

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Mô hình kết nối giữa frontend và backend
- ✓ Xây dựng FrontEnd với Ajax
- ✓ Xây dựng FrontEnd bằng AngularJS

PHẦN I

Bài 1 (3 điểm)

Xây dựng Front end cho trang quản lý blog có giao diện như sau

The screenshot shows a web application for managing blog posts. The header contains links: FPT PoLy, Trang Chủ, Quản lý bài viết, and Đăng xuất. The main content area is titled 'Quản lý bài viết' and contains a table with the following data:

#	Tiêu đề	Ngày cập nhật	Thao tác
1	FPT Poly Technic	2020-10-30	Xóa Sửa
2	C# - ngôn ngữ vẫn được ưa thích	2020-10-30	Xóa Sửa
3	Tại sao là ngôn ngữ NodeJS	2020-10-30	Xóa Sửa
4	5 lý do sử dụng VueJS	2020-10-30	Xóa Sửa

To the right of the table is a form titled 'Thêm - sửa bài viết'. It has two input fields: 'Tiêu đề bài viết' and 'Nội dung bài viết'. Below the fields is a green button labeled 'Lưu'.

Bước 1: Thiết kế giao diện dạng html (sinh viên sử dụng bootstrap để thiết kế)

Bước 2: Sử dụng REST API bài Lab05 để làm BackEnd

Bổ sung thêm header cho server như sau

```
app.use((req, res, next) => {  
  res.setHeader('Access-Control-Allow-Origin', '*');  
  res.setHeader('Access-Control-Allow-Methods', 'OPTIONS, GET, POST, PUT, PATCH, DELETE');  
  res.setHeader('Access-Control-Allow-Headers', 'Content-Type, Authorization');  
  next();  
});
```

```
});
```

Bước 3: Gọi và nhận dữ liệu từ Backend trả về, hiển thị trên giao diện

```
function getAllPost(){
    var request=new XMLHttpRequest();
    request.open('GET','http://127.0.0.1:3000/blog/posts');
    request.send();
    //xu ly du lieu
    request.onload=function(){
        if (this.readyState == 4 && this.status == 200) {
            var data=JSON.parse(request.responseText);
            console.log(data);
            let tr="";
            data.posts.forEach(s => {
                tr+=`<tr>
                <th scope="row">${s.postId}</th>
                <td>${s.title}</td>
                <td>${s.create_date.substr(0,10)}</td>
                <td><a href="#">Xóa</a> | <a href="#">Sửa</a></td>
                </tr>`;
            });
            document.querySelector('#listPost').innerHTML=tr;
        }
        else{

        }
    }
}
```

Bài 2 (2 điểm)

Thêm bài viết (post) qua method POST

Bước 1: Gán hàm xử lý thêm (addPost) cho sự kiện onclick của nút lưu

Bước 2: Viết hàm addPost:

```
function addPost(){
    //lấy dữ liệu trên form thêm
```

```
var title=document.querySelector('#title').value;
var content= document.querySelector('#content').value;
//tạo biến json chứa data cần thêm
var data = JSON.stringify({"title":title , "content": content});
var request=new XMLHttpRequest();
request.open('POST','http://127.0.0.1:3000/blog/posts');
//gán thêm header
request.setRequestHeader("Content-Type","application/json; charset=UTF-8");
//xử lý kết quả trả về
request.onload=function(){
    if (this.readyState == 4 && this.status == 201) {
        var data=JSON.parse(request.responseText);
        getAllPost();//load lại danh sách bài viết
    }
    else{

    }
}
request.send(data);
}
```

PHẦN II

Bài 3 (2 điểm)

Xây dựng Front end cho trang quản lý blog có giao diện như sau bằng angularjs

FPT Poly
Trang Chủ
Quản lý bài viết
Đăng xuất

Quản lý bài viết

#	Tiêu đề	Ngày cập nhật	Thao tác
1	New MV	20-08-2020	Xóa Sửa
2	Phong Tom	19-08-2020	Xóa Sửa

Thêm - sửa bài viết

Tiêu đề bài viết

Nội dung bài viết

Lưu

Bước 1: Khai báo sử dụng angularjs trong trang html

```

13  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.5/angular.min.js"></script>
14  </head>
15  <body>
16  > <header> ...
37  </header>
38  <main class="container-fluid" ng-app="myApp">
39  |   <h2>Quản lý bài viết</h2>
40  |   <div class="row" ng-controller="myController">
41  | >   <div class="col-9 article_customer"> ...
56  |   </div>
57  | >   <div class="col-3 aside_customer"> ...
71  |   </div>
72  | </div>
73  </main>
74  <!-- Optional JavaScript -->
75
76  <script>
77  |   var app=angular.module("myApp", []);
78  |   app.controller("myController",function($scope){
79  |
80  |   });
81  </script>

```

Bước 2: Dùng \$http service đọc danh sách các bài viết hiện hành và hiển thị lên giao diện

```
76 <script>
77   var app=angular.module("myApp", []);
78   app.controller("myController",function($scope){
79     app.controller("myCtrl", function($scope, $http) {
80       $http.get("http://127.0.0.1:3000/blog/posts").then(function(response) {
81         $scope.posts= response.data.posts;
82       });
83     });
84   });
85 </script>
```

```
<table class="table">
  <thead>
    <tr>
      <th scope="col">#</th>
      <th scope="col">Tiêu đề</th>
      <th scope="col">Ngày cập nhật</th>
      <th scope="col">Thao tác</th>
    </tr>
  </thead>
  <tbody>
    <tr ng-repeat='post in posts'>
      <th scope="row">{{post.postId}}</th>
      <td>{{post.title}}</td>
      <td>{{post.create_date.substr(0,10)}}</td>
      <td><a href="#">Xoá</a> | <a href="#">Sửa</a></td>
    </tr>
  </tbody>
</table>
```

Bài 4 (2 điểm)

Thêm bài viết (post) qua method POST

Bước 1: Gán hàm xử lý thêm (addPost) cho sự kiện onclick của nút lưu

Bước 2: Viết hàm addPost:

```
81 <script>
82     var app=angular.module("myApp", []);
83     app.controller("myController", function($scope, $http) {
84         var title='';
85         var content='';
86         $http.get("http://127.0.0.1:3000/blog/posts").then(function(response) {
87             $scope.posts= response.data.posts;
88         });
89         $scope.addPost=function(){
90             //tạo biến json chứa data cần thêm
91             var data = JSON.stringify({"title":title , "content": content});
92             $http.post('http://127.0.0.1:3000/blog/posts',data).then(function(response) {
93                 title='';
94                 content='';
95             });
96         }
97     });
98 </script>
```

Bài 5 (1 điểm)

Sinh viên tự hoàn thành 2 chức năng xoá và sửa.

*** Yêu cầu nộp bài:

SV nén file (hoặc share thư mục google drive) bao gồm các yêu cầu đã thực hiện trên, nộp LMS đúng thời gian quy định của giảng viên. KHÔNG NỘP BÀI COI NHƯ KHÔNG CÓ ĐIỂM.

---Hết---