

Tutorial 3

(Rest Services, Mashup)

CS551 Advanced Software Engineering

UMKC

Topics to cover

- REST Services
- Four Square Service
- Alchemy Service
- Angular JS
- Additional

REST Services

Rest(Representational State Transfer) is an architectural style that specifies certain constraints for communication over web.

The REST architectural style constrains an architecture to a client/server architecture and is designed to use a stateless communication protocol, typically HTTP.

In the REST architecture style, clients and servers exchange representations of resources by using a standardized interface and protocol.

REST Operations

- 1.**Get** - Retrieves the current state of a resource
- 2.**Post** - Updates the state of an existing resource
- 3.**Put** - Creates a new resource
- 4.**Delete** - Deletes some info

Four Square Search API

<https://developer.foursquare.com/start/search>

Search Venues

<https://api.foursquare.com/v2/venues/search>

Returns a list of venues near the current location, optionally matching a search term.

Parameters

All parameters are optional, unless otherwise indicated.

VENUE_ID : required ID of venue to retrieve.

Response fields

Venue :A complete venue.

Four Square Search API

<https://developer.foursquare.com/docs/venues/venue>

S

FOURSQUARE FOR DEVELOPERS

[My Apps](#) [Overview](#) [Endpoints](#) [Forum](#) [Twitter](#) [Blog](#)

developer.foursquare.com » API Documentation

Users

Venues

[add](#)
[categories](#)
[explore](#)
[managed](#)
[search](#)
[suggestcompletion](#)
[timeseries](#)
[trending](#)
Aspects
[events](#)
[herenow](#)
[hours](#)
[likes](#)
[links](#)
[listed](#)
[menu](#)
[nextvenues](#)
[photos](#)
[similar](#)
[tips](#)

Venue Detail

https://api.foursquare.com/v2/venues/VENUE_ID

Gives details about a venue, including location, mayorship, tags, tips, specials, and category.

Authenticated users will also receive information about who is here now.

If the venue ID given is one that has been merged into another "master" venue, the response will show data about the "master" instead of giving you an error.

HTTP Method	GET
Requires Acting User	No (learn more)
Modes supported	foursquare , swarm (learn more)

Parameters

All parameters are optional, unless otherwise indicated.

VENUE_ID	A9ABCD	required ID of venue to retrieve
----------	--------	----------------------------------

Four Square Tips(Reviews) API

<https://developer.foursquare.com/start/search>

Get Reviews/Tips Venues

Tips from a Venue :Returns tips for a venue.

https://api.foursquare.com/v2/venues/VENUE_ID/tips

HTTP Method : GET

Parameters:

All parameters are optional, unless otherwise indicated.

VENUE_ID	XXX123YYYY	required The venue you want tips for.
sort	recent	One of friends, recent, or popular.
limit	100	Number of results to return, up to 500.
offset	100	Used to page through results.

Four Square Tips(Reviews) API

Users

Venues

Venuegroups

Checkins

Tips

[add](#)
[search](#)

Aspects

[likes](#)
[listed](#)
[saves](#)

Actions

[flag](#)
[like](#)
[unmark](#)

Lists

Updates

Tip detail

https://api.foursquare.com/v2/tips/TIP_ID

Gives details about a tip, including which users (especially friends) have marked the tip to-do.

HTTP Method	GET
Requires Acting User	No (learn more)
Modes supported	foursquare (learn more)

