npm install --save express-validator

Cài thêm Connect-Flash vì trong đây sử dụng

$ npm install connect-flash express-session

const flash = require("connect-flash");

const session = require("express-session");

app.use(session({ secret: "xHjnM5sTJungtin", resave: true, saveUninitialized: false }));

app.use(flash());

## Hiển thị Form

/\* Route \*/

app.get("/", (req, res) => {

    const errObj = req.flash("errors");

    res.render("basic", {

        errors: errObj[0], // Vì bất giá trị nào truyền qua flash đều sẽ nằm bên trong []

    });

});

<div class="container">

    <div class="row">

        <div class="col-12 col-md-8 mx-auto mt-3">

            <h1>Xin chào</h1>

            <form action="/validation"

                  method="post">

                <div class="form-group">

                    <label>Email</label>

                    <input type="text"

                           name="email"

                           class="form-control {{#if errors.email}}is-invalid{{/if}}">

                    {{#if errors.email}}

                    <div class="invalid-feedback">

                        {{errors.email}}

                    </div>

                    {{/if}}

                </div>

{{!-- Còn nữa ở giữa --}}

                <div class="form-group">

                    <label>Re-password</label>

                    <input type="password"

                           name="re-password"

                           class="form-control {{#if errors.re-password}}is-invalid{{/if}}">

                    {{#if errors.re-password}}

                    <div class="invalid-feedback">

                        {{errors.re-password}} {{!-- chỗ này hyphen vì lấy từ input name="" --}}

                    </div>

                    {{/if}}

                </div>

                <button type="submit"

                        class="btn btn-success">Submit</button>

            </form>

        </div>

    </div>

</div>

## Validating

app.post(

    "/validation",

    [

        body("email").isEmail().withMessage("không phải là email"),

        body("username")

            .notEmpty()

            .withMessage("không được để trống")

            .isLength({ min: 8 })

            .withMessage("Phải dài ít nhất 8 chars"),

        body("password")

            .notEmpty()

            .withMessage("không được để trống")

            .isLength({ min: 8 })

            .withMessage("Phải dài ít nhất 8 chars"),

        body("re-password")

            .notEmpty()

            .withMessage("không được để trống")

            .custom((value, { req }) => {

                if (value !== req.body.password)

                    throw new Error("Re-password phải trùng với lại password");

                return true; // phải return True chứ không là nó cho fail luôn

            }),

    ],

    (req, res) => {

        const rs = validationResult(req);

        if (rs.errors.length > 0) {

            /\* Filter Errors for View \*/

            const errObj = {};

            rs.errors

                .filter((err) => {

                    return err.location === "body";

                })

                .forEach((err) => {

                    errObj[err.param] = `${err.param.charAt(0).toUpperCase() + err.param.slice(1)}

                        ${err.msg.toLowerCase()}`;

                });

            req.flash("errors", errObj);

            res.redirect("/");

            return;

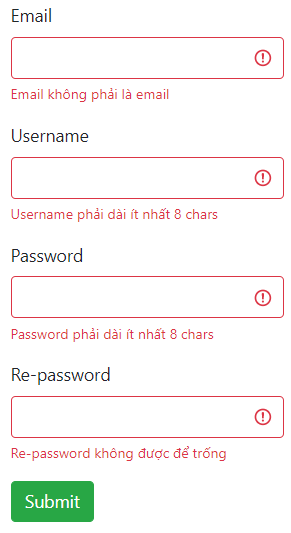
        }

        console.log("Không có lỗi gì cả! Thành công rồi <3");

        res.end();

    }

);

Kết quả : 

Validating Param

/\* Validate path \*/

app.get(

    "/students/:id",

    [

        param("id")

            .isInt()

            .withMessage("phải là số")

            .custom((value) => {

                // query trong database

                const existedIDs = [1, 2, 4, 5, 6, 7];

                if (!existedIDs.includes(parseInt(value)))

                    throw new Error("Không tìm thấy Student");

                return true;

            }),

    ],

    (req, res) => {

        const rs = validationResult(req);

        if (rs.errors.length > 0) {

            throw new Error(`404 Không tìm thấy student với id : ${req.params.id}`);

        }

        res.end();

    }

);