**LAB GUIDE 09: ANGULARJS**

**form VALIDATE, json data & storage**

**MỤC TIÊU**

Kết hợp giữa angularJS form và localStorage tạo trang web động đơn giản có các chức năng thêm , xóa dữ liệu

Trong đó

**localStorage** là nơi lưu trữ dữ liệu lâu dài cục bộ tại máy client ( máy cá nhân )

**sessionStorage** là lưu trữ tạm thời theo phiên

có 4 phưng thức cơ bản cần nắm vứng trong đối tượng **Storage** gồm

**localStorage.setItem(‘key’,’value’)** cho phép tạo một key và lưu trữ value vào bộ nhớ

**localStorage.getItem(‘key’)** cho phép lấy ra dữ liệu theo key đã lưu trữ truocs đó

**localStorage.removeItem(‘key’)** cho phép xóa một key đã lưu trước đó

**localStorage.clear()** xóa toàn bộ các dữ liệu đã lưu trữ trong storage

# BÀI TẬP HƯỚNG DẪN 1

# Tạo form đăng nhập với email và password

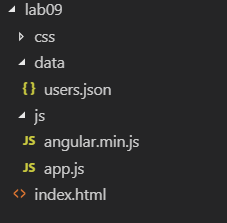
# Validate dữ liệu như bài lab07 ( Tham khảo code lab08)

# Khi nhập đủ và đúng thông tin thì lưu vào sessionStorage

# Sau đó hiển thị thông tin đã đăng nhập lên màn hình, thông tin này lấy từ sessionStorage

**Bước 1:** Tạo thư mục dự án **lab09** có cấu trúc như lab08, **có thêm thư mục data và trong đó có file users.json lưu trữ mảng đối tượng các người dùng**

Cấu trúc thư mục có dạng



**Bước 2:** File **index.html** có nội dung như sau

<!DOCTYPE html>

<html ng-app="app" ng-controller="AppCtrl">

<head>

<meta charset="UTF-8">

<title>Demo angular JS App</title>

<!-- Latest compiled and minified CSS -->

<link rel="stylesheet" href="https://netdna.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">

<link rel="stylesheet" href="css/style.css">

<script src="js/angular.min.js"></script>

<script src="js/app.js"></script>

</head>

<body>

<div class="container">

<div class="col-md-4 col-md-offset-4">

<!-- Thông báo tin nhắ đăng nhập thành công và thông tin người dùng -->

<div class="alert alert-success" ng-show="loginInfo">

<button type="button" class="close" data-dismiss="alert"ng-click="logout()">&times;</button>

<p>{{message}}</p>

<p>Xin chào {{loginInfo.name}}</p>

<p>Email {{loginInfo.email}}</p>

</div>

<div class="alert alert-danger" ng-show="!loginInfo && message">

<p>{{message}}</p>

</div>

<!-- form đăng nhập chỉ hiển thị khi chưa đăng nhập -->

<form action="" method="POST" name="MyForm" ng-show="!loginInfo">

<legend>Form Đăng nhập</legend>

<div class="form-group">

<label for="">Dịa chỉ email</label>

<input type="email" class="form-control" name="email" required ng-model="login\_email" placeholder="NHập địa chỉ email">

</div>

<div class="form-group">

<label for="">Mật khẩu</label>

<input type="password" class="form-control" name="password" required ng-model="login\_pass" placeholder="NHập mật khẩu">

</div>

<button type="button" ng-click="login()" class="btn btn-primary btn-block" ng-disabled="MyForm.email.$error.required || MyForm.email.$error.email || MyForm.password.$error.required">Login now</button>

</form>

</div>

</div>

</body>

</html>

**Bước 3:** File users.json có nội dung sau ( Dah sách các tài khoản )

Khi đăng nhập thì sẽ kiểm tra thông tin trong danh sách này nếu tồn tại thì đăng nhập thành công

[

{

"name" : "Nguyễn Tiến Hùng",

"email": "hungnt@gmail.com",

"password" : "123456"

},

{

"name" : "Phạn Minh Thắng",

"email": "thangpm@gmail.com",

"password" : "123456"