Parameters

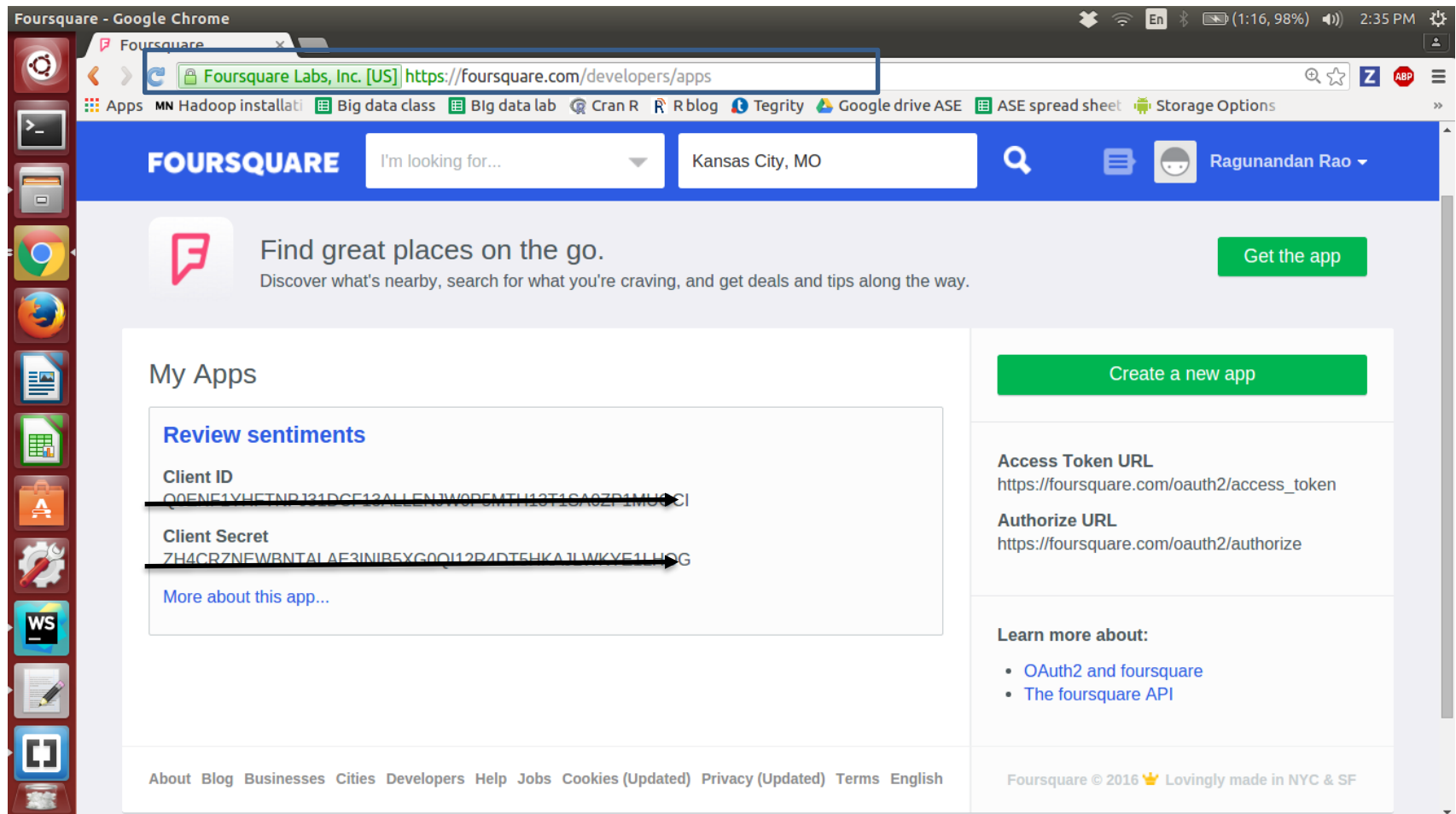
All parameters are optional, unless otherwise indicated.

TIP_ID	A9ABCD	required ID of tip to retrieve
--------	--------	--------------------------------

Response fields

tip	A complete tip.
-----	-----------------

What is needed to Access API ?.... A Key !!!



The screenshot shows the Foursquare Labs, Inc. [US] page in Google Chrome. The browser address bar highlights the URL <https://foursquare.com/developers/apps>. The page features a blue header with the Foursquare logo, a search bar, and a location dropdown set to "Kansas City, MO". Below the header, a green button says "Get the app". The main content area is titled "My Apps" and displays details for an app named "Review sentiments". Two black arrows point to the "Client ID" and "Client Secret" fields, which contain long alphanumeric strings. To the right, a green button says "Create a new app". Below this, the "Access Token URL" and "Authorize URL" are listed. At the bottom, there is a "Learn more about:" section with links to "OAuth2 and foursquare" and "The foursquare API". The footer contains links for "About", "Blog", "Businesses", "Cities", "Developers", "Help", "Jobs", "Cookies (Updated)", "Privacy (Updated)", "Terms", and "English", along with the copyright notice "Foursquare © 2016 Lovingly made in NYC & SF".

