Lab 2: Form Validation

In this lab you will perform the same field-by-field form validation you have done in previous courses on the client side with JavaScript, but this time on the server side. Your forms will include a number of different data types which will require different ways of validating their data.

Objectives & Outcomes

Successful completion of this activity will show that you can:

- Code server-side validation for form fields including: text, dates, numbers, emails, URLs, ranges.
- Integrate both client-side and server-side validation in the same forms.
- Create server-side validation code for arbitrary data rules.

Level of Effort

This activity should take approximately 4h to complete. It will require:

- 30m Research
- 5m Prep & Delivery
- 3h Work

If you find that this activity takes you significantly less or more time than this estimate, please contact me for guidance.

Instructions

This lab is similar to the last one: you will develop form validation solutions in both PHP and CFML, using the View classes you've already built. This lab will go in a `02_forms` folder.

Development

Build and develop two different types of forms and the server-side validation code to go with them:

- **User Authentication:** This should be a typical log-in form with a required email address and a password. The email address should look like an email address, and the password should be at least 8 characters long.
- **Data Entry:** The theme for the form is up to you, but there are some required elements. You must include: a date, an integer that can be positive or negative, a dollar value, an `http:` URL, at least one checkbox, a multi-select, and a set of radio buttons. There must also be at least one optional field of a data type of your choosing.

You may start with an existing form from a previous course if you like, as long as you extend it to include the elements above. Both of the forms should be implemented in both PHP and CFML.

The forms don't need to save any of the submitted data, but do need to include thorough server-side validation for each input field. Present a success message if everything looks okay, or inline error messages if anything looks incorrect.

It may be helpful to build form validation classes to keep track of the state of your forms.

Above & Beyond

It is not required for this assignment, but if you would like to go above and beyond you can add a layer of jQuery validation to your forms.

Git Commits & Video Reflection

The same rules about Git Commits and Reflection Videos apply for this Lab as for Lab 1. Review the Lab 1 documentation to ensure you meet all of the required points. Your required questions are mostly the same, with small differences:

1. What didn't go as well as you had hoped? Where did you run into problems with the lab?
2. What went better or was easier than you expected? What went right?
3. **When you compare client-side and server-side validation, what is the same and what is different?**
4. How do you think what you did in this lab is going to help you in your future school work, career, or life?

Where should you be?

By the end of this assignment you should feel much more comfortable with the syntax of PHP and CFML, as well as building classes in both languages. You should also see how similar the tasks of client-side and server-side validation can be.

Deliverables

Make sure your source code, assets, and reflection video URL are committed to the repository. Push your changes to the server before the start of the next lecture.