},

{

"name" : "Bùi Quốc Khánh",

"email": "khanhbq@gmail.com",

"password" : "123456"

},

{

"name" : "Cao Trung Hiếug",

"email": "hieuct@gmail.com",

"password" : "123456"

}

]

**Chú ý : tại nút login now gọi sự kiện ng-click=”login()”**

<button type="button" ng-click="login()" class="btn btn-primary btn-block" ng-disabled="MyForm.email.$error.required || MyForm.email.$error.email || MyForm.password.$error.required">Login now</button>

**Chú ý: nút x có sự kiện ng-click=”logout()” thoát tài khoản sau khi login**

<div class="alert alert-success" ng-show="loginInfo">

<button type="button" class="close" data-dismiss="alert"ng-click="logout()">&times;</button>

<p>{{message}}</p>

<p>Xin chào {{loginInfo.name}}</p>

<p>Email {{loginInfo.email}}</p>

</div>

**Bước 4: Viết mã trong file app.js giống như ở lab08 tuy nhiên hàm login() cần phải kiển tra dữ liệu trong file users.json nên có cod sau**

var app = angular.module('app', []);

app.controller('AppCtrl', function ($scope, $http) {

$scope.message = ""; // tin nhắn thông báo lỗi rỗng // Giữ nguyên lab08

// lấy thông tin đăng nhập đã lưu trong sessionStorage

var u = sessionStorage.getItem('login\_inffo');

if(u){

// chuyển chỗi json sang kiểu đối tượng trong javascript

$scope.loginInfo = angular.fromJson(u);

}

// cần lấy thông tin người dung trong file data/users.jdon

$http.get('data/users.json').then(function (res) {

// hàm này sẽ có 2 tham số là email và mật khâu

function check\_login(email, password) {

for (var i = 0; i < res.data.length; i++) {

// nếu đúng thông tin như trong data/users.json thì trả về đối tượng người dùng

if (res.data[i].email == email && res.data[i].password == password) {

return res.data[i];

}

}

return false; // trả về false nếu không có thông tin nào đúng

}

//Sự kiện login khi click

$scope.login = function () {

// gọi lại hàm check\_login truyền vào email, password

var user = check\_login($scope.login\_email,$scope.login\_pass);

if (user) {

// lưu thông tin vào sessionStorage

// vì storage chỉ lưu dữ liệu chuỗi nên cần chuyển đối tượng => chuỗi jsson

sessionStorage.setItem('login\_inffo', angular.toJson(user));

$scope.message = "Đăng nhập thành công"; // Giữ nguyên lab08

$scope.loginInfo = user; // gán thông tin với đối tượng user mới

} else {

$scope.message = "Tên đăng nhập hoặc mật khẩu không đúng"; // Giữ nguyên lab08

}

}

});

// phương thức thoát, khi click vào nút x

// Giữ nguyên lab08

$scope.logout = function () {

sessionStorage.removeItem('login\_inffo'); // xóa storage theo key

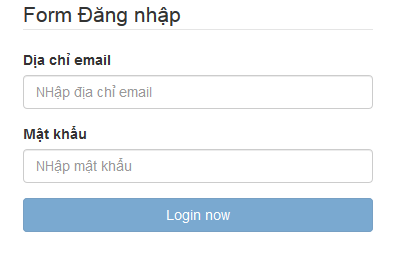
$scope.message = "Thoát tài khoản thành công";

$scope.loginInfo = null;

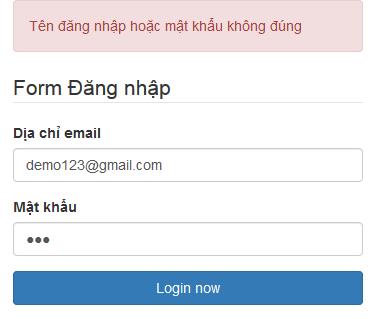
}

});

**Kết quả của bước 1,2,3,4**



**Khi người dùng nhập email hoặc mật khẩu không chính xác thông báo lỗi**



**Khi người dùng đăng nhập đúng thì form sẽ ẩn đi và thông tin sẽ hiện lên**



**Các chức năng còn lại giống lab08**