BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**KHOA KỸ THUẬT – CÔNG NGHỆ**



**BÁO CÁO GIỮA KÌ**

**LẬP TRÌNH TIÊN TIẾN**

**TÊN ĐỀ TÀI**

**TÌM HIỂU VÀ ỨNG DỤNG AngularJS**

**GVHD:** ThS. Nguyễn Tấn Lộc

**SVTH:** Hồ Thái An - 1524801030031

Trần Hồ Bảo Hoàng – 1524801030037

**Lớp:** D15PM01

Bình Dương – 2019

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**KHOA KỸ THUẬT – CÔNG NGHỆ**



**BÁO CÁO GIỮA KÌ**

**LẬP TRÌNH TIÊN TIẾN**

**TÊN ĐỀ TÀI**

**TÌM HIỂU VÀ ỨNG DỤNG ANGULARJS**

**GVHD:** ThS. Nguyễn Tấn Lộc

**SVTH:** Hồ Thái An - 1524801030031

Trần Hồ Bảo Hoàng - 1524801030037

**Lớp:** D15PM01

Bình Dương – 2019

**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**KHOA KỸ THUẬT – CÔNG NGHỆ**

**NHẬN XÉT VÀ CHẤM ĐIỂM CỦA GIẢNG VIÊN**

Họ và tên giảng viên: **ThS. Nguyễn Tấn Lộc**

Tên đề tài: **Tìm hiểu và ứng dụng AngularJS**

Nội dung nhận xét:

**Điểm:**

Bằng số:

Bằng chữ:

|  |  |
| --- | --- |
|  | **GIẢNG VIÊN**  *(Ký, ghi rõ họ tên)*  **ThS. Nguyễn Tấn Lộc** |

**MỤC LỤC**

[**MỤC LỤC** ii](#_Toc7459978)

[**DANH MỤC HÌNH** iii](#_Toc7459979)

[**CHƯƠNG 1. TỔNG QUAN VỀ ANGULARJS** 1](#_Toc7459980)

[**1.1. AngularJS là gì?** 1](#_Toc7459981)

[**1.2. Các đặc tính của AngularJS** 1](#_Toc7459982)

[**1.3. Các tính năng của AngularJS** 1](#_Toc7459983)

[**1.4. Ưu điểm và nhược điểm của AngularJS** 2](#_Toc7459984)

[**1.5. Các Components trong AngularJS** 2](#_Toc7459985)

[**CHƯƠNG 2. ỨNG DỤNG ANGULARJS** 4](#_Toc7459986)

[**2.1. Các bước sử dụng AngularJS** 4](#_Toc7459987)

[**2.2. Hình ảnh demo** 16](#_Toc7459988)

[**CHƯƠNG 3. TÀI LIỆU THAM KHẢO** 18](#_Toc7459989)

**DANH MỤC HÌNH**

[Hình 1.1. AngularJS 1](#_Toc7460024)

[Hình 2.1. Cơ sở dữ liệu. 4](#_Toc7460025)

[Hình 2.2. Tạo Entity data model. 4](#_Toc7460026)

[Hình 2.3. Tạo thành công Entity data model 5](#_Toc7460027)

[Hình 2.4. Tìm và cài đặt AngularJS. 5](#_Toc7460028)

[Hình 2.5. Thêm bundle cho angular. 16](#_Toc7460029)

[Hình 2.6. Giao diện demo. 17](#_Toc7460030)

[Hình 2.7. Chức năng Thêm. 17](#_Toc7460031)

[Hình 2.8. Chức năng Cập nhật 17](#_Toc7460032)

**CHƯƠNG 1. TỔNG QUAN VỀ ANGULARJS**

**1.1. AngularJS là gì?**

* Angular là một bộ Javascript Framework rất mạnh và thường được sử dụng để xây dựng project **Single Page Application** (SPA). Nó hoạt động dựa trên các thuộc tính mở rộng HTML (các atributes theo quy tắc của Angular). Đây là một Framework mã nguồn mở hoàn toàn miễn phí và được hàng ngàn các lập trình viên trên thế giới ưa chuộng và sử dụng.



Hình 1.1. AngularJS

**1.2. Các đặc tính của AngularJS**

* AngularJS là một Framwork phát triển dựa trên Javascript để tạo các ứng dụng web phong phú
* AngularJS thường dùng để phát triển frontend (giao diện khách hàng) thông qua các API để gọi data, sử dụng mô hình MVC rất mạnh mẽ
* Mã nguồn AngularJS tự động fix với các trình duyệt khác nhau nên bạn không cần phải lo vấn đề tương thích trình duyệt
* Angular là mã nguồn mở, hoàn toàn miễn phí và được phát triển bởi hàng ngàn các lập trình viên trên thế giới.

