



# **LẬP TRÌNH JAVASCRIPT**

## **BÀI 7: GIỚI THIỆU JAVASCRIPT FRAMEWORK ( ANGULARJS)**

❑ **Backbone.js**, một thư viện JavaScript cho phép các nhà phát triển có thể thêm cấu trúc vào các ứng dụng và trình bày dữ liệu như các Model.



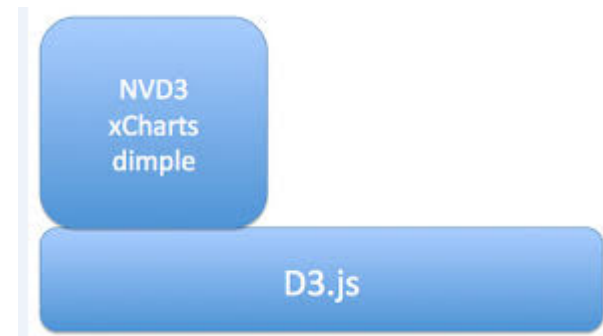
❑ **Bootstrap**, một framework thiết kế web dự định sẽ được sử dụng kết hợp với JQuery.



❑ **Enyo**, một framework JavaScript hướng đối tượng có thể được sử dụng để xây dựng các ứng dụng HTML5/CSS.



❑ **D3** (Data-Driven Documents), một thư viện JavaScript xử lý đồ họa vector hoạt động trong trình duyệt mà không cần thêm plug-in.



❑ **Ember.js**, một framework JavaScript rất có tiềm năng cho việc phát triển các ứng dụng MVC với chức năng phong phú.



❑ **Emscripten**, một trình biên dịch có thể chuyển đổi C++ thành asm.js, một tập hợp JavaScript có khả năng tối ưu hóa cao của Mozilla



❑ Node.js là một thư viện của JavaScript, hơn thế nữa node.js là cả một platform với 4 sức mạnh:

- ❖ Ngôn ngữ phổ biến
- ❖ Tốc độ nhanh (nhờ V8)
- ❖ Paradigm mới (non blocking I/O)
- ❖ Nhiều thư viện, dễ cài (nhờ npm)



❑ ...

- ❑ Dự án AngularJS được bắt đầu từ năm 2009, do lập trình viên Misko Hevery tại Google
- ❑ Với AngularJS, Misko đã rút ngắn số dòng code front-end từ 17000 dòng còn chỉ khoảng 1500
- ❑ Là một framework hỗ trợ lập trình viên xây dựng ứng dụng web một cách có hệ thống

## ❑ Viết chương trình Hello World

- ❖ Tạo thư mục theo cấu trúc sau
- ❖ Source code mẫu cho index.html

```
helloworld
-- index.html
-- scripts
---- controllers.js
```

```
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.0.7/angular.min.js"></script>
<script src="scripts/controllers.js"></script>
</head>
<body ng-app>
  <h1>AngularJS 101</h1>
  <h3>by <span id="author">HungNQ</span></h3>
  <div ng-controller="HelloCtrl">
    Ban ten la: <input type="text" ng-model="name"/>
    <p>Xin chào bạn! <b>{{ name }}</b><br/>Fpoly luôn vững mạnh</p>
  </div>
</body>
</html>
```

## ❑ Viết chương trình Hello World

❖ Source code controllers.js

```
function HelloCtrl($scope) {  
  $scope.name = "Polytechnic";  
}
```

❖ CSS

```
h1 {  
  color: red;  
}  
#author {  
  color: blue;  
}
```



## □ Viết chương trình Hello World

- ❖ Kết quả:
- ❖ (Có thể đăng ký và chạy chương trình trên <http://jsfiddle.net>)

**AngularJS 101**

by **HungNQ**

Ban ten la:

Xin chao ban **Polytechnic**  
Chúc Fpoly luôn vững mạnh

JSFIDDLE α
Run
Update
Fork
TidyUp
JSHint
Collaboration
Share ▼

### Frameworks & Extensions

AngularJS 1.1.1
No wrap - in <head>

### Fiddle Options

### External Resources

### Languages

### Ajax Requests

### Legal, Credits and Links

JSFiddle Job Board:

Front-End Engineer
Full-time at Positionly
Warsaw, Poland

```

1 <body ng-app>
2   <h1>AngularJS 101</h1>
3   <h3>by <span
4     id="author">HungNQ</span></h3>
5   <div
6     ng-controller="HelloCtrl">
7     Ban ten la: <input
8       type="text" ng-model="name"/>
9     <p>Xin chao ban <b>{{
10      name }}</b><br/>Chúc Fpoly luôn
11      vững mạnh</p>
12   </div>
13 </body>

