

JavaScript Form Validation Project v3

Project Background:

Validation Experts Inc. currently provides form validation services to a local audience of small businesses. In order to expand nationally and globally, Validation Experts Inc. has requested a javascript form validation library. They have provided a [basicvalidation.html](#) and [index.html](#) file that you should use, as-is, to test your form validation library. Your JavaScript code should be able to work with any html file that contains the proper form structure. Validation Experts Inc will be vetting your form validation library on multiple html forms to ensure consistency.

Functional Requirements (70%):

1. Given there are errors somewhere on a form, when the "Validate" input button for that form is clicked, then all error messages from this form will appear in the div with class "errors" immediately preceding the form.
2. Given the "Validate" input button for a form has been clicked, when errors exist, then they are displayed in an unordered list, with red text.
3. Given there are errors somewhere on a form, when the "Validate" input button for that form is clicked, then the form is not submitted.
4. Given there are no errors on the form, when the "Validate" input button for that form is clicked, then the form is submitted.
5. Given there are multiple forms on a page, when the "Validate" input button for any form is clicked, then only that form will run through validation.
6. Given an input of type "text" has a class of "required", and given nothing or only whitespace characters (spaces, tabs) has been entered into this input, when this form's validation runs, then an error appears in the correct location, and the error text says "Required fields must have a value that is not empty or whitespace."
7. Given an input of type "text" does not have a class of "required", and given nothing has been entered into this input, when this form's validation runs, then no error will appear from this input.
8. Given an input of type "text" has a class of "numeric", and given a non-numeric character has been entered into this input, when this form's validation runs, then an error appears in the correct location, and the error text says "Numeric fields must be a series of numbers."
9. Given an input of type "text" has a class of "required_size", and given the input has a minlength attribute, and given text has been entered with a length that does not exactly match the minlength attribute's value, when this form's validation runs, then an error appears in the correct location, and the error text says "Required_size field lengths must exactly match the minlength attribute of that field."

10. Given an input of type "text" has a class of "username", and given a non-alphanumeric character has been entered into this input, when this form's validation runs, then an error appears in the correct location, and the error text says "Username fields must contain only alphanumeric characters."
11. Given an input of type "text" has a class of "username", and given fewer than eight characters have been entered into this input, when this form's validation runs, then an error appears in the correct location, and the error text says "Username fields must contain at least 8 characters."
12. Given an input of type "text" has a class of "date", and given an entry that does not match the "XX/XX/XXXX" format has been entered into this input, when this form's validation runs, then an error appears in the correct location, and the error text says "Date fields must match the format of XX/XX/XXXX."
13. Given an input of type "text" has a class of "phone", and given an entry that does not match the "XXX-XXX-XXXX" format has been entered into this input, when this form's validation runs, then an error appears in the correct location, and the error text says "Phone fields must match the format of XXX-XXX-XXXX."
14. Given an input of type "text" has a class of "password", and an entry that does not contain one of: (uppercase letter, lowercase letter, number, special character) has been entered into this input, when this form's validation runs, then an error appears in the correct location, and the error text says "Password fields must contain one or more of each of the following types: uppercase letters, lowercase letters, numbers, special characters."
15. Given an input of type "text" has a class of "alphabetic", and given a non-alphabetic character has been entered into this input, when this form's validation runs, then an error appears in the correct location, and the error text says "Alphabetic fields must be a series of alphabetic characters."
16. Given an input of type "text" has multiple classes, when this form's validation runs, then all pertinent validation for this input is run:
 - a. For example, if an input has alphabetic and required_size, then both the alphabetic validation and required_size validation would need to run.

Non Functional Requirements (30%):

1. No global variables are used
2. No validation libraries are used
3. Console or alert statements must not be present
4. Descriptive code comments must be present on all functions
5. Camel case must be used for all JavaScript variable identifiers.
6. All functions are named, including callbacks in the event handlers.
7. All function names accurately describe the function