MONGO DB:

- MongoDB is a cross-platform document-oriented database. Classified as a NoSQL database, MongoDB eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas.
- Mongoose is a Node.js library that provides MongoDB object mapping similar to ORM with a familiar interface within Node.js. If you aren't familiar with Object Relational Mapping (ORM) .

Express.js-Node.js

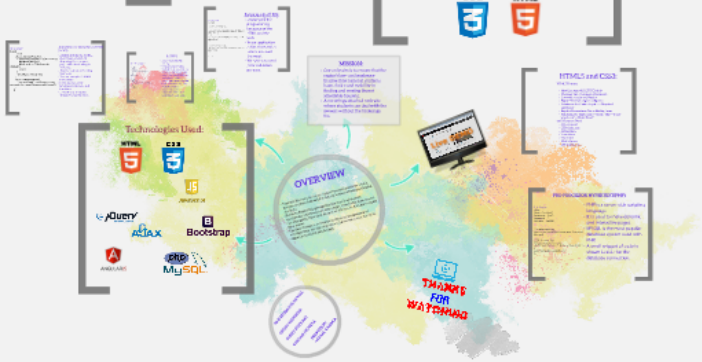
- Express.js is a Node.js web application server framework, designed for building single-page, multi-page, and hybrid web applications.
- Node.js is an open-source, cross-platform runtime environment for developing server-side Web applications.

Libraries and References:

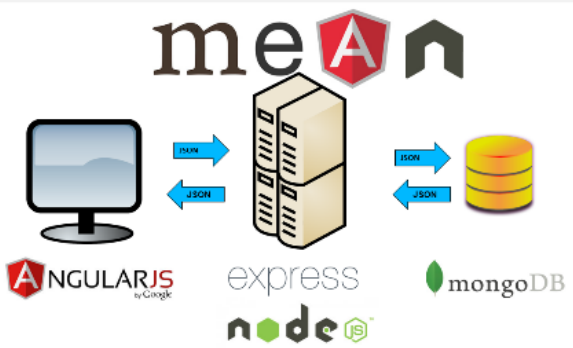
Technologies Used:

ANGULAR JS:

- AngularJS Animations
- AngularJS Directives
- Angular 2-way binding
- Auto Complete
- Angular Validations
- SPA using templating & routing



Overview of MEAN:



Collection of JavaScript based technologies used to develop web applications.

Code Snippet:

CSS, Bootstrap and JQuery Features:

- Create the autocomplete object, restricting the search to geographical location types.
- CSS3 animate, transform and transition
- JQuery Events as Angular Directives

FEATURES:

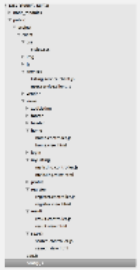
- Implemented RESTful Custom API

- Passport Local Strategy

Project Folder: Server Side

```
server
  models
    listing.model.server.js
    listing.schema.server.js
    place.schema.server.js
    user.model.server.js
    user.schema.server.js
  services
    listing.service.server.js
    user.service.server.js
  app.js
index.html
.gitignore
package.json
README.md
server.js
```

Project Folder: Client Side



NEU STUDENTS RENTAL

GROUP MEMBERS:

SMEET KOTHARI

CHETAN NIGUDGI

PROFESSOR:
-VISHAL CHAWLA

OVERVIEW

- Students Rental is an house/property rental platform which focuses on providing reliable data to home seekers prioritizing students.
- Students Rental brings together thousands of student apartments and houses on one simple, searchable map, so you can stop going from door to door or site to site. It's time to make your move.
- The Search page is accessible to all users, irrespective of whether they are signed in or not. However, a user has to be signed in to add or pin listings.

MISSION:

- Our only aim is to ensure that the region's low- and moderate-income international students have choice and mobility in finding and renting decent affordable housing.
- A no strings attached web-site where students can deal with the owners without the brokerage fee.

Technologies Used:

JavaScript

jQuery
write less, do more.

AJAX

B
Bootstrap

ANGULARJS

php

MySQL


AX
Bootstrap
php
MySQL™


jQuery
write less, do more.
AJAX


JavaScript

Technologies Used

PRE-PROCESSOR HYPERTEXT(PHP):

Code Snippet:

```
<?php
 $host = "localhost";
 $username = "root";
 $password = "";
 $dbname = "webDB";


 // Create connection
 $conn = new mysqli($host, $username, $password,
$dbname);

 // Check connection
 if ($conn->connect_error) {
  die("Connection failed: " . $conn->connect_error);
 }
 echo "Connected successfully";
?>
```

- PHP is a server side scripting language.
- It is used to make dynamic and interactive pages.
- MYSQL is the most popular database system used with PHP.
- A small snippet of code is shown beside for the database connection.