```

HTML

```

1 h1 {
2   color: red;
3 }
4 #author {
5   color: blue;
6 }

```

CSS

```

1 function HelloCtrl($scope) {
2   $scope.name = "Polytechnic";
3 }

```

JavaScript

## AngularJS 101

by **HungNQ**

Ban ten la:

Xin chao ban **Polytechnic**

Chúc Fpoly luôn vững mạnh

Result

## ❑ **ng-app** là gì?

- ❖ **ng-app** là nơi bạn muốn đánh dấu cho AngularJS biết ứng dụng sẽ bắt đầu từ đâu
- ❖ **ng-app** thường nằm trong body tag hoặc div

## ❑ **ng-controller** là gì?

- ❖ AngularJS dùng mô hình MVC, mỗi controller sẽ có nhiệm vụ quản lý những model bên trong nó.

## ❑ **\$scope là gì?**

- ❖ Kí hiệu \$ được dùng để phân biệt thành phần do AngularJS tạo ra, như ng- namespace vậy. Việc \$scope được đưa vào controller là do dependency injection. Khi bạn dùng AngularJS với jQuery, \$ sẽ được wrap bằng angular.element.

## ❑ **ng là gì?**

- ❖ Là namespace cho AngularJS (hoặc data-ng-)

## ❑ **Cú pháp:**

- ❖ AngularJS sử dụng cú pháp: `{{data}}` để hiển thị giá trị của data trong DOM.

## ❑ **Lưu ý:**

- ❖ Kết quả mà data bạn cập nhật sẽ được hiển thị ngay tại chỗ chứ không cần thiết phải reload page

## □ Cú pháp:


- ❖ Cú pháp khai báo trong AngularJS giống cú pháp khai báo trong javascript. Tuy nhiên, khai báo AngularJS có sự khác biệt sau:
  - Ngữ cảnh: Javascript dựa vào **window**. AngularJS dựa vào đối tượng **scope**
  - Trong Javascript, những giá trị undefined được phát hiện bởi ReferenceError hoặc TypeError. Trong AngularJS, biểu thức lờ đi giá trị undefined và null
  - Bộ lọc: sử dụng filters trong biểu thức để định dạng dữ liệu trước khi biểu diễn nó






## □ Cú pháp:

- ❖ Cú pháp khai báo trong AngularJS giống cú pháp khai báo trong javascript. Tuy nhiên, khai báo AngularJS có sự khác biệt sau:


(<https://docs.angularjs.org/guide/expression>)


- Không kiểm soát dòng lệnh
- Không khai báo function
- Không tạo **RegExp**, ngoại trừ ng-pattern
- Không tạo **Object** với toán tử **new**
- Không toán tử thao tác Bit, dấu phẩy (,), void

 JSFIDDLE α

▶ Run  Update  Set as base  TidyUp  JSHint  Collaboration

Frameworks & Extensions

AngularJS 1.1.1 

No wrap - in <body> 


Fiddle Options

External Resources

Languages

Ajax Requests

Legal, Credits and Links

1 


2 <div class='container' ng-app>


3 <input type='text' ng-


4 model='userName' />

5 <p>Hello {{userName}}!</p>


6 </div>

1 

1 

HungNQ 

Hello HungNQ!

 Keyboard shortcuts

16



JSFIDDLE α

▶ Run
 
 Update
 Fork
 ☆ Set as base
 ✓ TidyUp
 ✓ JSHint
 Collaboration

**Frameworks & Extensions**

AngularJS 1.1.1

No wrap - in <body>

Fiddle Options

External Resources

Languages

Ajax Requests

Legal, Credits and Links

```

1 <div class='container' ng-app>
2   <div ng-controller='myController'>
3     <form ng-submit='addNewUser()'>
4       <input type='text' ng-model='NewUser' />
5       <input type="submit" />
6     </form>
7     <p>Hello {{NewUser}}!</p>
8     <p>User Array: {{userList}}</p>
9   </div>
10 </div>

```

HTML

```

1 // Javascript
2 var myController = function($scope){
3   $scope.userList = ['CuongCH'];
4
5   $scope.addNewUser = function(){
6     $scope.userList.push($scope.NewUser);
7   };
8 }

