**Web Programming Design Document**

By: Glenroy Logan

**Table of Contents**

[Problem 3 Design 3](#_bookmark0)

[Overview 3](#_bookmark1)

[“How it works” 3](#_bookmark2)

[Notes 3](#_bookmark3)

[Test Plan 4](#_bookmark4)

# System Design

## Overview

The design for this project requires the use of external PHP scripts to validate user input from a HTML form then insert this data into a MySQL or xampp database if it is valid.

## “How it works”

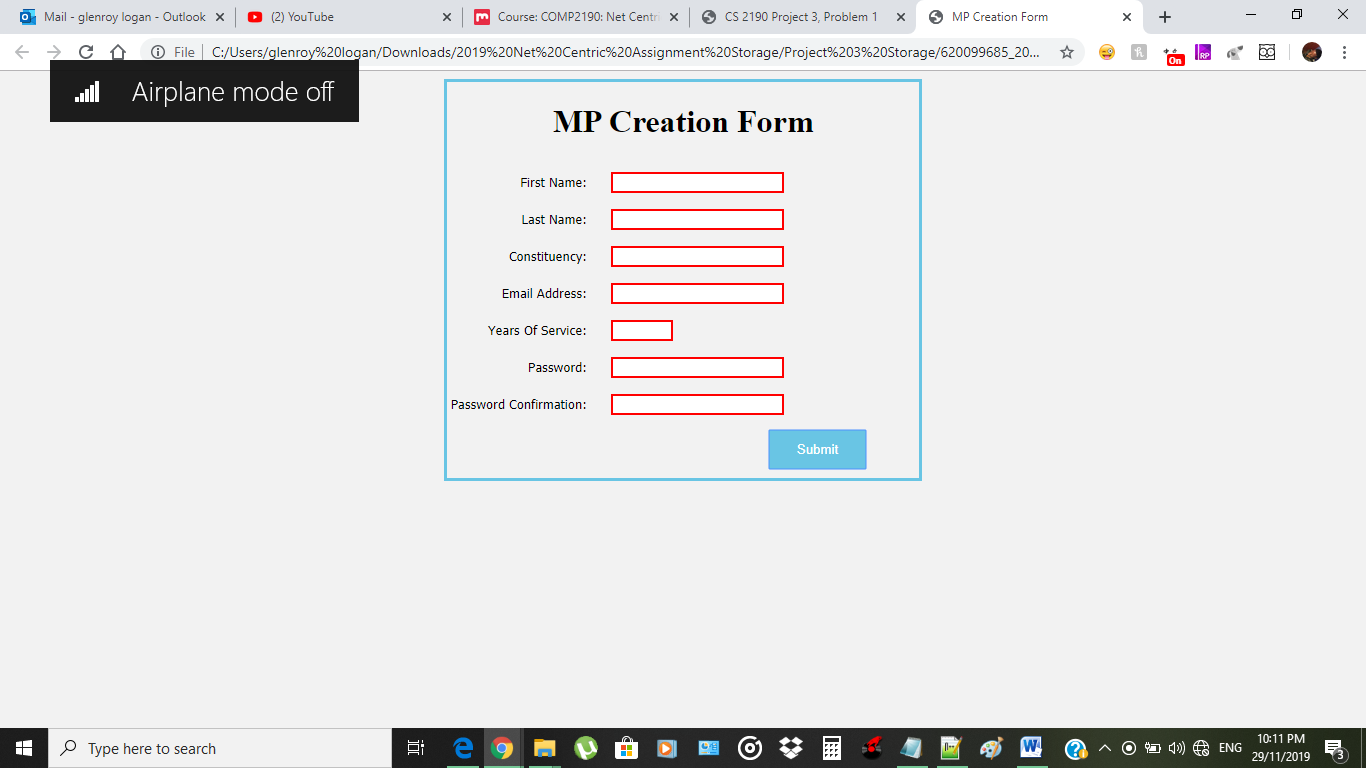
The HTML form displays various input text fields to gather data from a user who is looking to create a member of parliament (MP) entry into a system. This data submitted has both client and server side validation. After the data passes the client-side Javascript validation the form is submitted using the HTTP\_POST method to a PHP script. The script then extracts the data from the super global “\_POST” variable in PHP and places it into a custom ‘form’ object. The script then validates several fields of the data such as:

* None of the fields listed is empty,
* The email address contains at least one letter to the left of the ‘@’ symbol, at least one character immediately after the ‘@’ symbol, then a period, and then at least one character after the period.
* Years of service is a number between 0 and 50.
* The values of Password and Password confirmation match.