JQUERY:

- Jquery is a Javascript Library.
- Purpose of Jquery is to make it much easier to use Javascript on our website.
- Jquery also has UI Widgets and different effects.
- Jquery is the most powerful framework of JS and also most extentable.
- Jquery is used for Pagination and Infinite Scrolling of a web page.

UI Widgets example:

1.

```
<script>
 $(function() {
   $( document ).tooltip();
 });
</script>
```

2.

```
 <script>
 $(function() {
   $( "#datepicker" ).datepicker();
 });
 </script>
```

Code Snippet:

```
$.ajax({
        url: url,
        success: function(response) {
           if (!response || response.trim() == "NONE") {
              $(buttonId).fadeOut();
              $(loadingId).text("Thank you for
Visiting!!");
              return;
           }
           appendContests(response);
        },
        error: function(response) {
           $(loadingId).text("Sorry, there was some
error with the request. Please refresh the page.");
        }
     });
```

Asynchronous JavaScript and XML (AJAX):

- AJAX is nothing but loading data in background and displaying it on the web page, without reloading the web page.
- JQuery makes AJAX coding very easy.
- The best example of AJAX is Facebook.
- Lazy Loading can be implemented using AJAX functions.
- A small code snippet is shown for Ajax function.

Code Snippet:

```
<script>

  function modalFunc() {

document.getElementById("myModal").style.dis
play = 'block';
   //
document.getElementById("longwood").style.dis
play = 'block';
}

  function modalClose() {
 document.getElementById("myModal").style.dis
play = 'none';
}

</script>
```

JavaScript(JS):

- Javascript is the programming language of the HTML and the web.
- In our application JS functions and JS Events are used the most.
- We have also used JS for validation purpose.

HTML5 and CSS3:

HTML5 Features:

- New Doctype: <!DOCTYPE html>
- Marquee Tag: <marquee></marquee>
- Semantic Header and Footer:
- Figure Element: <figure></figure>
- Autofocus Attribute: <input ..... Required autofocus>
- Regular Expressions: For validating form.
- Placeholders: <input name="email" type="email" placeholder="Enter Email"

CSS3 Features Used:

- CSS Carousel
- CSS Modal box
- Animations
- Transitions
- Gradients
- Media Query
- CSS Grid Layout

File Edit View History Bookmarks Tools Help

pduCoder - Play with code...

localhost:3000/#!/

Search

</>PDU Coder

Phòng chat Articles Create New Article Videos Gửi video Quản trị viên -> Đăng xuất

Chào mừng bạn đến với pduCoder. Để tiếp tục sử dụng, bạn hãy đăng kí thành viên, để có thể viết bài, hoặc tham gia phòng chát với mọi người.

Live DEMO

CÁC GÓ G D G

Dành cho "giảng viên" đăng tải các video/clip bài giảng của mình phục vụ cho việc học tập.

Như mọi bài viết thông thường ở mọi nơi khác, mọi thành viên đều có thể đăng bài.

PHÒNG CHÁT

Tham gia tán gẫu, trò chuyện, làm quen giữa các thành viên với nhau.