Foursquare Labs, Inc. [US] <https://foursquare.com/developers/apps>

FOURSQUARE I'm looking for... Kansas City, MO

Find great places on the go.
Discover what's nearby, search for what you're craving, and get deals and tips along the way. [Get the app](#)

My Apps

Review sentiments

Client ID
OQENF1YHETNP131DCF13ALLEN3W0PE5MTH13T1GA0ZP1MU0CI

Client Secret
ZH4CR7NEWBNTALAE3INIB5XG0Q112P4DT5HKAJLWICYE1LHOG

[More about this app...](#)

[Create a new app](#)

Access Token URL
https://foursquare.com/oauth2/access_token

Authorize URL
<https://foursquare.com/oauth2/authorize>

Learn more about:

- [OAuth2 and foursquare](#)
- [The foursquare API](#)

[About](#) [Blog](#) [Businesses](#) [Cities](#) [Developers](#) [Help](#) [Jobs](#) [Cookies \(Updated\)](#) [Privacy \(Updated\)](#) [Terms](#) [English](#)

Foursquare © 2016 Lovingly made in NYC & SF

Alchemy Sentiment Analysis API

AlchemyAPI provides easy-to-use mechanisms to identify positive/negative sentiment within any document or web page.

AlchemyAPI Sentiment Analysis APIs are capable of computing document-level sentiment, user-specified sentiment targeting, entity-level sentiment, emoticons and keyword-level sentiment. Multiple modes of sentiment analysis provide for a variety of use cases ranging from social media monitoring to trend analysis.

API Call: TextGetTextSentiment

Description: The TextGetTextSentiment call is utilized to extract positive/negative sentiment from within a posted text document.

Endpoint:

<http://gateway-a.watsonplatform.net/calls/text/TextGetTextSentiment>

Alchemy Sentiment Analysis API

[Services](#)[Resources](#)[Developers](#)[Company](#)[Blog](#)[SCHEDULE A DEMO](#)[Documentation](#) » [Sentiment Analysis](#) » [API Calls: Sentiment Analysis](#)

API DOCUMENTATION MENU

- ▼ [Documentation](#)
 - › [Calling the API](#)
 - › [News API](#)
- ▼ [Sentiment Analysis](#)
 - ▼ [API Calls: Sentiment Analysis](#)
 - › [HTML API](#)
 - › [Text API](#)
 - › [Web API](#)
 - › [Languages Support: Sentiment Analysis](#)
 - › [Response Formats: Sentiment Analysis](#)

API Calls: Sentiment Analysis

AlchemyAPI offers easy-to-use facilities for processing your content: Extract positive/negative sentiment from any publicly-accessible webpage or posted HTML/text document.

To extract document-level sentiment from a publicly-accessible webpage or posted HTML document, use the following API calls:

[URLGetTextSentiment](#)

Used for processing publicly-accessible Internet web pages.

[TextGetTextSentiment](#)

Used for processing uploaded text content.

[HTMLGetTextSentiment](#)

Used for processing uploaded HTML content.

