

“Developing a search site using Angular JS & ElasticSearch”

A

Project Report

*Submitted in partial fulfillment of the
Requirements for the award of the degree of*

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

By

Name	Roll No.
G.S. Aditya Rao	R110211014
Yuvraj Gupta	R110211050
Herambh Bhatnagar	R110211054

Under the guidance of

Mr. Anil Kumar



Department of Computer Science & Engineering

Centre for Information Technology

University of Petroleum & Energy Studies

Bidholi, Via Prem Nagar, Dehradun, UK

April – 2015



The innovation driven
E-School

CANDIDATE'S DECLARATION

We hereby certify that the project work entitled “**Developing a search site using Angular JS & ElasticSearch**” in partial fulfilment of the requirements for the award of the Degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING with specialization in Cloud Computing & Virtualization Technology and submitted to the Department of Computer Science & Engineering at Center for Information Technology, University of Petroleum & Energy Studies, Dehradun, is an authentic record of my/ our work carried out during a period from **January, 2015** to **April, 2015** under the supervision of **Mr. Anil Kumar (Assistant Professor)**.

The matter presented in this project has not been submitted by me/ us for the award of any other degree of this or any other University.

G.S. Aditya Rao
R110211014
Yuvraj Gupta
R110211050
Herambh Bhatnagar
R110211054

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date: 12th April 2015

Mr. Anil Kumar
Project Guide

Dr. Amit Agarwal
Program Head - CCVT
Center for Information Technology
University of Petroleum & Energy Studies
Dehradun – 248 001 (Uttarakhand)

ACKNOWLEDGEMENT

We wish to express our deep gratitude to our guide **Mr. Anil Kumar**, for all advice, encouragement and constant support he has given us throughout our project work. This work would not have been possible without his support and valuable suggestions.

We sincerely thank to our respected Program Head of the Department, **Dr. Amit Agarwal**, for his great support in doing our project.

We are also grateful to **Dr. Manish Prateek, Associate Dean** and **Mr. Kamal Bansal** Dean CoES, UPES for giving us the necessary facilities to carry out our project work successfully.

We would like to thank all our **friends** for their help and constructive criticism during our project work. Finally we have no words to express our sincere gratitude to our **parents** who have shown us this world and for every support they have given us.

Name	G.S .Aditya Rao	Yuvraj Gupta	Herambh Bhatnagar
-------------	------------------------	---------------------	--------------------------

Roll No.	R110211014	R110211050	R110211054
-----------------	-------------------	-------------------	-------------------

ABSTRACT

World has seen an intense change in seeking speed and everybody needs a decent interface alongside precision. A web internet searcher is a product framework that is intended to hunt down data on the World Wide Web. The indexed lists may be a blend of pages, pictures, and different sorts of documents.

Some web crawlers likewise mine information accessible in databases or open registries. Dissimilar to web registries, which are kept up just by human editors, web indexes additionally keep up continuous data by running a calculation on a web crawler.

The reason for the project is to produce an efficient searching (site) which incorporates effective searching alongside an alluring front-end and a solid back-end. We will be utilizing ElasticSearch for the back-end as it is a gigantic report store and for the front-end GUI creation we will be utilizing Angular JS.

Our project is identified with our area as taking care of Big Data is an immense test which opens up another test of effective and compelling searching which gives vital insights to the business. The search site developed would help users find, discover data precisely and results will be indicated inside seconds of extensive database.

Technologies Used: ElasticSearch, Angular JS, Node.js, JSON files, HTML, CSS, AJAX, Java Script.

TABLE OF CONTENTS

S.No.	Contents	Page No
1.	Introduction	1
	1.1. Problem Statement	1
	1.2. Objective	2
	1.3. PERT Chart Legend	2
2.	Literature Review	3
3.	Project Requirement Analysis	4
4.	Methodology	5
	4.1 ElasticSearch	6
	4.1.1 Introduction to ElasticSearch	6
	4.1.2 History of ElasticSearch	7
	4.1.3 What ElasticSearch is not	7
	4.1.4 ElasticSearch Node Information	8
	4.2 AngularJS	9
	4.2.1 Introduction to AngularJS	9
	4.2.2 History of AngularJS	9
	4.2.3 Structure of AngularJS	10
	4.2.4 Features of AngularJS	11
	4.2.5 Why AngularJS	11
	4.2.6 Example of AngularJS Ease	13
	4.3 Analysis	14
	4.3.1 General Analysis based on framework	14
	4.3.2 Analysis based on Growth	15
	4.3.3 Analysis based on Review	16
	4.3.4 Analysis based on Regional preference	17
	4.4 Selection of Framework	18
	4.5 Elasticsearch on Google Compute Engine	19
	4.6 Developing Front End	26
	4.7 Connecting Front End (AngularJS) with ElasticSearch	31

5. Diagrams	33
5.1 Working of ElasticSearch	33
5.2 DFD (Data Flow Diagram)	33
5.3 Class Diagram	34
5.4 Activity Diagram	34
6. Snapshots	35
7. Conclusion	44
8. Definition	45
9. Bibliography	46

List of Figures

SNo.	Title	Page No.
1	PERT Chart	2
2	Analysis based on Framework	14
3	Analysis based on Growth	15
4	Analysis based on Review	16
5	Analysis based on Regional Preference	17
6	Configuration Options	21
7	Working of ElasticSearch	33
8	DFD (Data Flow Diagram)	33
9	Class Diagram	34
10	Activity Diagram	34

1. Introduction

Site Search is helpful to e-business and content based sites alike. An incredible search site can help you hold clients, expand deals, track failing to meet expectations pages and keywords and considerably more.

