/cgi-bin, chmod … )
* You must obtain all information about the request directly from environment variables  
  (See simple.cgi program)
* You must handle all URIs within your namespace.   
  (See Mod Rewrite Rules)
* You must emit both the HTTP header and body

**Mod Rewrite Rules** (.htaccess)

|  |
| --- |
| <IfModule mod\_rewrite.c>    RewriteEngine On  # Redirect Trailing Slashes...  RewriteRule ^(.\*)/$ /$1 [L,R=301]  # Handle Front Controller...  RewriteCond %{REQUEST\_FILENAME} !-d  RewriteCond %{REQUEST\_FILENAME} !-f  RewriteRule ^ index.php [L]  </IfModule> |

**Simple CGI**

|  |
| --- |
| **$ cat simple.cgi**  #! /bin/bash  # This is a little CGI program  ###################################################################  # The following are environment variables that are available to you  #  # CONTENT\_TYPE: The desired MIME type for the response.  # CONTENT\_LENGTH: The length of the query information.  # GATEWAT\_INTERFACE: Currently CGI/1.1  # HTTP\_HOST: The name of the vhost of the server.  # HTTP\_USER\_AGENT: Information about the requesting client.  # QUERY\_STRING: The query string.  # REQUEST\_METHOD: The method used to make the request.  # REQUEST\_URI: The URI of the request.  # SERVER\_PROTOCOL: Currently “HTTP/1.1”.  # SCRIPT\_FILENAME: The full path to the CGI script.  # SCRIPT\_NAME: The name (i.e., URI) of the CGI script.  # SERVER\_NAME: The server's hostname or IP Address.  # SERVER\_PORT: The port of the server.  # Add a content type and a blank line  echo "X-COMP-490: ${USER}"  echo "Content-type: text/plain"  echo ""  if [ -n "${QUERY\_STRING}" ] ; then  cat ./${QUERY\_STRING}  fi |