To extract sentiment targeted towards a specific "target phrase" inside a publicly-accessible webpage, or posted HTML document, or text document:

Alchemy Sentiment Analysis API

API Call: TextGetTextSentiment

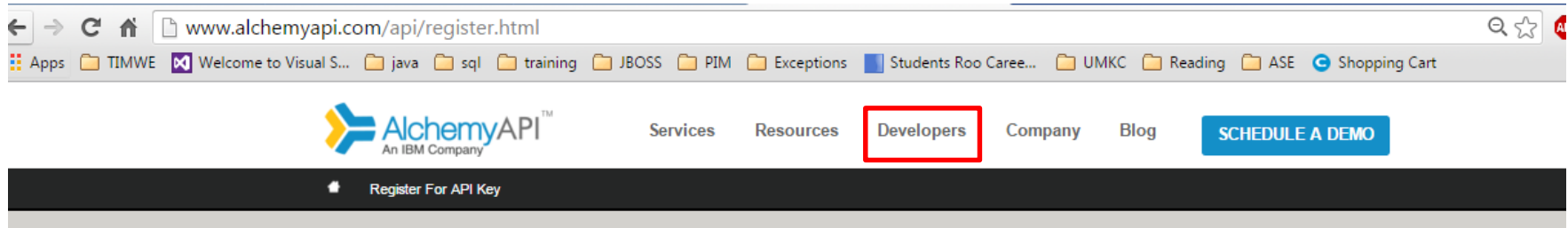
Description: The TextGetTextSentiment call is utilized to extract positive/negative sentiment from within a posted text document.

Endpoint: <http://gateway-a.watsonplatform.net/calls/text/TextGetTextSentiment>

Parameters:

HTTP ARGUMENT	PARAMETER DESCRIPTION
apikey	your private api key (required parameter)
text	Text document content (must be uri-argument encoded) (required parameter)
url	Text document URL (must be uri-argument encoded) (optional parameter, for response tracking purposes.)
outputMode	desired API output format

You can get a API key all the times on Developers.ServiceProviderSite !!



It's Easy to Get Your Free AlchemyAPI Key!

Start building smarter data apps right away. With a key, you get immediate access to:

- 1,000 daily transactions
- Technical email support
- All of our text and image analysis APIs

Ask [Support](#) a technical question or [upgrade now](#).

Fill out this form to get your free API key

What are you working on?*

Please Select: ▼

First Name*

Last Name*

Get Venues in app.js

```
'use strict';

// Declare app level module which depends on views, and components
angular.module('myApp', [])

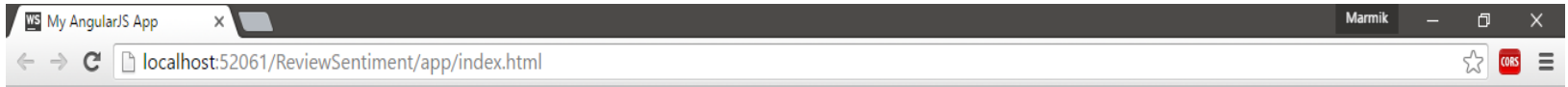
.controller('View1Ctrl', function ($scope, $http) {
    $scope.venueList = new Array();
    $scope.mostRecentReview;
    $scope.getVenues = function () {
        var placeEntered = document.getElementById("txt_placeName").value;
        var searchQuery = document.getElementById("txt_searchFilter").value;
        if (placeEntered != null && placeEntered != "" && searchQuery != null && searchQuery != "") {
            document.getElementById('div_ReviewList').style.display = 'none';
            //This is the API that gives the list of venues based on the place and search query.
            var handler = $http.get("https://api.foursquare.com/v2/venues/search" +
                "?client_id=QOENF1YHFTNPJ31DCF13ALLENJWOP5MTH13T1SA0ZP1MUOCI" +
                "&client_secret=ZH4CRZNEWBNTALAE3INIB5XGQOI12R4DT5HKAJLWKYE1LHOG" +
                "&v=20160215&limit=5" +
                "&near=" + placeEntered +
                "&query=" + searchQuery);
            handler.success(function (data) {

                if (data != null && data.response != null && data.response.venues != undefined && data.response.venues != null) {
                    for (var i = 0; i < data.response.venues.length; i++) {
                        $scope.venueList[i] = {
                            "name": data.response.venues[i].name,
                            "id": data.response.venues[i].id,
                            "location": data.response.venues[i].location
                        };
                    }
                }
            })
            handler.error(function (data) {
```