BÌNH LUẬN

Bình luận các chủ đề, bài viết, video với thời gian thực

pdu Coder by Mr.Quang Bão

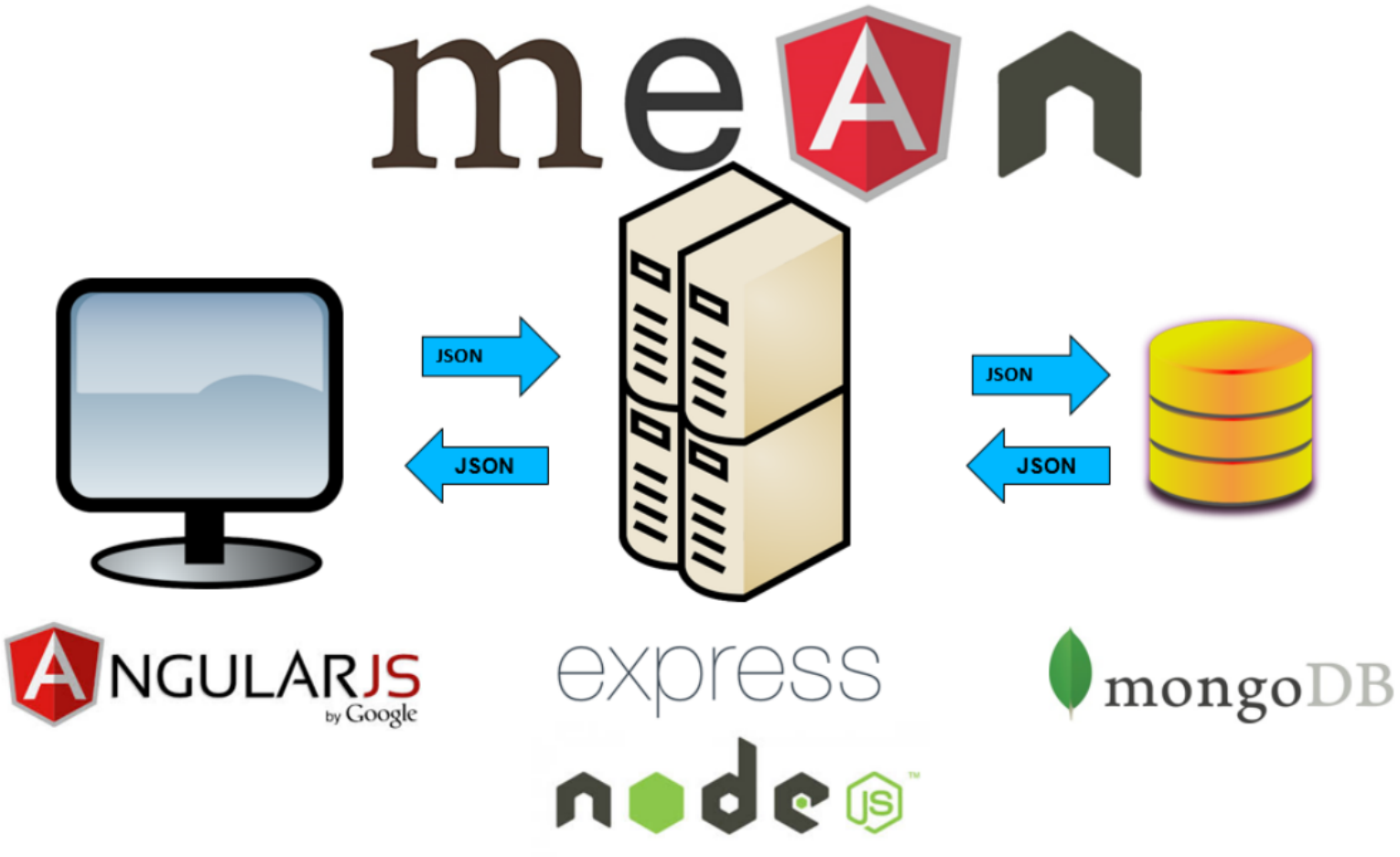
Trang web ứng dụng cho việc giảng dạy, học hỏi và trao đổi kiến thức, giao lưu giữa các giảng viên, sinh viên khoa công nghệ thông tin.

Khoa công nghệ thông tin trường Đại học Phạm Văn Đồng
+ 99 Hùng Vương - TP Quảng Ngãi

© Copyright 2015
Design & Develop by Mr.Quang Bão

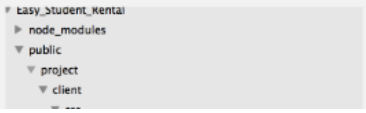
Technologies Used:

Overview of MEAN:



Collection of JavaScript based technologies used to develop web applications.

Project Folder: Client Side



MONGO DB:

- MongoDB is a cross-platform document-oriented database. Classified as a NoSQL database, MongoDB eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas.
- Mongoose is a Node.js library that provides MongoDB object mapping similar to ORM with a familiar interface within Node.js. If you aren't familiar with Object Relational Mapping (ORM) .

