12/8/2020

Cohort 4.2

Techcareers

APP Testing Guide

A-Z Learn LMS Application

Contents

[Creating a course 2](#_Toc58380168)

[Pass case 2](#_Toc58380169)

[Fail cases 3](#_Toc58380170)

[HTML5 Validation demo 3](#_Toc58380171)

[JavaScript Validation demo 4](#_Toc58380172)

[Backend Validation demo 5](#_Toc58380173)

[Registering a user 6](#_Toc58380174)

[Pass case- Registering a student 6](#_Toc58380175)

[Pass case- Registering an Instructor 7](#_Toc58380176)

[Fail cases 8](#_Toc58380177)

[HTML5 Validation demo 8](#_Toc58380178)

[JavaScript Validation demo 9](#_Toc58380179)

[Backend Validation demo 11](#_Toc58380180)

Application Testing Instructions

We have implemented three layers of validation across our application. Those are,

1. HTML5 validation
2. JavaScript validation before data submission to backend
3. Backend validation before data submission to database

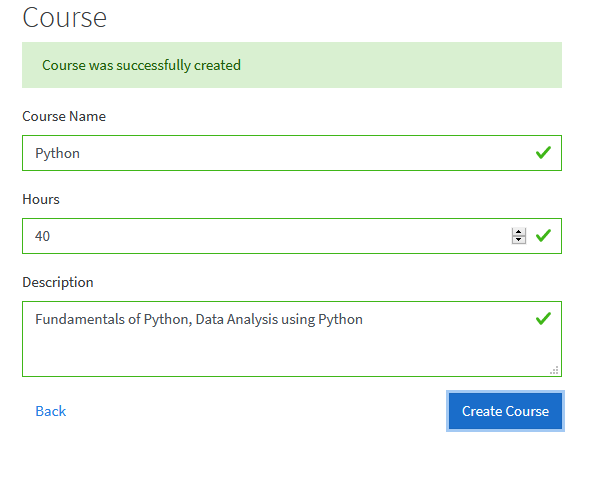
We have kept consistent validation pattern in the front-end views, and backend end points.

Following are some test cases to check the validation implementation across the three layers mentioned above. Note, same pattern can be followed to test any other front end view or backend end point.

# Creating a course

## Pass case

* Enter a course name (maximum 50 characters)
* Enter hours required for this course (0-999.99)
* Enter description for the course (maximum 250 characters)

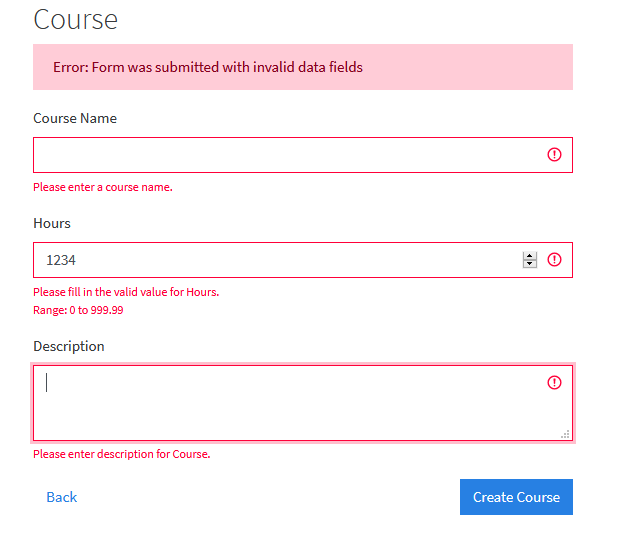


### Fail cases

### HTML5 Validation demo

* Leave Course Name field empty: this field is required
* Enter 1234 for Hours field: maximum range for this field is 0-999.99
* Leave Description field empty: this field is required

The app will show error messages detailing the errors, see below-



### JavaScript Validation demo

* Leave Course Name field empty: this field is required
* Enter 1234 for Hours field: maximum range for this field is 0-999.99
* Leave Description field empty: this field is required