Get Tips in view1ctrl Controller of App.js

```
$scope.getReviews = function (venueSelected) {
    if (venueSelected != null) {
        //This is the API call being made to get the reviews(tips) for the selected place or venue.
        var handler = $http.get("https://api.foursquare.com/v2/venues/" + venueSelected.id + "/tips" +
            "?sort=recent" +
            "&client_id=QOENF1YHFTNPJ31DCF13ALLENJWOP5MTH13T1SA0ZP1MUOCI" +
            "&client_secret=ZH4CRZNEWBNTALAE3INIB5XG0QI12R4DT5HKAJLWKYE1LHOG&v=20160215" +
            "&limit=5");
        handler.success(function (result) {
            if (result != null && result.response != null && result.response.tips != null &&
                result.response.tips.items != null) {
                $scope.mostRecentReview = result.response.tips.items[0];
                //This is the Alchemy API for getting the sentiment of the most recent review for a place.
                var callback = $http.get("http://gateway-a.watsonplatform.net/calls/text/TextGetTextSentiment" +
                    "?apikey=d0e7bf68cdda677938e6c186eaf2b755ef737cd8" +
                    "&outputMode=json&text=" + $scope.mostRecentReview.text);
                callback.success(function (data) {
                    if (data != null && data.docSentiment != null) {
                        {
                            $scope.ReviewWithSentiment = {"reviewText" : $scope.mostRecentReview.text,
                                "sentiment":data.docSentiment.type,
                                "score":data.docSentiment.score };
                            document.getElementById('div_ReviewList').style.display = 'block';
                        }
                    }
                })
            }
        })
        handler.error(function (result) {
            alert("There was some error processing your request. Please try after some time.")
        })
    }
}
```


Search using Foursquare API

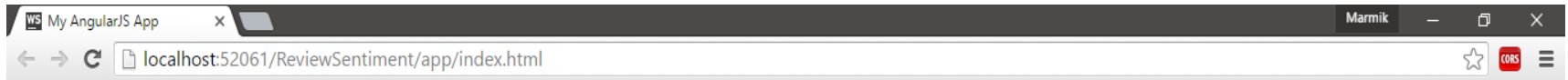


Search your place of interest!

<input type="text" value="Enter the place you want to search for"/>	<input type="text" value="Enter the search query"/>	<input type="button" value="Get Venues"/>
---	---	---

Top Five Picks From FourSquare Review API

Top 5 Results of Foursquare API



Search your place of interest!

<input type="text" value="kansas"/>	<input type="text" value="pizza"/>	<input type="button" value="Get Venues"/>
-------------------------------------	------------------------------------	---

Top Five Picks From FourSquare Review API

[Claflin Pizza Co.](#)

[Pizza Hut](#)

[Pizza Hut](#)

[Gambino's Pizza](#)

[Pizza Hut](#)

Sentiment Analysis using Alchemy API

WS My AngularJS App

Marmik

localhost:52061/ReviewSentiment/app/index.html

☆ CORB

Search your place of interest!

kansas

pizza

Get Venues

Top Five Picks From FourSquare Review API

[Claflin Pizza Co.](#)

[Pizza Hut](#)

[Pizza Hut](#)