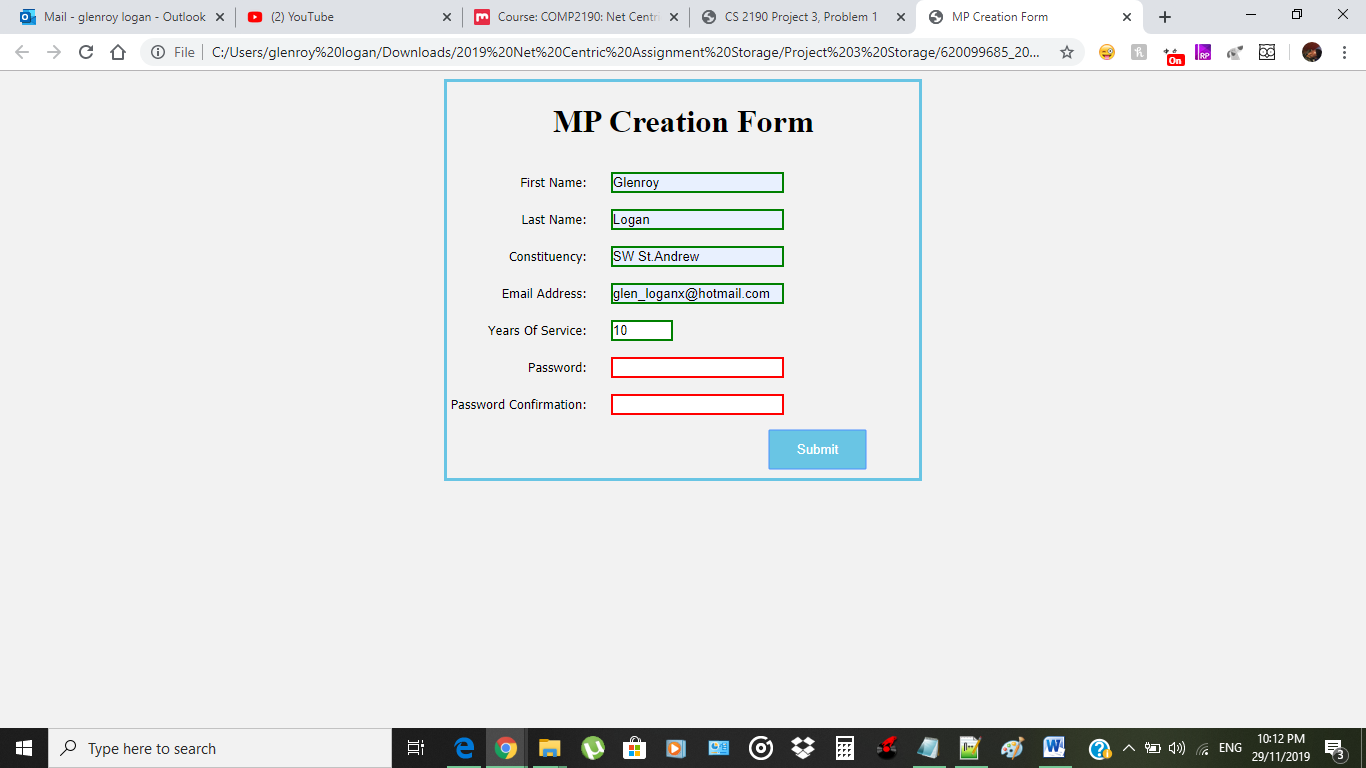
Before the data is validated a connection to the mySQL database is established and validation commences if the connection was successful. After validation, if the data passes all tests, all fields are stored in the “MPRepresentativesTableCreate” table in the “MPMgmtDB” database on the mySQL server hosted locally. In order to execute SQL commands the PHP object-oriented ‘PDO’ methods were used to connect and “INSERT” data into the table. After validation and possible insertion into the database the connection between the php script and mySQL database is closed.