**1.3. Các tính năng của AngularJS**

* Data-binding: (liên kết dữ liệu) tự động đồng bộ dữ liệu giữa model và view
* Scope: (Phạm vi) Đây là những đối tượng kết nối giữa Controller và View
* Controller: Đây là những hàm javascript xử lý kết hợp với bộ điều khiển Scope
* Service: Như tôi đề cập ở trên, AngularJS sử dụng các API được xây dựng từ các web service (PHP, ASP) để thao tác với DB.
* Filters: Bộ lọc lọc ra các thành phẩn của một mảng và trả về mảng mới
* Directives:  đánh dấu vào các yếu tố của DOM, nghĩa là sẽ tạo ra các thẻ HTML tùy chỉnh
* Templates: hiển thị thông tin từ controller, đây là một thành phần của views
* Routing:  chuyển đổi giữa các action trong controller
* MVC: Mô hình chia thành phần riêng biệt thành Model, View, Controller. Đây là một mô hình khá hay nhưng trong Angular thì nó được chế biến lại một chút gần giốn với MVVM (Model View View Model)
* Deep Linking: Liên kết sâu, cho phép bạn mã hóa trạng thái của ứng dụng  trong các URL  để nó có thể đánh dấu được với công cụ tìm kiếm.
* Dependency Injection: Angular giúp các nhà phát triển tạo ứng dụng  dễ dàng hơn để phát triển, hiểu và thử nghiệm dễ dàng.

**1.4. Ưu điểm và nhược điểm của AngularJS**

* Ưu điểm:
* Angular cho phép tạo ra các ứng dụng một cách đơn giản, code sạch
* Angular sử dụng data bind giống .NET với tính năng liên kết với HTML nên giúp người dùng cảm thấy dễ chịu.
* Angular đang ở giai đoạn thử nghiệm
* Angular có thể chạy trên hầu hết các trình duyệt điện thoại thông minh.
* Nhược điểm:
* Không an toàn: Được phát triển từ javascript nên nó không an toàn, phía máy chủ phải thường xuyên xác nhận quyền để hệ thống chạy trơn tru.
* Phụ thuộc: Nếu người dùng vô hiệu hóa javascript thì coi như xong.

**1.5. Các Components trong AngularJS**

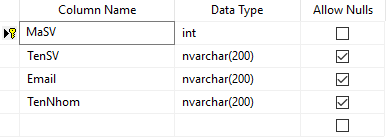
Có 3 components chính là

* ng-app: định nghĩa này chỉ thị một kết nối ứng dụng Angular JS tới HTML
* ng-model: chỉ thị này liên kết với dữ liệu của ứng dụng Angular
* ng-bind: chỉ thị này dùng đưa dữ liệu vào HTML tags.S