Benefits:

Site Search Benefits Your Consumers:

- It fulfills the "I need it now" issue throughout today's searchers
- Consumers expect searching on today's sites
- Reduces skip and way out rates and expands time on site
- Can help recommend related content the client is keen on yet didn't know to search.

Great For E-Commerce

On e-trade locales, up to 30 % of clients will utilize the pursuit box and show purpose to purchase by writing in hunt down item names, item codes and item classes. Extraordinary e-trade destinations use webpage look with included highlights that expand online customer's expectation to purchase. E-Commerce locales can utilize sifting choices to tight down "Garments" to "Men's" and afterward much further to "Dress Pants."

- Improved Sales
- Increased Time on Site
- Customer Retention & Loyalty

Allows Better Analytics

By what means would you be able to demonstrate that site search is advantageous? In what capacity would it be able to help the site proprietor? The answer ... Analytics. Site search emphasize that can help you dive more profound into the psyches of your customers and enhance your site with your discoveries.

1.1 Problem Statement

How about we comprehend this through a case:

You have to comprehend thousands or even a great many lines of information, and you have a brief while to do it in. The information may originate from your group, in which case maybe you're now acquainted with what its measuring and what the outcomes are liable to be. Then again it may originate from another group, or perhaps a few groups

immediately, and be totally new. You have an assignment of seeking data among tons and huge amounts of information. To recover precise results & get comes about rapidly we are creating quest website application for a site utilizing most recent open source advancement.

1.2 Objective

The principle goals of this project are:

1. Recognize the subject for which the search site would be created.
2. Making sense of all the important information to build a search site ie.all related pages..
3. Making an information stockroom that stores data and push data into ElasticSearch.
4. Making an Attractive front-end utilizing Angular JS.
5. Internal database search insights as needed by users.
6. Viably, effectively, exquisitely, precisely giving important correspondence information.

1.3 PERT Chart Legend

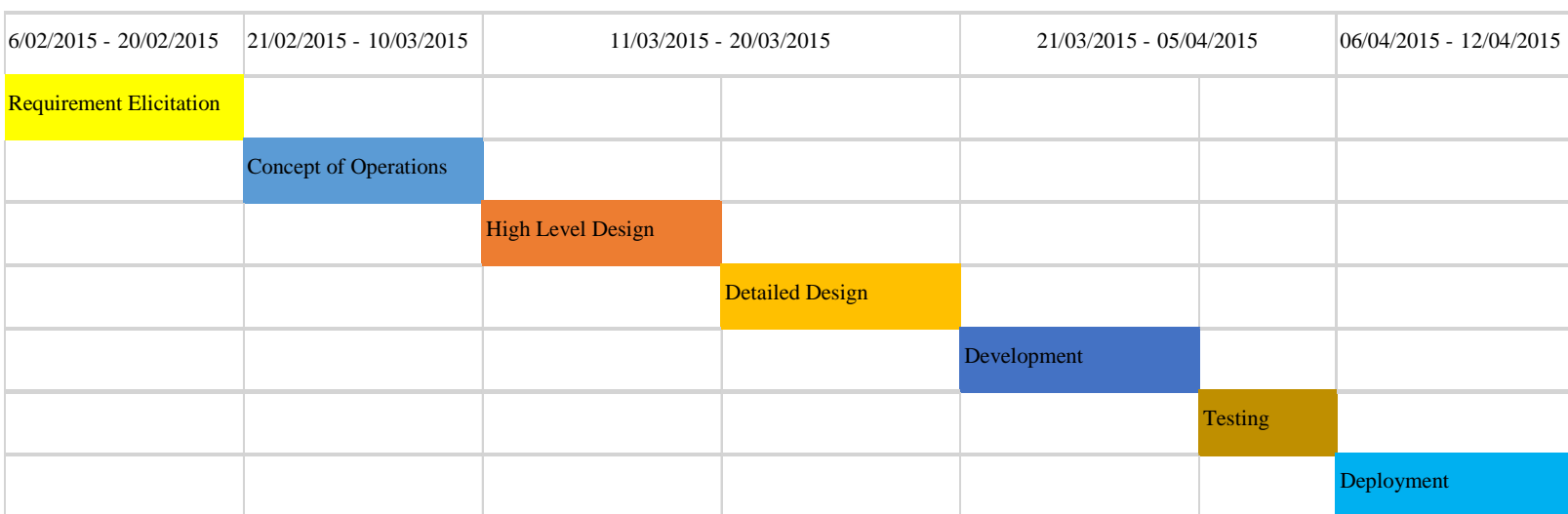


Fig 1.1 PERT Chart

2. Literature Review

With our bolster memberships, deploy significantly speedier. Shay Banon made Compass in 2004. While pondering the third form of Compass he understood that it would be important to revise huge parts of Compass to "make an adaptable search solution". So he created "a solution developed from the beginning and distributed" and utilized a typical interface, JSON over HTTP, suitable for programming dialects other than Java too. Shay Banon released the first form of Elasticsearch in February 2010.