## Test Plan

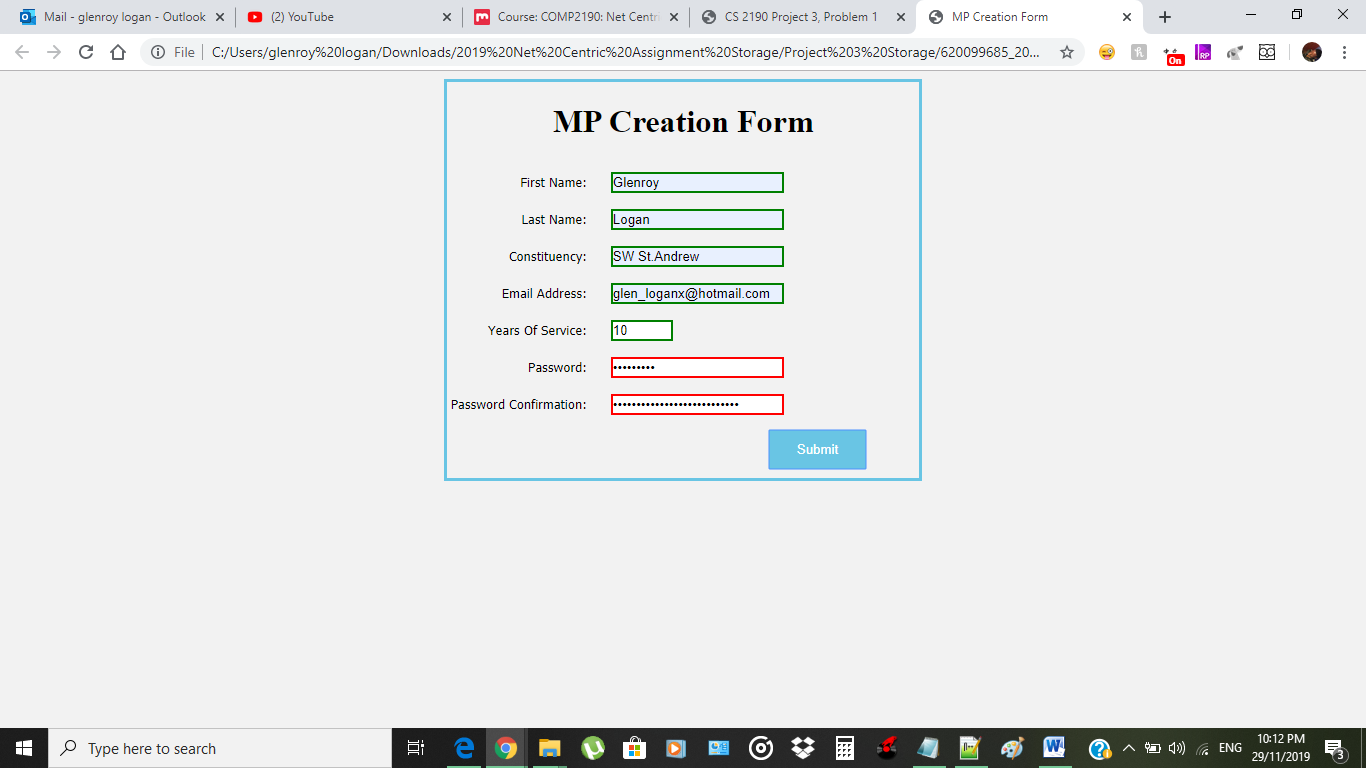
Test plan for validation input



Screenshot 1 Checks the validation to see if the fields listed are empty and if so will display a red message indicating to the user that more data needs to be entered