If you try to create course with above data, error messages will be displayed due to HTML5 validation. See below to see validations in web inspector-



Tamper HTML5 validations in web inspector as shown in image below-

* Remove “required” from input for Course Name in web inspector tool
* Change type from “number” to “text” for input for Hours in web inspector tool
* Remove “required” from input for Description in web inspector tool

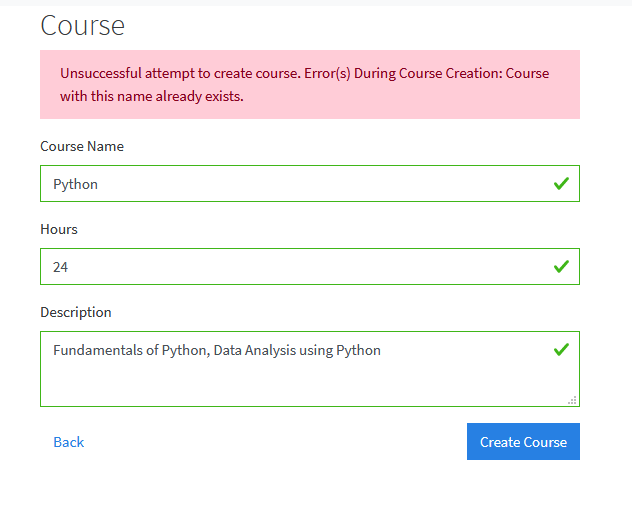
Now as you have tampered with HTML5 validation, error messages will not be displayed from HTML5 validation. But the second layer of validation- JavaScript validation layer will not allow the form to be submitted. It will show error.



### Backend Validation demo

* Try to create a course with valid parameters but keep the course name same as the course which is already created and exists in the database

The HTML5 validation and JavaScript validation will allow the form to be submitted as the data is valid but since it is a duplicate course name backend validation will not allow to create the course with same name. See below-



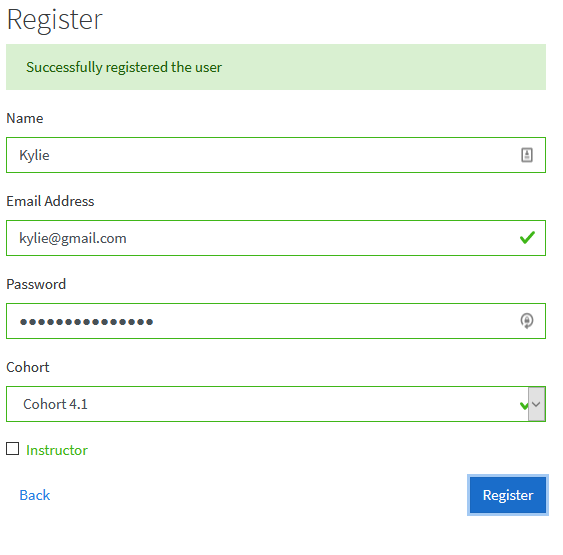
Similarly, all other features have three layers of validation in our application. You can follow similar steps for creating cohort, creating homework, updating cohort, updating homework, updating course.

## Registering a user

### Pass case- Registering a student

* Enter a student name (maximum 50 characters)
* Enter a valid email (maximum 250 characters, format should be a valid email format)
* Enter a password (minimum 8 characters, maximum 250 characters, password should have at least: 1 upper case, 1 lower case, 1 digit and 1 special character)
* Select a cohort ID for student from the drop-down list
* Keep the checkbox for Instructor unchecked as we are registering a student