In June 2014, the organization reported bringing \$70 million up in a Series C financing round, only 18 months after framing the organization. The round was driven by New Enterprise Associates (NEA). Extra funders incorporate Benchmark Capital and Index Ventures. This round conveys aggregate financing to \$104M.

Direction was likewise taken from Mr. Anil Kumar, Assistant Professor, University of Petroleum and Energy Studies the subject master in the area of Data mining and other Data recovering methods.

Banon, Shay. "The Elastic Search Solution". <http://www.elasticsearch.com/>. Web. 29 January 2013

In this article Shay Banon clarifies the force of Elasticsearch over the relational database and the under lying building design that includes the Apache Lucene. Additionally he talked about the Full Text Search offer that is not accessible in databases.

Khan, Rashid. "Make Sense Of Mountain Logs". <http://rashidkpc.github.io/Kibana/>. Web. 19 November 2012

In this article Rashid Khan clarifies the utilization cases for diverse visualizations in Kibana with simple techniques taking care of procedures.

Biplap. "Whitepaper: Enriching UI with AngularJS". <http://blog.aspiresys.com/whitepaper-enriching-ui-with-angularjs/> Web. 20 November 2014

In this article focus is on AngularJS that that permits to art refined and versatile UI with great ease using its structured framework design and vast accumulation of UI widgets.

3. Project Requirement Analysis

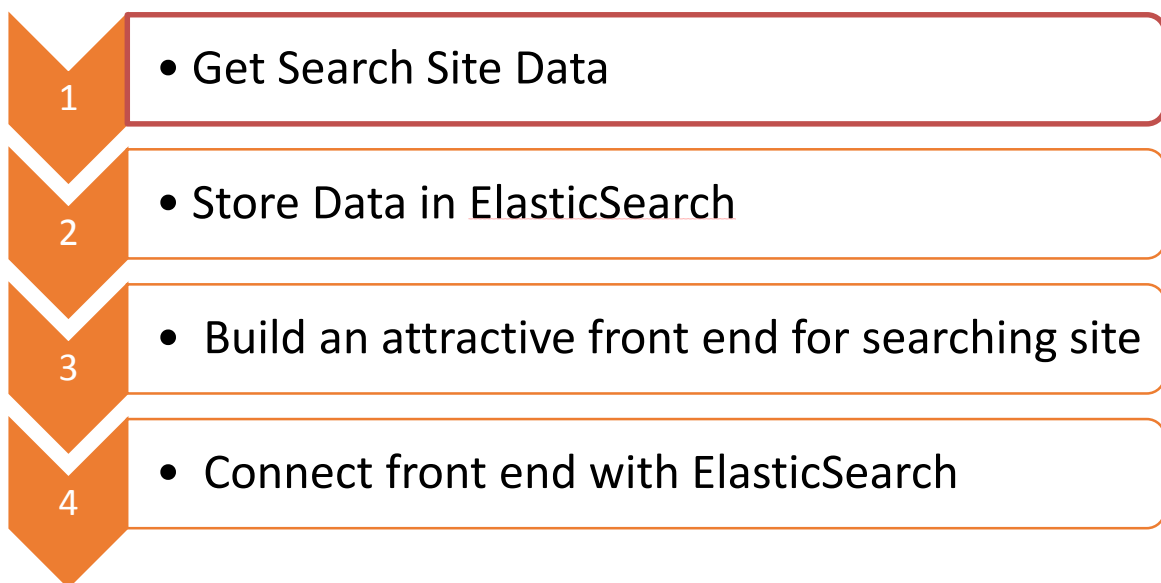
- 1.** Analysis of different Front End Framework.
- 2.** Finalizing the Front End Framework according to need.
- 3.** Getting Site Search Data in JSON format.
- 4.** Deploying of Elasticsearch on Google Compute Engine.
- 5.** Storing Data in ElasticSearch on Google Compute Engine.
- 6.** Creation of multiple instances for Distributed computing.
- 7.** Building an attractive front end utilizing AngularJS
- 8.** Connecting Front End with ElasticSearch.
- 9.** Obtaining the results of the remote host on the local host via tunneling.
- 10.** Providing real time search using Open Source Technologies.