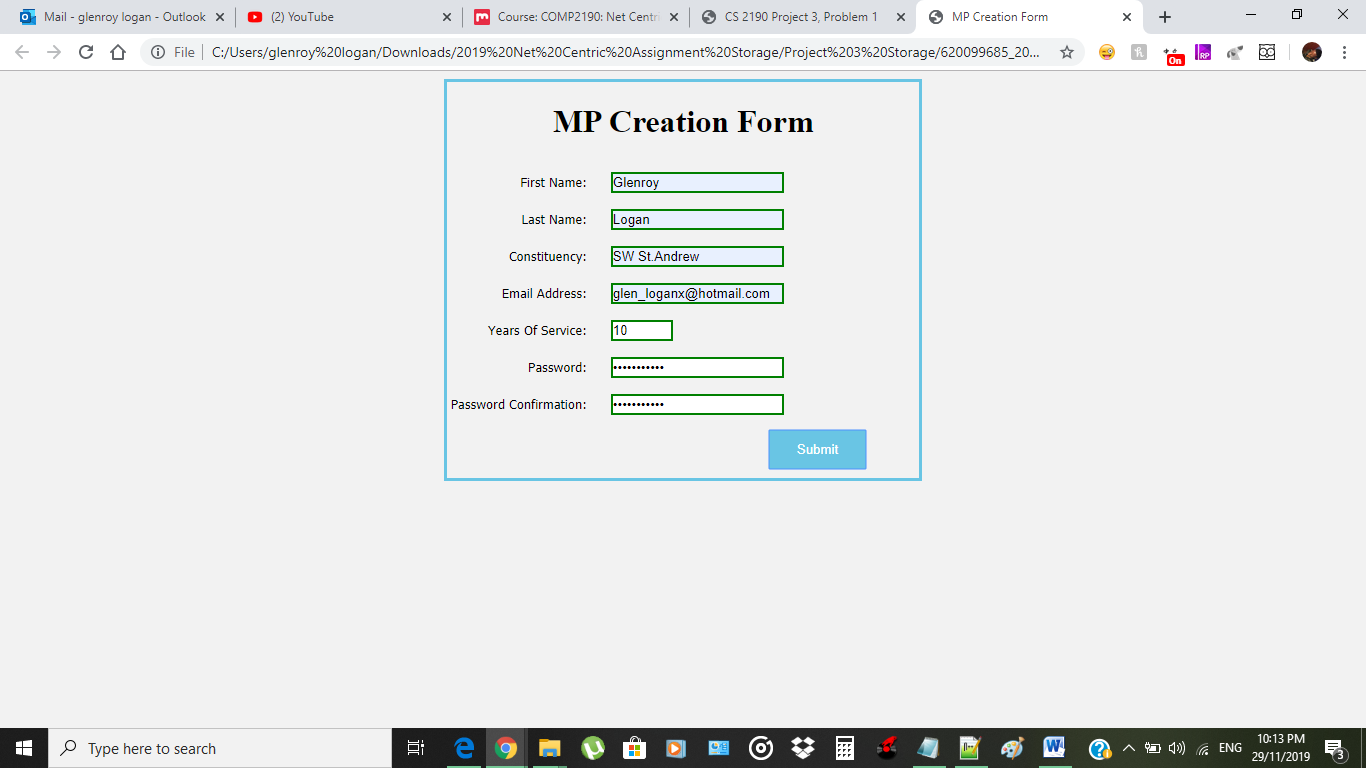


Screenshot 2 Shows what happens when a user inputs valid data but still has not completely filled out the remaining fields if the correct data type for each field is entered correctly then a green field is displayed to indicate to the user that they have entered the data correctly



In Screenshot 3 This shows where a user enters data into the respective fields but for the password section they have not matched the exact number of characters to gain confirmation to submit and as a result, the password boxes are marked as red to deny entry.

**Note: The password requirements were set up in such a way where the user was required to have a capital letter as well as a common letter along with a number and any symbol this was done so as to ensure that the passwords can be uniquely different as possible**



In the final screenshot this displays when all fields are filled and are done so correctly along with the user entering a password that matches the requirements that were previously stated above. When all fields meet the required entries then all boxes are displayed in green and access is granted to submit and create the MP Form.