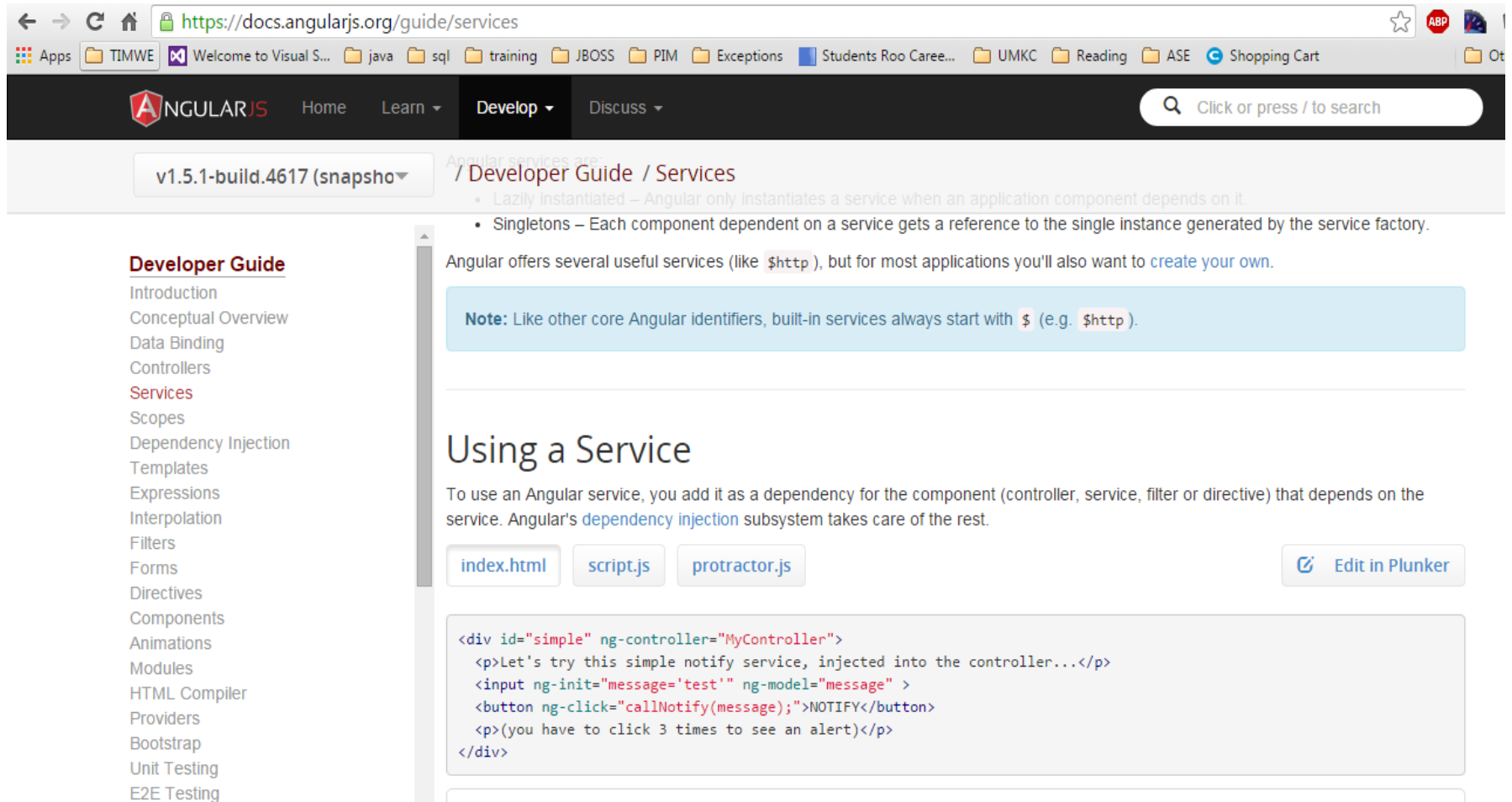
[Gambino's Pizza](#)

[Pizza Hut](#)

Sentiment Analysis From Using Alchemy API	
Most recent review	It's a great place to eat and good pizza too!!!!!!:)
Sentiment of the review	positive
Sentiment Score of the review	0.951734

It's a great place to eat and good pizza too!!!!!!:)

Angular Service



The screenshot shows the AngularJS Developer Guide page for Services. The browser address bar shows the URL <https://docs.angularjs.org/guide/services>. The page has a dark navigation bar with the AngularJS logo and links for Home, Learn, Develop, and Discuss. A search bar is on the right. Below the navigation bar, there's a version selector showing 'v1.5.1-build.4617 (snapsho)'. The main content area is titled '/ Developer Guide / Services' and lists two types of services: 'Lazily instantiated' and 'Singletons'. A note states that built-in services always start with '\$'. The page also features a sidebar with a table of contents, a 'Using a Service' section with an example code snippet, and buttons for 'index.html', 'script.js', 'protractor.js', and 'Edit in Plunker'.