4. Methodology

Technologies used-

- ElasticSearch – v.1.3.4
- Angular JS
- Node.js
- JSON files
- HTML
- CSS
- AJAX
- Java Script.
- Git Bash

The methodology includes the following steps which are shown or depicted below:



4.1 ElasticSearch

With the prologue to the portability gadgets and the informal communication lead to the evolution of Big Data. An extraordinary measure of both structured and unstructured information is being created. The problem statement for utilizing the ElasticSearch starts the line of the speed, volume, mixture of information. Gartner characterizes Big Data as high volume, speed and mixed bag data resources that request savvy, inventive types of data transforming for improved knowledge and choice making. We work with numerous clients crosswise over numerous commercial enterprises with various particular information challenges, however in conversing with such a variety of clients, we are additionally ready to see examples rise on specific sorts of information and the quality that could convey to a business. This produced information can be of an incredible use as it can give an extraordinary business resource exceptionally created structure the long range interpersonal communication locales. Elasticsearch is a search server based on Apache Lucene. It gives a conveyed, multitenant-competent full-content web crawler with a RESTful web interface and outline free JSON archives. Elasticsearch is created in Java and is distributed as open source under the terms of the Apache License.

4.1.1 Introduction of ElasticSearch

- Elasticsearch is a search server based on Lucene. It gives a distributed, multitenant-competent full-content web search tool with a RESTful web interface and schema-free JSON records.
- ElasticSearch is created in Java and is released as open source under the terms of the Apache License. Kibana is an open source (Apache Licensed), browser based analytics and search dashboard for ElasticSearch.
- Elasticsearch is an open-source, distributed web application that runs on top of Lucene, and it is written in Java, and it uses REST API.
- Apache Lucene is the best open-source web index, and likely one of the best web search tools accessible, and holds its own even when analyzed against the industry choices.
- It provides the realistic potential for you to run your own search engine service (like a Bing or a Google) yet with say, private, touchy, or classified information/records that you don't want on general web.
- It has some extremely cool properties and abilities concerning operations that include numerous nodes.
- It scales amazingly effortlessly. It has its own optimized binary protocol and makes its own "internal network".

In short, it can be thought of as "*search engine software*".

4.1.2 History of ElasticSearch

Initially there was a search application project called Apache Compass, which was fundamentally taken a shot at by @kimchy.

Compass additionally depended on Lucene, but was not distributed.

Kimchy chose to compose elasticsearch to be distributed from the get go, thus you could say it was built with the cloud in mind.

Add more servers and they play into a single unit pleasantly and they know how to cooperate to split up the work load (and search queries can be resource intensive and expensive in terms of memory/disk requirements).

4.1.3 What ElasticSearch is not?

- **It is NOT safe as an essential steady data store**
 - Meaning – you ought not to believe it as a “system of record”.
 - Always be prepared to reload from scratch, if there should be an occurrence of data corruption.
 - “Don't let yourself get a connected to anything you are not eager to leave in 30 seconds flat if you feel the heat around the corner.” -Neil McCauley *Heat*
 - Although Neil's to-a-fault discipline doesn't make a difference to everything in life, ElasticSearch is a unique little something that it functions well in the event that you apply that philosophy: always be prepared to drop and reload your data if something goes frightfully wrong later on.
- **It's not extremely well documented**
 - There is a ton of documentation, however it is now and then hard to parse/read the sentences because of syntactic lapses and so on.
 - Plus there is great deal of language when you begin discussing about analyzers, etc. You need to do a lot of examination to make utilization of what documentation there is.

4.1.4 ElasticSearch Nodes Information

Each machine is a node in the cluster.

You've heard this phrasing before on the off chance that you have utilized Hadoop, Zookeeper, or any number of appropriate frameworks.

Nodes can have "types" (master, data and client)

Data nodes need disk and memory

Client nodes need memory

Master nodes need stability and to not be "stressed out" or "upset"

The simplest cluster: one node

- It's the master
- It's the data node
- It's the client node

4.2 AngularJS

AngularJS is a toolset for building the framework most suited to your application development. It is completely extensible and functions well with different libraries.

HTML is extraordinary for declaring static documents, yet it flounders when we attempt to utilize it for pronouncing dynamic views in web-applications. AngularJS gives you a chance to expand HTML vocabulary for your application. The subsequent environment is exceptionally expressive, intelligible, and fast to create.

Different structures manage HTML's weaknesses by either abstracting away HTML, CSS, and/or JavaScript or by giving a basic approach to controlling the DOM. Neither of these address the root issue that HTML was not intended for dynamic views.

4.2.1 Introduction of AngularJS

- Developed and maintained by Google
- MVC Javascript Framework for Rich Web Application Development
- Hot new front-end JavaScript framework
- Rapidly develop powerful, responsive web apps
- Philosophy like jQuery
- Simplicity
- Enhance HTML, work with it not around it
- Testable, maintainable, extendable

4.2.2 History of AngularJS

Developed in 2009 by MiškoHevery and Adam Abrons for commercial purposes, however later Angular as an open-source library.