Code Snippet:

```
"use strict";

module.exports = function (mongoose) {
    var PlaceSchema = new mongoose.Schema({
        street_number: String,
        route: String,
        administrative_area_level_1: String,
        locality: String,
        postal_code: String,
        country: String,
        displayAddress: String,
        formatted_address: String,
        lat: String,
        lng: String
    });

    return PlaceSchema;
};
```

```
"use strict";

module.exports = function (mongoose) {
    var UserSchema = new mongoose.Schema({
        firstname: String,
        lastname: String,
        username: String,
        password: String,
        email: String,
        pinned: [String]
    }, {collection: 'easyRental.project.user'});

    return UserSchema;
};
```

Express.js-Node.js

- Express.js is a Node.js web application server framework, designed for building single-page, multi-page, and hybrid web applications.
- Node.js is an open-source, cross-platform runtime environment for developing server-side Web applications.

FEATURES:

- Implemented RESTful Custom API

```
module.exports = function (app, model) {
    app.post('/api/project/listing/:userid', createListing);
    app.get('/api/project/listing/user/:userid', findAllListingsForUser);
    app.get('/api/project/listing/:listingid', findListingById);
    app.delete('/api/project/listing/:listingid/user/:userid', deleteListing);
    app.get('/api/project/search', searchListings);

    function createListing(req, res) {
        var listing = req.body;
        var userid = req.params.userid;

        model.Create(listing)
            .then(function (newListing) {
                res.json(newListing);
            });
    }

    function findAllListingsForUser(req, res) {
        var userid = req.params.userid;

        model.FindAll(userid)
            .then(function (listings) {
                res.json(listings);
            });
```

```
            });
    }

    function findListingById(req, res){
        var listingid = req.params.listingid;

        model.FindById(listingid)
            .then(function (listing) {
                res.json(listing);
            });
    }

    function deleteListing(req, res) {
        var listingid = req.params.listingid;
        var userid = req.params.userid;

        model.Delete(listingid, userid)
            .then(function (listings) {
                res.json(listings);
            });
    }
```

- Passport Local Strategy

```
'use strict';

module.exports = function (app, mongoose, db, passport, LocalStrategy) {
    var userModel = require('./models/user.model.server.js')(mongoose, db, passport, LocalStrategy);
    var listingModel = require('./models/listing.model.server.js')(mongoose, db);

    require('./services/user.service.server.js')(app, userModel, passport, LocalStrategy);
    require('./services/listing.service.server.js')(app, listingModel);
```

```
module.exports = function (app, model, passport, LocalStrategy) {
    app.post("/api/project/user", createUser);
    app.get("/api/project/user", findUsers);
    app.get("/api/project/user/:id", findUserById);
    app.put("/api/project/user/:id", updateUser);
    app.delete("/api/project/user/:id", deleteUser);
    app.get("/api/project/loggedin", loggedin);
    app.post("/api/project/logout", logout);
    app.put("/api/project/user/:userid/listing/:listingid", pinListing);

    function loggedin(req, res) {
        res.send(req.isAuthenticated() ? req.user : '0');
    }

    function logout(req, res) {
        req.logOut();
        res.send(200);
    }

    passport.use(new LocalStrategy(
        function (username, password, done) {
            model
                .FindByAuth(username, password)
                .then(function (user) {
                    if (!user) {
                        return done(null, false);
                    }
                    return done(null, user);
                }, function(err) {
                    done(err);
                });
        }));
```

ANGULAR JS:

- AngularJS Animations
- AngularJS Directives
- Angular 2-way binding
- Auto Complete
- Angular Validations
- SPA using templating & routing

```
// [START region_fillform]
    function fillInAddress() {
        // Get the place details from the autocomplete object.
        var place = autocomplete.getPlace();

        for (var component in componentForm) {
            document.getElementById(component).value = '';
            document.getElementById(component).disabled = false;
        }

        // Get each component of the address from the place details
        // and fill the corresponding field on the form.
        for (var i = 0; i < place.address_components.length; i++) {
            var addressType = place.address_components[i].types[0];
            if (componentForm[addressType]) {
                var val = place.address_components[i][componentForm[addressType]];
                document.getElementById(addressType).value = val;
            }
        }
    }

// [END region_fillform]

// [START region_geolocation]
// Bias the autocomplete object to the user's geographical location,
// as supplied by the browser's 'navigator.geolocation' object.
    function geolocate() {
        if (navigator.geolocation) {
            navigator.geolocation.getCurrentPosition(function (position) {
                var geolocation = {
                    lat: position.coords.latitude,
                    lng: position.coords.longitude
                };
                var circle = new google.maps.Circle({
                    center: geolocation,
                    radius: position.coords.accuracy
                });
                autocomplete.setBounds(circle.getBounds());
            });
        }
    }
```