**CHƯƠNG 2. ỨNG DỤNG ANGULARJS**

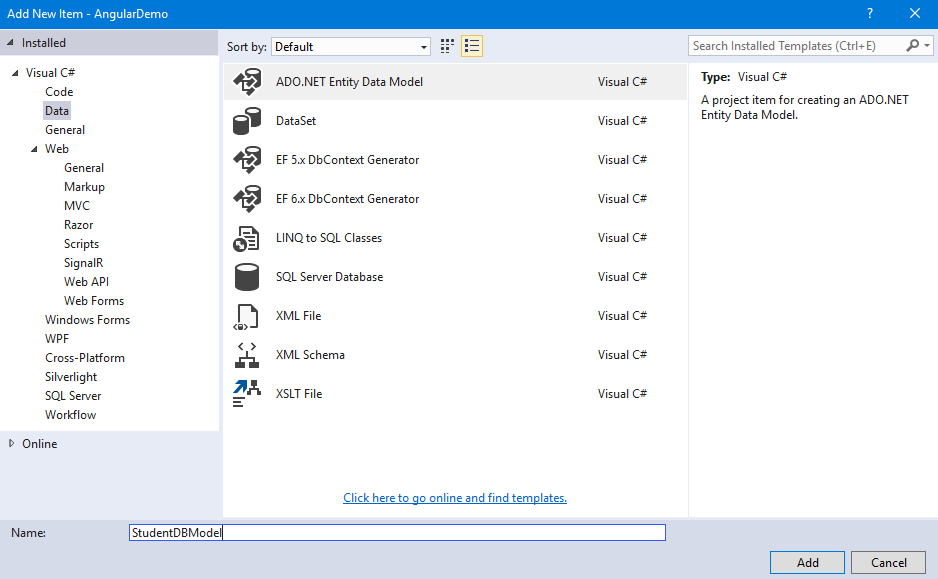
**2.1. Các bước sử dụng AngularJS**

* Bước 1: Tạo project website asp.net MVC.
* Bước 2: Sử dụng SQL server tạo database dùng cho project.

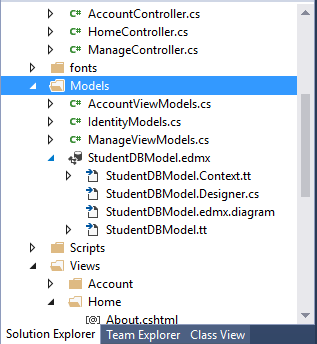


Hình 2.1. Cơ sở dữ liệu.

* Bước 3: Tạo **Entity Data Model lấy database đã tạo ở bước 2.**

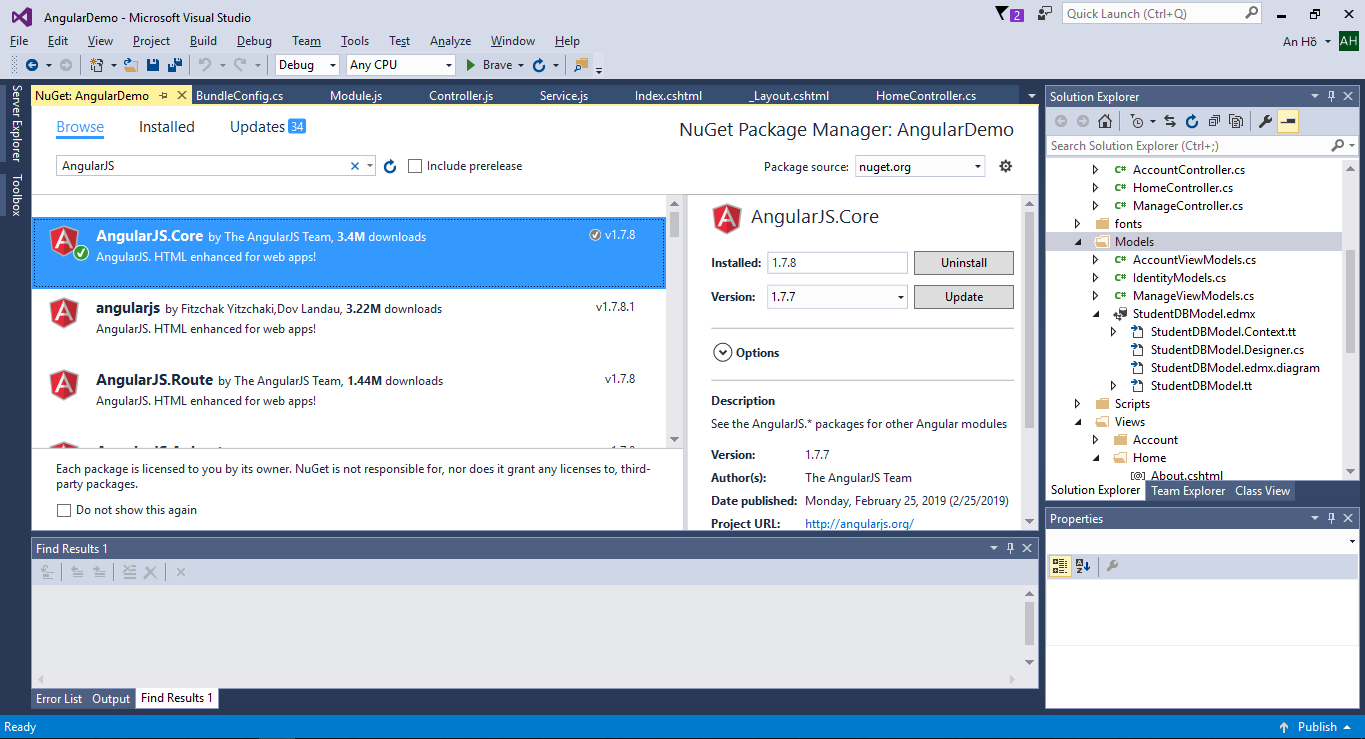
****

Hình 2.2. Tạo Entity data model.