Hevery, who works at Google, keeps on developing and maintain the library. Version 1.0 released in December 2012.

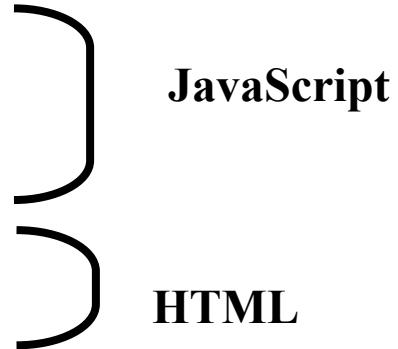
Shouldn't we think about WebToolkit?

WebToolkit assembles Java down to JavaScript and was utilized by the Google broadly. With the ascent of HTML5, CSS3, and JavaScript, Google is changing directions.

4.2.3 Structure of AngularJS

MVC

- **Model**
 - The data
- **Controller**
 - The behavior
 - Modifying / updating the models
- **View**
 - The interface
 - How the data is presented to the user



Views

- Make utilization of special ng attributes (directives) on the HTML elements
 - **ng-app**
 - Determines which piece of the page will utilize AngularJS
 - If given a value it will load that application module
 - **ng-controller**
 - Determines which Javascript Controller ought to be utilized for that part of page.
 - **ng-model**
 - Determines what model the estimation of an data field will be bound to
 - Used for two-way binding

Controller

- **Function that takes no less than one parameter: \$scope**
 - Function is a constructor
 - Ex:
 - `function MyCtrl($scope) { ... }`
- **\$scope**
 - JavaScript object
 - Contains information (i.e. models) and methods (i.e. functions)
 - Can include own properties
 - `$scope.<my new property> = <value>;`
- **The \$scope variable – Link your controllers and view**
- **Dependency Injection**

- Pass the modules and services that is required as parameters
- In the past case \$scope is a service that will be injected
- Can be gone as an array of strings to the controller function too
- Prevents slips when performing adjustment
- **Other useful services**
 - **\$http**
 - Used to handle Ajax calls
 - Wrappers around jQuery

Models

- Properties on the Controller's \$scope object
- Standard JavaScript values
- Can be utilized to divide the application into parts
- Application module can incorporate alternate modules by posting them as dependencies

4.2.4 Features of AngularJS

- **Templating**
- **Databinding**
- **Routing**
- **MVC**
- **Server-Side Communication**
- **Dependency Injection**
- **Extensibility ("Directives")**
- **Allows for testing**
- **Deep Linking (Map URL to route Definition)**

4.2.5 Why AngularJS

Backed by Google- Actively kept-up and maintained

- Angular is constructed and kept up by devoted Google engineers.
- This implies you not just have an extensive open group to gain from, however you additionally have talented, very accessible architects tasked to help you get your AngularJS inquiries replied.
- AngularJS happened to institutionalize web application structure and give a future format to how client side applications ought to be created.

Since AngularJS is built by Google, you can make certain that you're managing productive & solid reliable code that will scale with your task.

Comprehensive feature set

- Angular a complete answer for fast front-end development.
- No different plugins or frameworks are necessary to assemble a data-driven web application.
- Overview of Angular's highlights

REST Easy

- Restful actions are rapidly turning into the standard for communication from the server to customer.
- In one line of JavaScript, you can rapidly converse with server and get the data you have to connect with your web site.
- Angular a complete solution for front end advancement.

Extends HTML

- Most sites developed today are a goliath arrangement of <div> tags.
- You need to make broad and thorough CSS classes to express the intention of every object in the Document Object Model.
- With AngularJS, you can work your HTML like XML, giving you unlimited possibilities for tags and attributes.
- Angular achieves this through its HTML compiler and utilization of directives to trigger behaviors taking into account the newly created syntax.

Makes HTML your Template

- You can rapidly get a hand on the bracket syntax of Angular's templating engine, because *it's equitable with HTML*.
- Angular travels the DOM for the templates, which house the directives specified above.
- The templates are then gone to the AngularJS compiler as Document Object Model elements, which can be broadened, executed or reused.
- This is key taking into account direct control and manipulation of the DOM.

Enterprise-level Testing

- AngularJS obliges no extra systems, frameworks or plugins, including testing.
- Easy to learn Angular's JS unit-testing API which helps you through executing your tests in as reality to the actual condition of your production as possible.