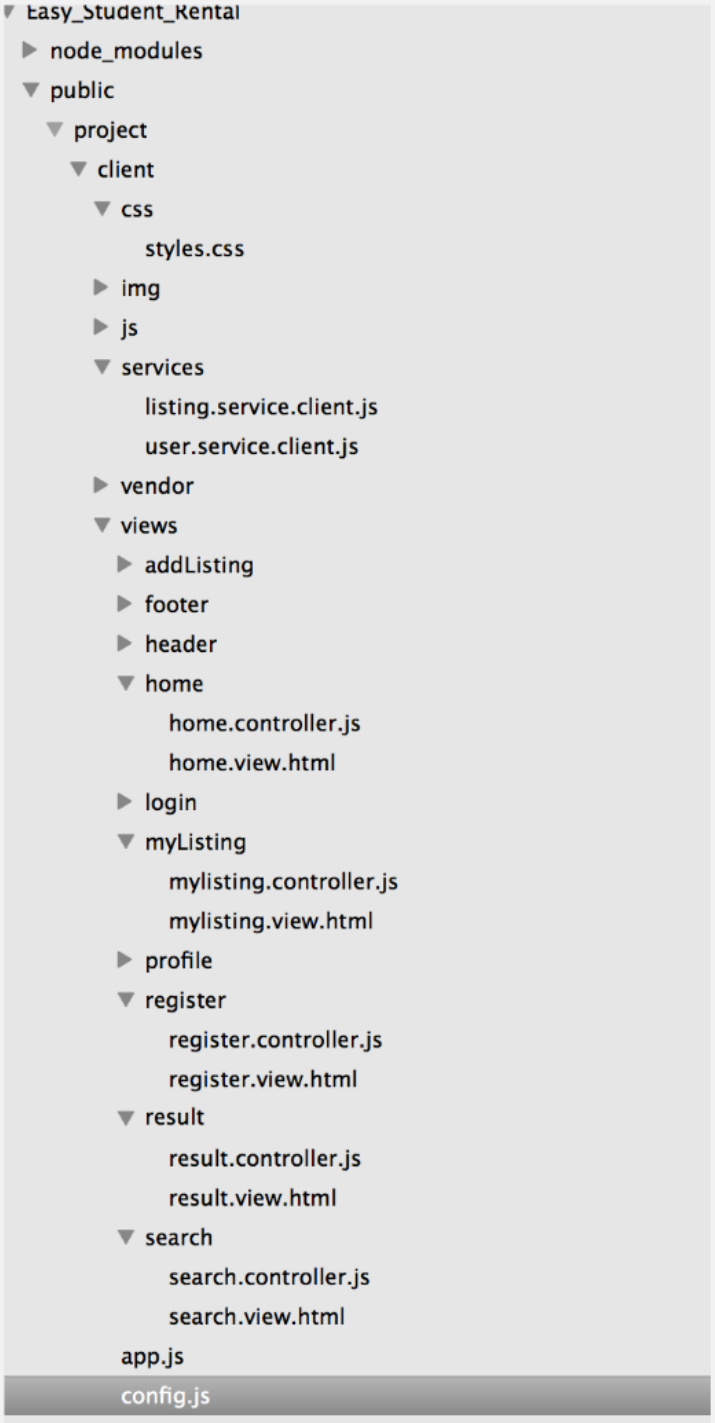
CSS, Bootstrap and JQuery Features:

- Create the autocomplete object,
- restricting the search to geographical location types.
- CSS3 animate, transform and transition
- JQuery Events as Angular Directives