Hình 2.3. Tạo thành công Entity data model

* Bước 4: Vào giao diện NuGet tìm và cài đặt AngularJS.core.



Hình 2.4. Tìm và cài đặt AngularJS.

* Bước 5: Sử dụng HomeController và thêm đoạn code sau đây để lấy danh sách sinh viên từ cơ sở dữ liệu và các chức năng thêm, sửa, xóa.
* Code HomeController

[HttpGet]

public JsonResult GetStudent()

{

using (AngularEntities db = new AngularEntities())

{

List<Student> stuList = db.Students.ToList();

return Json(stuList, JsonRequestBehavior.AllowGet);

}

}

[HttpPost]

public JsonResult Insert(Student student)

{

if (student != null)

{

using (AngularEntities db = new AngularEntities())

{

db.Students.Add(student);

db.SaveChanges();

return Json(new { success = true });

}

}

else

{

return Json(new { success = false });

}

}

[HttpPost]

public JsonResult Update(Student updatedStudent)

{

using (AngularEntities db = new AngularEntities())

{

Student tontaiSinhVien = db.Students.Find(updatedStudent.MaSV);

if (tontaiSinhVien == null)

{

return Json(new { success = false });

}

else

{

tontaiSinhVien.TenSV = updatedStudent.TenSV;

tontaiSinhVien.Email = updatedStudent.Email;

tontaiSinhVien.TenNhom = updatedStudent.TenNhom;

db.SaveChanges();

return Json(new { success = true });

}

}

}

[HttpPost]

public JsonResult Delete(int id)

{

using (AngularEntities db = new AngularEntities())

{

Student student = db.Students.Find(id);

if (student == null)

{

return Json(new { success = false });

}

db.Students.Remove(student);

db.SaveChanges();

return Json(new { success = true });

}

}

* Bước 6: Tạo thư mục Angular và tạo 3 file javascript Controller.js, Module.js, Service.js.
* Code file Controller.js

myapp.controller('student-controller', function ($scope, studentService) {

//Loads all Student records when page loads

loadStudents();

function loadStudents() {

var StudentRecords = studentService.getAllStudents();

StudentRecords.then(function (d) {

$scope.Students = d.data;

},

function () {

alert("Lỗi!");

});

}

//save Student data

$scope.save = function () {

var Student = {

MaSV: $scope.MaSV,

TenSV: $scope.TenSV,

Email: $scope.Email,

TenNhom: $scope.TenNhom,

};

var saverecords = studentService.save(Student);

saverecords.then(function (d) {

if (d.data.success === true) {

loadStudents();

alert("Thêm thành công");

$scope.resetSave();

}

else { alert("Thêm không thành công"); }

},

function () {

alert("Lỗi!");

});

}

//reset controls after save operation

$scope.resetSave = function () {

$scope.MaSV = '';

$scope.TenSV = '';

$scope.Email = '';

$scope.TenNhom = '';

}

//get single record by ID

$scope.getForUpdate = function (Student) {

$scope.UpdateMaSV = Student.MaSV;

$scope.UpdateTenSV = Student.TenSV;

$scope.UpdateEmail = Student.Email;

$scope.UpdateTenNhom = Student.TenNhom;

}

//get data for delete confirmation

$scope.getForDelete = function (Student) {

$scope.UpdateMaSV = Student.MaSV;

$scope.UpdateTenSV = Student.TenSV;

}

//update Student data

$scope.update = function () {

var Student = {

MaSV: $scope.UpdateMaSV,

TenSV: $scope.UpdateTenSV,

Email: $scope.UpdateEmail,

TenNhom: $scope.UpdateTenNhom,

};

var updaterecords = studentService.update(Student);

updaterecords.then(function (d) {

if (d.data.success === true) {

loadStudents();

alert("Cập nhật thành công");

$scope.resetUpdate();

}

else {

alert("Cập nhật không thành công");

}

},

function () {

alert("Lỗi!");

});

}

//reset controls after update

$scope.resetUpdate = function () {

$scope.UpdateMaSV = '';

$scope.UpdateTenSV = '';

$scope.UpdateEmail = '';

$scope.UpdateTenNhom = '';