4.2.6 Example of AngularJS Ease

HTML

```
<p>Hello World!</p>
```

Javascript

```
<p id="greeting1"></p>
<script>
varisIE = document.attachEvent;
varaddListener = isIE
? function(e, t, fn) {
e.attachEvent('on' + t, fn);}
: function(e, t, fn) {
e.addEventListener(t, fn, false);}
addListener(document, 'load', function(){
var greeting = document.getElementById('greeting1');
if (isIE) {
greeting.innerText = 'Hello World!';
} else {
greeting.textContent = 'Hello World!';
}});</script>
```

JQuery

```
<p id="greeting2"></p>
<script>
$(function(){
$('#greeting2').text('Hello World!');
});
</script>
```

AngularJS

```
<p ng:init="greeting = 'Hello World!'">{{ greeting }}</p>
```

4.3 Analysis

4.3.1 General Analysis based on framework

We observed different frameworks which incorporated frameworks like Angular JS, Backbone JS, Ember JS, ExtJS, KendoUI, CanJS, SammyJS, Agility JS, Maria, Serenade JS, Rapid JS, Epitome, Soma JS, Stapes JS, Plastron JS these are the various frame works which we have explored in the observation of different frameworks. The exploration was in light of the different parameters like Application Readiness which came out as 81% Value Proposition came out to as 86% and on the basis of various voting it got 827 votes. Beating the research results of alternate frameworks like Backbone JS, Knockout, and EmberJS. Thus as seen in the result figure below we can discover that Angular JS turned out to be the best fit solution according to the latest market trend which made us pick Angular JS as the Framework. We chose Different Frameworks and analyzed them and angular JS was a complete winner.

Total number of participants: 1566


















Option	Adoption Readiness	Value Proposition	Votes	Details
AngularJS	81%	86%	827	Heatmap 
Backbone.js	79%	74%	717	Heatmap 
Knockout	73%	73%	435	Heatmap 
Ember.js	67%	73%	393	Heatmap 
ExtJS	73%	72%	332	Heatmap 
Kendo UI	61%	73%	189	Heatmap 
Spine	59%	61%	141	Heatmap 
CanJS	59%	64%	113	Heatmap 
Sammy.js	56%	62%	110	Heatmap 
Agility.js	57%	56%	92	Heatmap 
Maria	50%	50%	81	Heatmap 
Serenade.js	45%	55%	74	Heatmap 
rAppid.js	48%	58%	73	Heatmap 
Epitome	45%	56%	73	Heatmap 
soma.js	43%	56%	71	Heatmap 
Stapes.js	45%	55%	70	Heatmap 
PlastronJS	42%	56%	70	Heatmap 

Fig 4.1 Analysis based on framework

4.3.2 Analysis Based On Growth:

Based on Growth, how the distinctive technologies advanced over the time it is seen and watched that the growth rate of Angular JS here developed as the Clear winner with an evolution of 379 % while the other contemporary framework's like the Spine, Batman, Backbone, Knockout, Ember, Meteor, CanJS with 19%, 53%, 64%, 76%, 104%, 130%, 133% respectively.