```

JavaScript

1

CSS

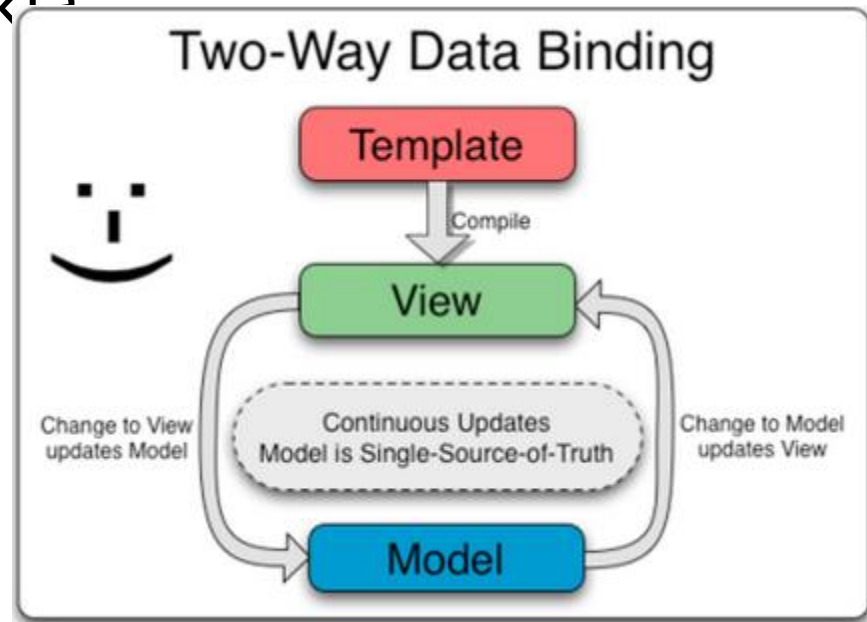
HungNQ

Submit

Hello HungNQ!

User Array:  
["CuongCH","HungNQ"]

Quá trình đồng bộ luôn xảy ra giữa model và view. Mọi sự thay đổi dữ liệu trong lớp này, ngay lập tức được cập nhật vào lớp kia.



## ❑ 4 cách:

- ❖ Trực tiếp: dựa vào **DOM**
- ❖ AngularJS sử dụng cú pháp: **{{data}}** để hiển thị giá trị của data trong DOM.
- ❖ Dùng bộ lọc **filter**
- ❖ Các điều khiển **form**: điều hướng dữ liệu đầu vào của người dùng

- ❑ Sự kết hợp giữa AngularJS với web-socket là: mọi thứ được cập nhật tức thời.
  - ❖ Một thao tác cập nhật ở server xảy ra, ngay tức thì, nó sẽ được AngularJS ghi nhận và cập nhật lại Model của mình. Tiếp tục, đến lượt View, nó cũng sẽ cập nhật lại dữ liệu hiển thị của nó theo Model

- ❑ Sự kết hợp giữa AngularJS với web-socket là: mọi thứ được cập nhật tức thời.
  - ❖ Một thao tác cập nhật ở server xảy ra, ngay tức thì, nó sẽ được AngularJS ghi nhận và cập nhật lại Model của mình. Tiếp tục, đến lượt View, nó cũng sẽ cập nhật lại dữ liệu hiển thị của nó theo Model



# **LẬP TRÌNH JAVASCRIPT**

## **BÀI 7: GIỚI THIỆU JAVASCRIPT FRAMEWORK ( ANGULARJS)**

❑ Firebase là một cơ sở dữ liệu online, realtime. Nó giúp bạn đọc, ghi dữ liệu trực tuyến:

❖ Cài đặt Firebase:

➤ Đưa dòng code sau vào phần <head>: **<script src='https://cdn.firebase.com/v0/firebase.js'></script>**

```
html>
  <head>
    <script src='https://cdn.firebase.com/v0/firebase.js'></script>
  </head>
  <body>
    <script>
      var myDataRef = new Firebase('https://tkz5am1ctvs.firebaseio-demo.com/');
    </script>
  </body>
</html>
```

## ❑ Truy cập dữ liệu FireBase:

- Đưa dòng code sau vào thẻ <script> trong vùng <body>
- **var myDataRef = new  
Firebase('https://tkz5am1ctvs.firebaseio-demo.com/');**

```
html>
  <head>
    <script src='https://cdn.firebase.com/v0/firebase.js'></script>
  </head>
  <body>
    <script>
      var myDataRef = new Firebase('https://tkz5am1ctvs.firebaseio-demo.com/');
    </script>
  </body>
</html>
```