}

//delete Student record

$scope.delete = function (UpdateMaSV) {

var deleterecord = studentService.delete($scope.UpdateMaSV);

deleterecord.then(function (d) {

if (d.data.success === true) {

loadStudents();

alert("Xóa thành công");

}

else {

alert("Xóa không thành công");

}

});

}

});

* Code file Module.js

var myapp;

(function () {

myapp = angular.module('my-students', []);

})();

* Code file Service.js

//Service to get data from student mvc controller

myapp.service('studentService', function ($http) {

//read students

this.getAllStudents = function () {

return $http.get('/Home/GetStudent');

}

//add new student

this.save = function (Student) {

var request = $http({

method: 'post',

url: '/Home/Insert',

data: Student

});

return request;

}

//update Student records

this.update = function (Student) {

var updaterequest = $http({

method: 'post',

url: '/Home/Update',

data: Student

});

return updaterequest;

}

//delete record

this.delete = function (UpdateMaSV) {

return $http.post('/Home/Delete/' + UpdateMaSV);

}

});

* Bước 7: Tạo View để hiển thị dữ liệu.
* Code file Index.cshtml

@{

ViewBag.Title = "Home Page";

}

<div class="container" ng-controller="student-controller">

<div class="panel panel-info">

<div class="panel-heading">

Quản Lý Sinh Viên

</div>

<div class="panel-body">

<button type="button" class="btn btn-info"

data-target="#AddNew" data-toggle="modal">

Thêm Sinh Viên

</button>

<br />

</div>

@\*Records\*@

<table class="table table-bordered">

<thead style="background-color:#ffd800;">

<tr>

<th>Mã Sinh Viên</th>

<th>Tên Sinh Viên</th>

<th>Email</th>

<th>Tên Nhóm</th>

<th>Actions</th>

</tr>

</thead>

<tbody>

<tr ng-repeat="stu **in** Students">

<td>**{{**stu.MaSV**}}**</td>

<td>**{{**stu.TenSV**}}** </td>

<td>**{{**stu.Email**}}**</td>

<td>**{{**stu.TenNhom**}}**</td>

<td style="width:200px;">

<a href="#"

class="btn btn-info"

data-toggle="modal"

data-target="#Update"

ng-click="getForUpdate(stu)">

Cập Nhật

</a>

<a href="#" class="btn btn-danger"

id="btnDelete"

data-toggle="modal"

data-target="#deleteDialog"

ng-click="getForDelete(stu)">

Xóa

</a>

</td>

</tr>

</tbody>

</table>

</div>

@\*New record Modal addition..\*@

<div class="modal" id="AddNew" role="dialog">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal">&times;</button>

<h3 class="text-info">Thêm</h3>

</div>

<div class="modal-body" style="margin-left:20px">

<form class="form-horizontal" name="AddNewForm">

<div class="form-group">

<input class="form-control" readonly="readonly" name="MaSV" ng-model="MaSV" type="hidden" placeholder="" />

</div>

<div class="form-group">

<label class="control-label"> Tên Sinh Viên</label>

<input class="form-control" name="TenSV" ng-model="TenSV" type="text" placeholder="" />

</div>

<div class="form-group">

<label class="control-label"> Email</label>

<input class="form-control" name="Email" ng-model="Email" type="email" placeholder="" />

</div>

<div class="form-group">

<label class="control-label"> Tên Nhóm</label>

<input class="form-control" name="TenNhom" ng-model="TenNhom" type="text" placeholder="" />

</div>

</form>

</div>

<div class="modal-footer">

<button type="button" class="btn btn-primary" id="btnSave" data-dismiss="modal" ng-click="save()">Lưu</button>

<button type="button" class="btn btn-default" data-dismiss="modal" ng-click="resetSave()">Đóng</button>

</div>

</div>

</div>

</div>

<div class="modal" id="Update" role="dialog">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal">&times;</button>

<h3 class="text-info">Cập Nhật</h3>

</div>

<div class="modal-body" style="margin-left:20px">

<form class="form-horizontal" name="UpdateStudent">

<div class="form-group">

<input class="form-control" readonly="readonly" name="MaSV" ng-model="UpdateMaSV" type="hidden" placeholder="" />

</div>

<div class="form-group">

<label class="text-info">Tên Sinh Vien</label>

<input class="form-control" name="TenSV" ng-model="UpdateTenSV" type="text" placeholder="" />

</div>

<div class="form-group">

<label class="text-info">Email</label>

<input class="form-control" name="Email" ng-model="UpdateEmail" type="email" placeholder="" />