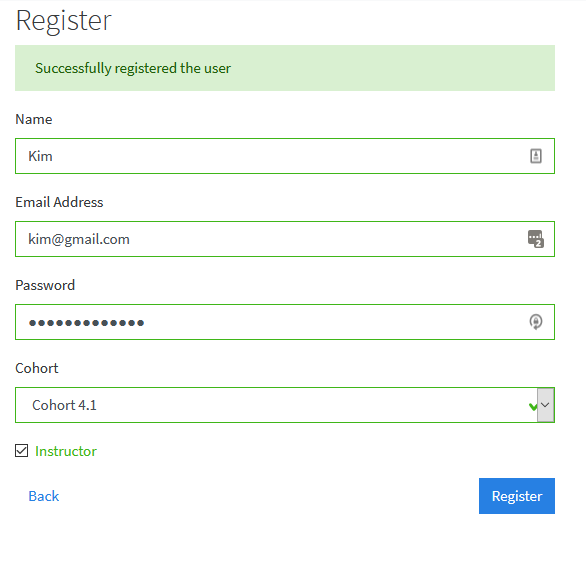
This will successfully register a student. See below-



### Pass case- Registering an Instructor

* Enter Instructor name (maximum 50 characters)
* Enter a valid email (maximum 250 characters, format should be a valid email format)
* Enter a password (minimum 8 characters, maximum 250 characters, password should have at least: 1 upper case, 1 lower case, 1 digit and 1 special character)
* Select a cohort ID for student from the drop-down list
* Check the checkbox for Instructor as we are registering an Instructor

This will successfully register an Instructor. See below-

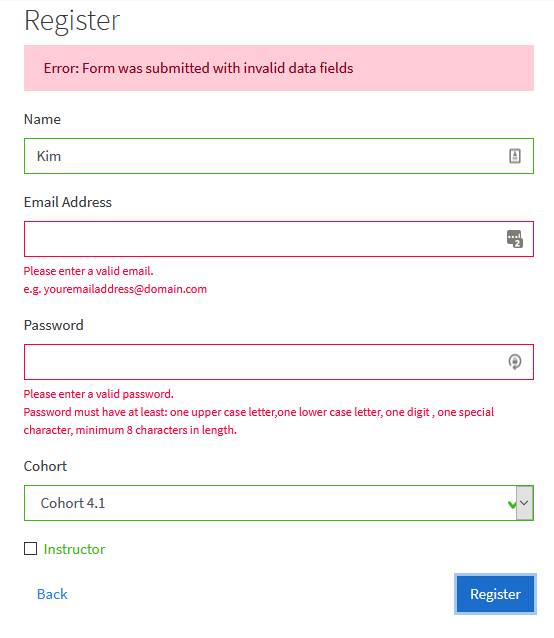


## Fail cases

### HTML5 Validation demo

* Enter a student name (maximum 50 characters)
* Leave the email field empty: This field is required, maximum 250 characters, format should be a valid email format
* Leave the password field empty: This field is required, minimum 8 characters, maximum 250 characters, password should have at least: 1 upper case, 1 lower case, 1 digit and 1 special character
* Select a cohort ID for student from the drop-down list
* Keep the checkbox for Instructor unchecked as we are registering a student