## ❑ Ghi dữ liệu trong Firebase:

- ❖ Giả sử ta cần gửi một chat message, ta dùng dòng lệnh sau để ghi dữ liệu **myDataRef.set('User ' +**

```
<html>
<head>
  <script src='https://cdn.firebase.com/v0/firebase.js'></script>
  <script src='https://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js'></script>
</head>
<body>
  <input type='text' id='nameInput' placeholder='Name'>
  <input type='text' id='messageInput' placeholder='Message'>
  <script>
    var myDataRef = new Firebase('https://tkz5amlctvs.firebaseio-demo.com/');
    $('#messageInput').keypress(function (e) {
      if (e.keyCode == 13) {
        var name = $('#nameInput').val();
        var text = $('#messageInput').val();
        myDataRef.set('User ' + name + ' says ' + text);
        $('#messageInput').val('');
      }
    });
  </script>
</body>
</html>
```

## ❑ Ghi dữ liệu Object trong FireBase:

- ❖ Ta dùng dòng lệnh sau để ghi dữ liệu  
**myDataRef.set({name: name, text: text});**

```
<html>
  <head>
    <script src='https://cdn.firebase.com/v0/firebase.js'></script>
    <script src='https://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js'></script>
  </head>
  <body>
    <input type='text' id='nameInput' placeholder='Name'>
    <input type='text' id='messageInput' placeholder='Message'>
    <script>
      var myDataRef = new Firebase('https://tkz5amlctvs.firebaseio-demo.com/');
      $('#messageInput').keypress(function (e) {
        if (e.keyCode == 13) {
          var name = $('#nameInput').val();
          var text = $('#messageInput').val();
          myDataRef.set({name: name, text: text});
          $('#messageInput').val('');
        }
      });
    </script>
  </body>
</html>
```

## ❑ Ghi dữ liệu List:

- ❖ Ta dùng dòng lệnh sau để ghi dữ liệu  
**myDataRef.push({name: name, text: text});**

```
<html>
<head>
  <script src='https://cdn.firebase.com/v0/firebase.js'></script>
  <script src='https://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js'></script>
</head>
<body>
  <input type='text' id='nameInput' placeholder='Name'>
  <input type='text' id='messageInput' placeholder='Message'>
  <script>
    var myDataRef = new Firebase('https://tkz5am1ctvs.firebaseio-demo.com/');
    $('#messageInput').keypress(function (e) {
      if (e.keyCode == 13) {
        var name = $('#nameInput').val();
        var text = $('#messageInput').val();
        myDataRef.push({name: name, text: text});
        $('#messageInput').val('');
      }
    });
  </script>
</body>
</html>
```

## ❑ Đọc và hiển thị dữ liệu:

```
<script src='https://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js'></script>
</head>
<body>
  <div id='messagesDiv'></div>
  <input type='text' id='nameInput' placeholder='Name'>
  <input type='text' id='messageInput' placeholder='Message'>
  <script>
    var myDataRef = new Firebase('https://tkz5am1ctvs.firebaseio-demo.com/');
    $('#messageInput').keypress(function (e) {
      if (e.keyCode == 13) {
        var name = $('#nameInput').val();
        var text = $('#messageInput').val();
        myDataRef.push({name: name, text: text});
        $('#messageInput').val('');
      }
    });
    myDataRef.on('child_added', function(snapshot) {
      var message = snapshot.val();
      displayChatMessage(message.name, message.text);
    });
    function displayChatMessage(name, text) {
      $('#<div/>').text(text).prepend($('#<em/>').text(name+' : ')).appendTo($('#messagesDiv'));
      $('#messagesDiv')[0].scrollTop = $('#messagesDiv')[0].scrollHeight;
    };
  </script>
</body>
</html>
```

```

<!doctype html>
<html>
  <head>
    <script src='https://cdn.firebase.com/v0/firebase.js'></script>
    <script src='https://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js'></script>
    <link rel='stylesheet' type='text/css' href='../css/example.css'>
  </head>
  <body>
    <div id='messagesDiv'></div>
    <input type='text' id='nameInput' placeholder='Name'>
    <input type='text' id='messageInput' placeholder='Message'>
    <script>
      var myDataRef = new Firebase('https://tkz5am1ctvs.firebaseio-demo.com/');
      $('#messageInput').keypress(function (e) {
        if (e.keyCode == 13) {
          var name = $('#nameInput').val();
          var text = $('#messageInput').val();
          myDataRef.push({name: name, text: text});
          $('#messageInput').val('');
        }
      });
      myDataRef.on('child_added', function(snapshot) {
        var message = snapshot.val();
        displayChatMessage(message.name, message.text);
      });
      function displayChatMessage(name, text) {
        $('#<div/>').text(text).prepend($('#<em/>').text(name+' : ')).appendTo($('#messagesDiv'));
        $('#messagesDiv')[0].scrollTop = $('#messagesDiv')[0].scrollHeight;
      }
    </script>
  </body>
</html>

```

# FPT POLYTECHNIC



KẾT THÚC