</div>

<div class="form-group">

<label class="text-info">Tên Nhóm</label>

<input class="form-control" name="TenNhom" ng-model="UpdateTenNhom" type="text" placeholder="" />

</div>

</form>

</div>

<div class="modal-footer">

<button type="button" class="btn btn-primary" id="btnUpdate" data-dismiss="modal" ng-click="update()">

Cập Nhật

</button>

<button type="button" class="btn btn-default" data-dismiss="modal">Đóng</button>

</div>

</div>

</div>

</div>

<div class="modal" role="dialog" id="deleteDialog">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal">&times;</button>

<h3 class="text-info">Bạn muốn Xóa sinh viên này?</h3>

</div>

<div class="modal-body">

<div>

<p>Mã Sinh Viên : **{{**UpdateMaSV**}}**</p>

<p>Tên Sinh Viên : **{{**UpdateTenSV**}}**</p>

</div>

</div>

<div class="modal-footer">

<button type="button" class="btn btn-danger" data-dismiss="modal" ng-click="delete(UpdateMaSV)">Xóa</button>

<button type="button" class="btn btn-default" data-dismiss="modal">Đóng</button>

</div>

</div>

</div>

</div>

</div>

@section scripts{

<script src="~/Angular/Module.js"></script>

<script src="~/Angular/Service.js"></script>

<script src="~/Angular/Controller.js"></script>

}

* Code file \_Layout.cshtml cho file Index.cshtml.

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>@ViewBag.Title - My ASP.NET Application</title>

@Styles.Render("~/Content/css")

@Scripts.Render("~/bundles/modernizr")

</head>

<body ng-app="my-students">

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

@Html.ActionLink("Application name", "Index", "Home", new { area = "" }, new { @class = "navbar-brand" })

</div>

<div class="navbar-collapse collapse">

<ul class="nav navbar-nav">

<li>@Html.ActionLink("Home", "Index", "Home")</li>

<li>@Html.ActionLink("About", "About", "Home")</li>

<li>@Html.ActionLink("Contact", "Contact", "Home")</li>

</ul>

</div>

</div>

</div>

<div class="container body-content">

@RenderBody()

<hr />

<footer>

<p>&copy; @DateTime.Now.Year - My ASP.NET Application</p>

</footer>

</div>

@Scripts.Render("~/bundles/jquery")

@Scripts.Render("~/bundles/angular")

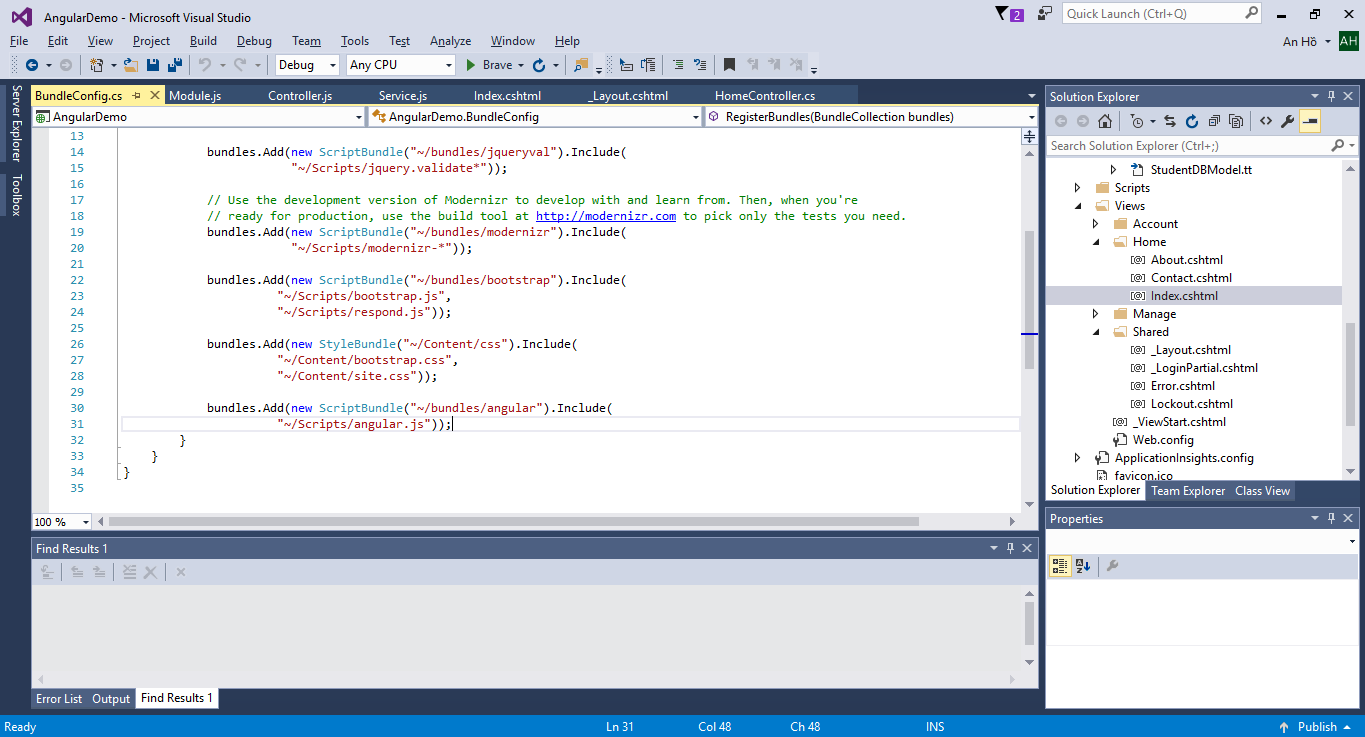
@Scripts.Render("~/bundles/bootstrap")