AngularJS Home Learn Develop Discuss

Search Click or press / to search

v1.5.1-build.4617 (snapsho)

/ Developer Guide / Services

- Lazily instantiated – Angular only instantiates a service when an application component depends on it.
- Singletons – Each component dependent on a service gets a reference to the single instance generated by the service factory.

Angular offers several useful services (like `$http`), but for most applications you'll also want to [create your own](#).

Note: Like other core Angular identifiers, built-in services always start with `$` (e.g. `$http`).

Using a Service

To use an Angular service, you add it as a dependency for the component (controller, service, filter or directive) that depends on the service. Angular's [dependency injection](#) subsystem takes care of the rest.

[index.html](#) [script.js](#) [protractor.js](#) [Edit in Plunker](#)

```
<div id="simple" ng-controller="MyController">
  <p>Let's try this simple notify service, injected into the controller...</p>
  <input ng-init="message='test'" ng-model="message" >
  <button ng-click="callNotify(message);">NOTIFY</button>
  <p>(you have to click 3 times to see an alert)</p>
</div>
```

Angular Dependency Injection

The screenshot shows the AngularJS documentation website. The browser address bar displays `https://docs.angularjs.org/guide/di`. The page header includes the AngularJS logo, navigation links (Home, Learn, Develop, Discuss), and a search bar. Below the header, a breadcrumb trail shows the path `/ Developer Guide / Dependency Injection`. A sidebar on the left lists the contents of the Developer Guide, with `Dependency Injection` highlighted. The main content area begins with a definition of Dependency Injection (DI) as a software design pattern. It then explains the role of the Angular injector subsystem. The section `Using Dependency Injection` follows, describing how DI is pervasive in Angular and how it is used with `run` and `config` blocks. A list of four bullet points details the types of components that can be injected into these blocks. The section concludes with a reference to the `Modules` section for more details. The `Factory Methods` section is partially visible at the bottom.

← → ↺ 🏠 <https://docs.angularjs.org/guide/di> ☆ ABB

Apps TIMWE Welcome to Visual S... java sql training JBOSS PIM Exceptions Students Roo Caree... UMKC Reading ASE Shopping Cart

ANGULARJS Home Learn Develop Discuss 🔍 Click or press / to search

v1.5.1-build.4617 (snapsho) / Developer Guide / Dependency Injection [Improve this Doc](#)

Developer Guide

- Introduction
- Conceptual Overview
- Data Binding
- Controllers
- Services
- Scopes
- Dependency Injection**
- Templates
- Expressions
- Interpolation
- Filters
- Forms
- Directives
- Components
- Animations
- Modules
- HTML Compiler
- Providers
- Bootstrap
- Unit Testing
- E2E Testing
- Using \$location

Dependency Injection (DI) is a software design pattern that deals with how components get hold of their dependencies.

The Angular injector subsystem is in charge of creating components, resolving their dependencies, and providing them to other components as requested.

Using Dependency Injection

DI is pervasive throughout Angular. You can use it when defining components or when providing `run` and `config` blocks for a module.

- Components such as services, directives, filters, and animations are defined by an injectable factory method or constructor function. These components can be injected with "service" and "value" components as dependencies.
- Controllers are defined by a constructor function, which can be injected with any of the "service" and "value" components as dependencies, but they can also be provided with special dependencies. See [Controllers](#) below for a list of these special dependencies.
- The `run` method accepts a function, which can be injected with "service", "value" and "constant" components as dependencies. Note that you cannot inject "providers" into `run` blocks.
- The `config` method accepts a function, which can be injected with "provider" and "constant" components as dependencies. Note that you cannot inject "service" or "value" components into configuration.

See [Modules](#) for more details about `run` and `config` blocks.

Factory Methods

Mashup

A mashup, in web development, is a web page, or web application, that uses content from more than one source to create a single new service displayed in a single graphical interface

Example:

- <http://www.makeuseof.com/tag/10-unique-google-maps-mashups-explore-information/>
- <http://mashable.com/2008/05/14/17-google-maps-mashups-to-waste-away-your-day/>

Thank You