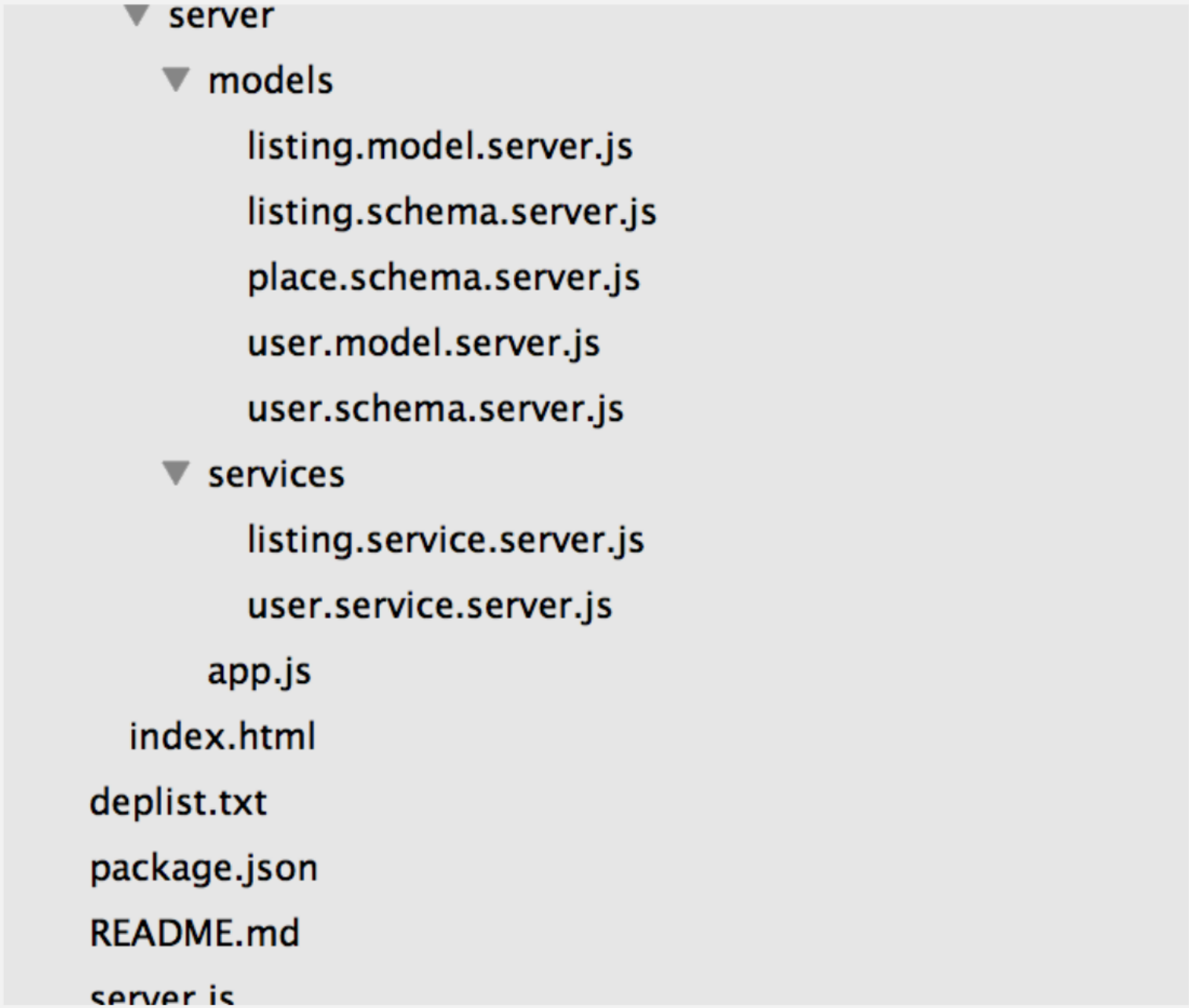
Libraries and References:

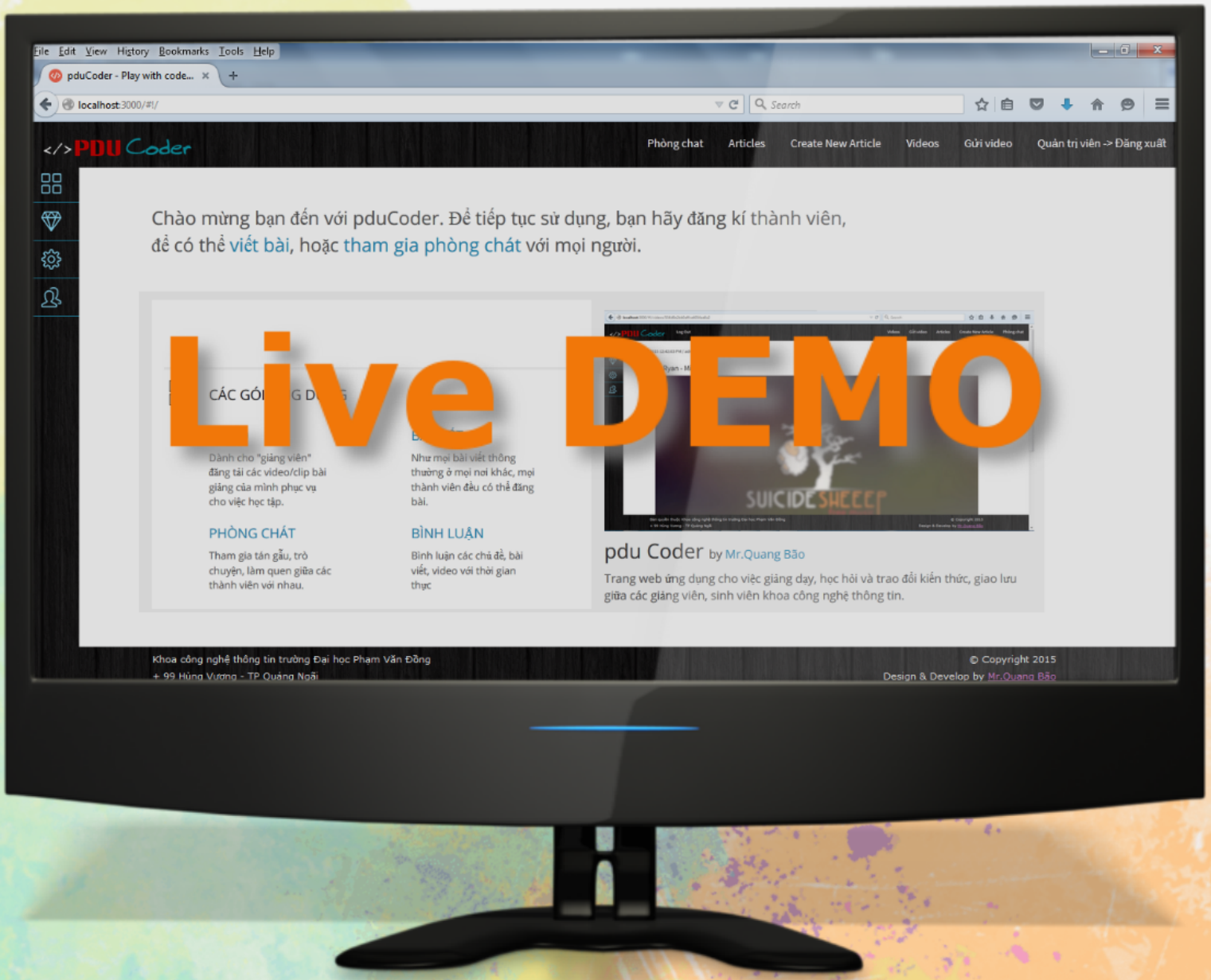
- W3schools
- JsFiddle
- <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
- <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/font-awesome/4.5.0/css/font-awesome.min.css">
- <link href="chelseaCss.css" rel="stylesheet"> <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
- <script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
- <script src="https://code.jquery.com/jquery-2.2.3.min.js"></script>
- <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>

Project Folder: Client Side



Project Folder: Server Side




File Edit View History Bookmarks Tools Help
pduCoder - Play with code...
localhost:3000/#!/
Search
</>PDU Coder
Phòng chat
Articles
Create New Article
Videos
Gửi video
Quản trị viên -> Đăng xuất
Chào mừng bạn đến với pduCoder. Để tiếp tục sử dụng, bạn hãy đăng kí thành viên,
để có thể viết bài, hoặc tham gia phòng chát với mọi người.
Live DEMO
Dành cho "giảng viên"
đăng tải các video/clip bài
giảng của mình phục vụ
cho việc học tập.
Như mọi bài viết thông
thường ở mọi nơi khác, mọi
thành viên đều có thể đăng
bài.
PHÒNG CHÁT
Tham gia tán gẫu, trò
chuyện, làm quen giữa các
thành viên với nhau.
BÌNH LUẬN
Bình luận các chủ đề, bài
viết, video với thời gian
thực
pdu Coder by Mr.Quang Bão
Trang web ứng dụng cho việc giảng dạy, học hỏi và trao đổi kiến thức, giao lưu
giữa các giảng viên, sinh viên khoa công nghệ thông tin.
Khoa công nghệ thông tin trường Đại học Phạm Văn Đồng
+ 99 Hùng Vương - TP Quảng Ngãi
© Copyright 2015
Design & Develop by Mr.Quang Bão

THANKS FOR WATCHING