@RenderSection("scripts", required: false)

</body>

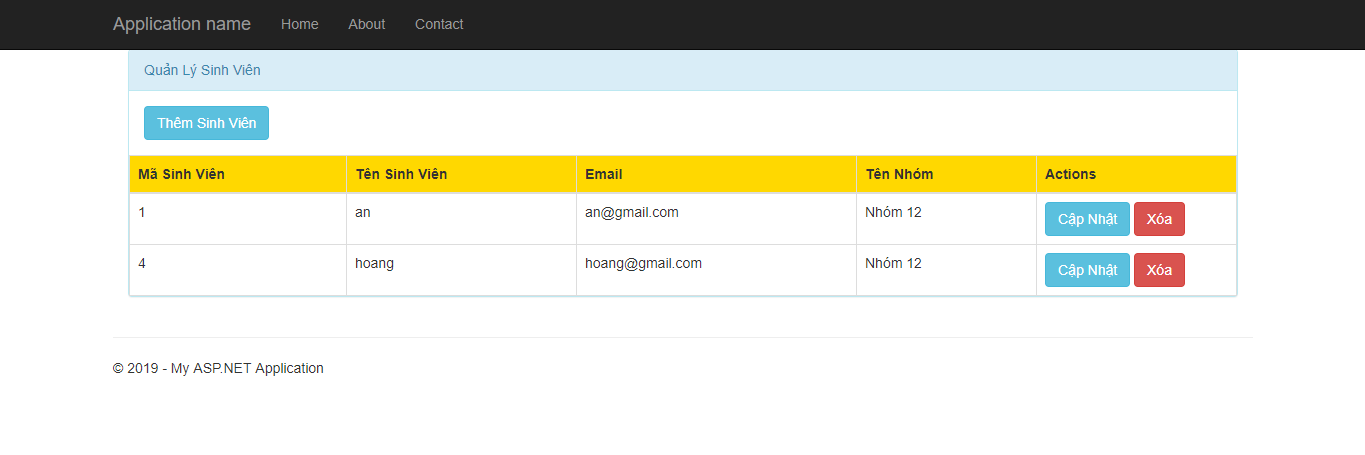
</html>

* Bước 8: Mở file BundleConfig.cs trong thư mục App\_Start để thêm bundle cho angular.