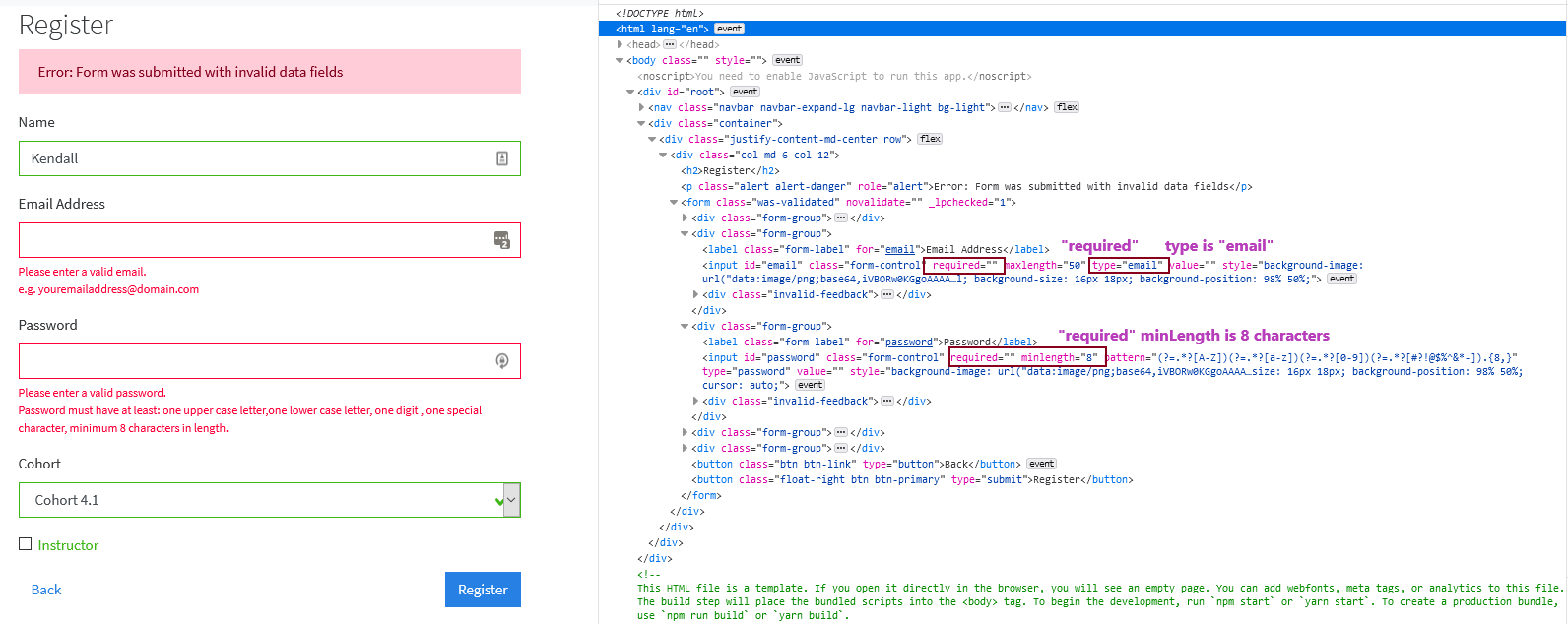
The app will show error messages detailing the errors because some required fields are missing. See below-



### JavaScript Validation demo

* Enter a student name (maximum 50 characters)
* Leave the email field empty: This field is required, maximum 250 characters, format should be a valid email format
* Leave the password field empty: This field is required, minimum 8 characters, maximum 250 characters, password should have at least: 1 upper case, 1 lower case, 1 digit and 1 special character
* Select a cohort ID for student from the drop-down list
* Keep the checkbox for Instructor unchecked as we are registering a student