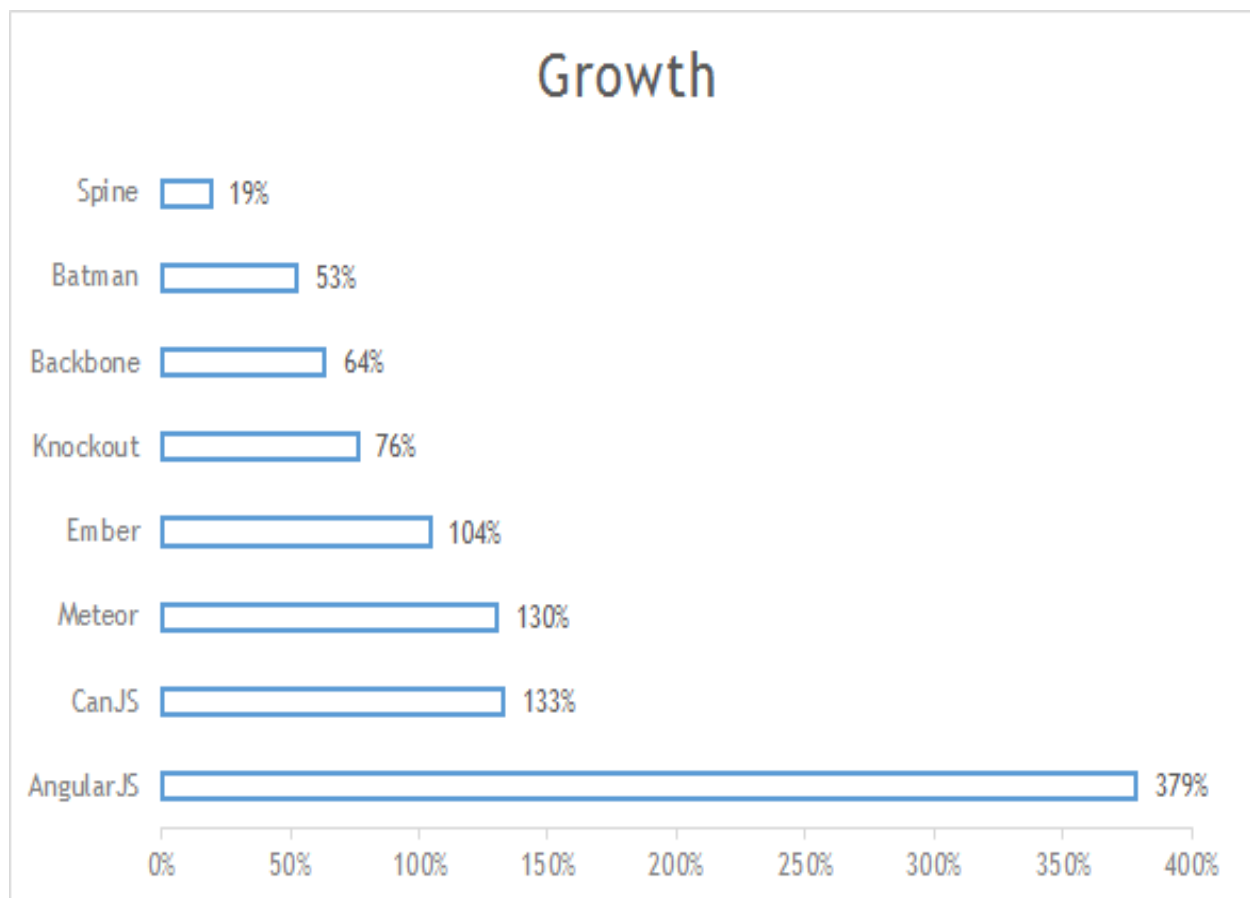


Fig 4.2 Analysis based on growth

4.3.3 Analysis based on Review

Indeed, even this turned out to be agreeable to AngularJS as we contrasted Angular and BackboneJS and EmberJS by its stars on GitHub, Third party modules, Stack overflow questions, YouTube Results, GitHub contributors, Chrome Extension Users. Thus the most sought framework proved out to be the AngularJS.

Metric	AngularJS	Backbone.js	Ember.js
Stars on Github	27.2k	18.8k	11k
Third-Party Modules	800 <i>ngmodules</i>	236 <i>backplugs</i>	21 <i>emberaddons</i>
StackOverflow Questions	49.5k	15.9k	11.2k
YouTube Results	~75k	~16k	~6k
GitHub Contributors	928	230	393
Chrome Extension Users	150k	7k	38.3k

Fig 4.3 Analysis based on review

4.3.4 Analysis Based on Regional Preference

When the current progressing projects were connected here additionally when we discuss of the Australia ,Angular JS had a share of the 38 % as compared to the 26 % of backbone JS and 16 % of ember JS and 20 % of nodeJS. When talking of the great Britain angular JS proportion was 35 % as compared to the 23 % of node JS and 33 % of Backbone JS. While in USA Angular JS was 33% as compared to the Backbone JS with the majority of 39 %.but overall majority helped us to choose Angular JS.

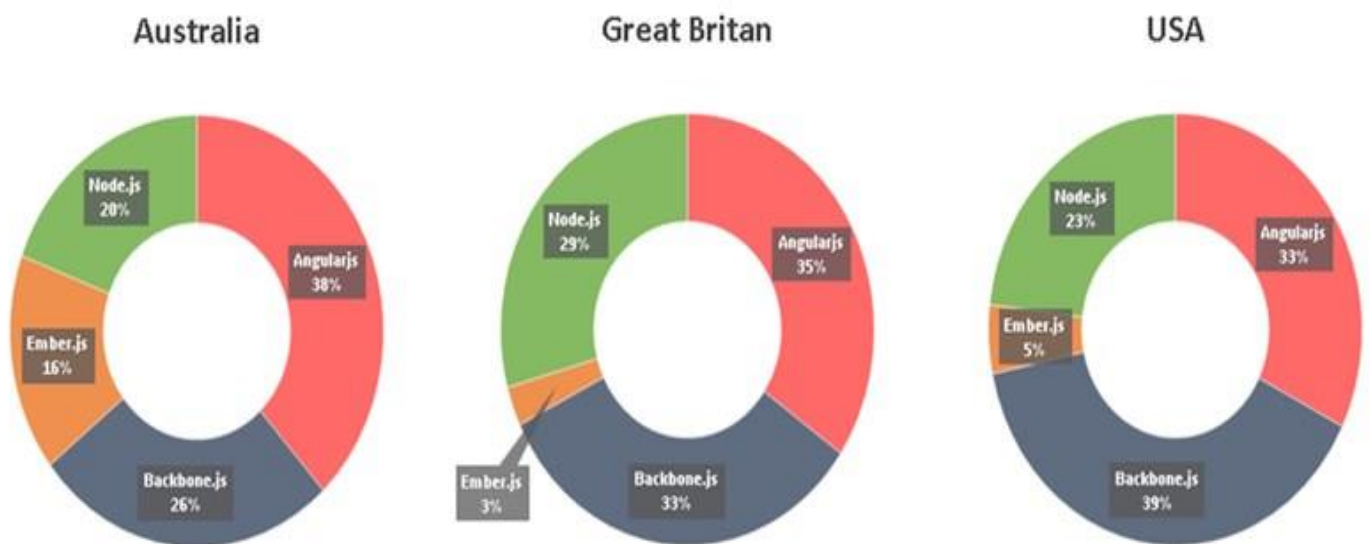


Fig 4.4 Analysis based on regional preference

4.4 SELECTION OF FRAMEWORK

We choose Angular JS (because of following features):