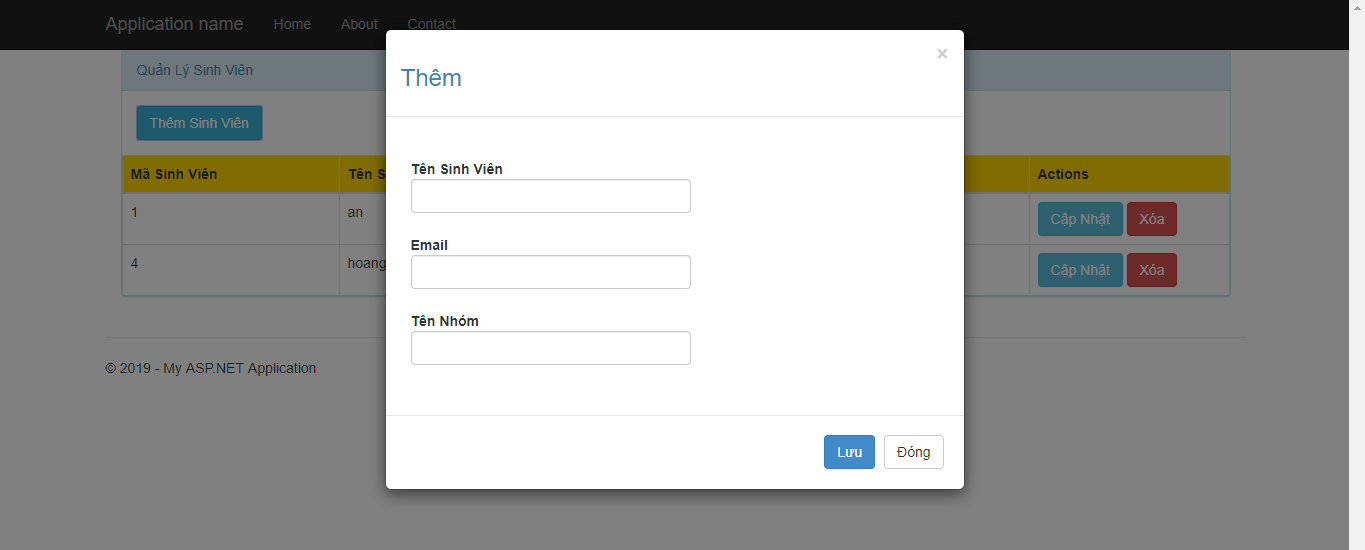


Hình 2.5. Thêm bundle cho angular.

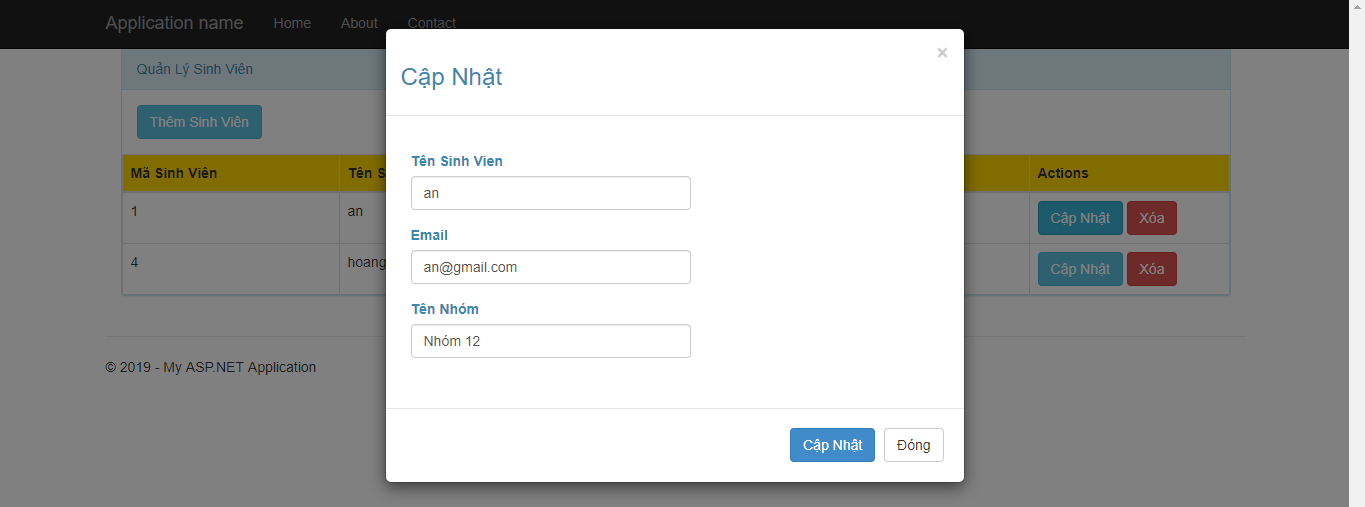
**2.2. Hình ảnh demo**



Hình 2.6. Giao diện demo.



Hình 2.7. Chức năng Thêm.



Hình 2.8. Chức năng Cập nhật

**CHƯƠNG 3. TÀI LIỆU THAM KHẢO**

* <https://vi.wikipedia.org/wiki/AngularJS>
* <https://www.w3schools.com/angular/>