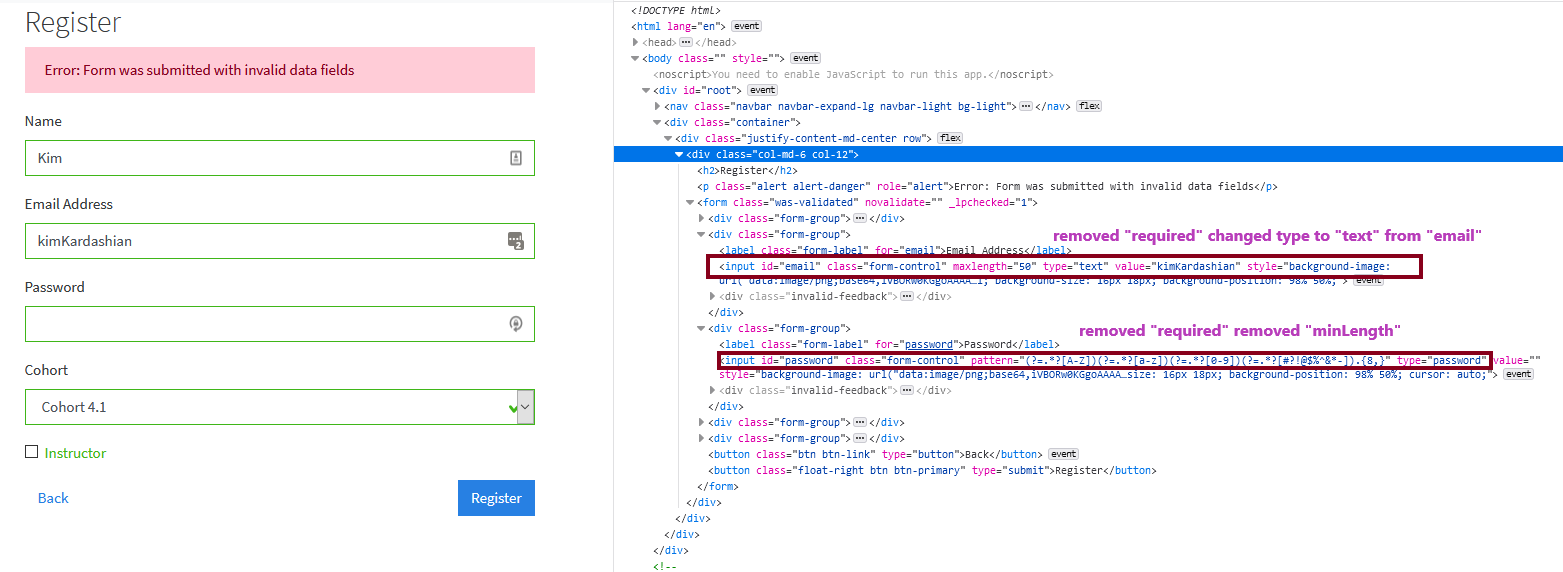
If you try to register student with missing data as above, error messages will be displayed due to HTML5 validation. See below to see validations in web inspector-



Tamper HTML5 validations in web inspector as shown in image below-

* Keep the inputs as in image above
* Remove “required” from input for Email Address in web inspector tool
* Change type from “email” to “text” for input for Email Address in web inspector tool
* Remove “required” from input for Password and remove “minLength” in web inspector tool

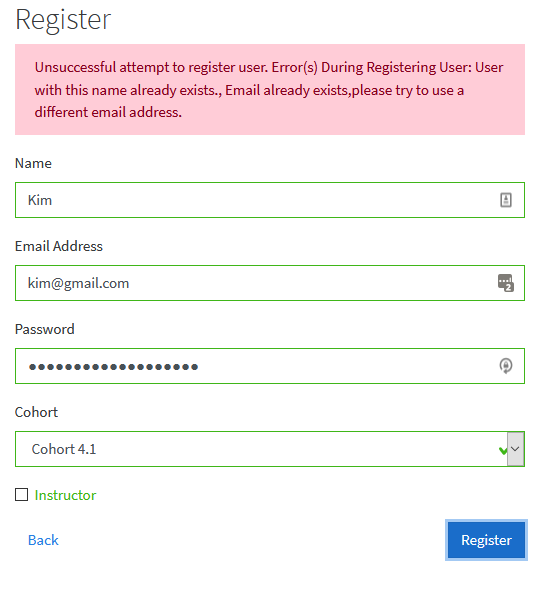
Now as you have tampered with HTML5 validation, error messages will not be displayed from HTML5 validation. But the second layer of validation- JavaScript validation layer will not allow the form to be submitted. It will show error. See below-



### Backend Validation demo

* Try to register a user with valid parameters but keep the Name same as the username which is already created and exists in the database

The HTML5 validation and JavaScript validation will allow the form to be submitted as the data is valid but since it is a duplicate username, backend validation will not allow to register the user with same name. See below-



**End of document**