- **Reusable Components**
 - **Lightweight**
 - **Extensibility & Maintainability**
 - **Like jQuery, it doesn't get in the way**
 - **Get started in minutes**
 - **Large & quickly growing community**
 - **Modern (REST, SPAs)**
 - **Open Source**
-
- These are the basic principles that guide AngularJS to make an effective, performance-driven and viable front-end codebase.
 - As long as you have a source to store data, AngularJS can do the majority of the difficult work on the client, while giving a rich, quick experience for the end user.
 - “HTML? Build UI Declaratively! CSS? Animations! JavaScript? Use it the plain old way!”

4.5 Elasticsearch on Google Compute Engine

Introduction

Searching and examining your data continuously with Elasticsearch is presently accessible as a single click installation on Google Compute Engine (GCE). Google teamed up with Elasticsearch to take the Lucene-based open source product you know and love and made it simple to use on cloud infrastructure you trust.

GCE is the Infrastructure-as-a-Service (IaaS) part of the Google Cloud Platform, which offers a suite of cloud-based services to help clients develop and run high number of applications. Utilizing virtual machines facilitated on Google's infrastructure, GCE lets clients run substantial scale workloads with the unwavering quality and execution of Google's worldwide fiber system.

Elasticsearch on GCE is a performant option for enterprises which are looking for a cloud-based, adaptable answer to hunt and analyze data progressively. From unified logging, to effective search applications, to operational analytics, Elasticsearch on GCE provides you a chance to just point, click, and provision Elasticsearch cluster immediately. The integration additionally incorporates Marvel, the Elasticsearch monitoring product that helps you keep a heartbeat on your deployment and enhance utilization of Elasticsearch.

Click-to-deploy Elasticsearch on Google Compute Engine

In minutes raise an Elasticsearch cluster on Google Compute Engine. As a matter of course, the finished deployment incorporates 3 servers arranged in a cluster with the accompanying software installed:

- Debian 7 Linux
- Java OpenJDK 1.7
- Elasticsearch 1.4
- Elasticsearch Marvel – monitoring, administration and management service.

When we deploy a cluster, we can choose the cluster size, type of machine, size of disk, and other available settings. After the cluster is deployed and running, we can utilize SSH to connect with available instances for further design, testing and configuration.

Quickly deploy Elasticsearch on Compute Engine

The Google Developers Console gives a basic interface that can be used to rapidly design the Elasticsearch deployment.

1. In the Elasticsearch click-to-deploy page in the Google Developers Console, indicates the setup settings used for configuration or utilize by default given values.

Field	Default	Description
Zone	us-central1-a	Compute Engine zone for the server.
Number of nodes	3	The number of server nodes in the cluster.
Server node machine type	n1-standard-1	Compute Engine machine type for the server.
Install Marvel monitoring agent from Elasticsearch	true	Select to install Elasticsearch Marvel . Marvel is monitoring and management software that will run on your cluster and provides dashboards with metrics on cluster operation.

Field	Default	Description
Advanced option		
Data disk type	Standard Persistent Disk	The type of disk that stores the Elasticsearch indices and other data. Options include Standard Persistent Disk or SSD Persistent Disk
Data disk size in GB	100	The size of the disk that stores the Elasticsearch indices and other data.
Cluster name	elasticsearch-cluster	The name assigned to each member of your Elasticsearch cluster. Cluster name is used for the auto-discovery of new nodes.
Elasticsearch server name prefix	elasticsearch-	By default, the server name is created by appending four random, alphanumeric characters to this prefix.
Network name	default	The name of an existing Compute Engine network that the Elasticsearch cluster belongs to.

Fig 4.5 Configuration Option Details

2. Click **Deploy Cluster**.
3. Optional: Install the Cloud SDK to have the ability to connect with your Elasticsearch servers when deployment wraps up.

The deployment ought to take less than couple of minutes. On completion of deployment, you can click **View your Elasticsearch instances in the Developers Console** to see the running virtual machine instances and connect with them over SSH from your browser.

On the off chance that you introduced the Cloud SDK, you can likewise connect from CLI. To start with, determine the name of instance using the `gcloud compute` command:

```
$ gcloud compute instances list
```

Now you can SSH into the instance:

```
$ gcloud compute ssh --zone=<zone><instance-name>
```

When you're finished with your development stack, click the **Delete cluster** button in the Developers Console. This will close down all instances and delete related data disks.

Connect to your Elasticsearch cluster

To get to your Elasticsearch cluster, you can SSH to any instance and specifically communicate with the cluster using curl commands. Also, you can tunnel requests from your local machine to the cluster over SSH. Note: Installation of the **Cloud SDK** is obliged to utilize this SSH tunneling .On creation of the SSH tunnel, navigate to **http://localhost:9200** which returns:

```
{
  "status":200,
  "name":"elasticsearch-abcd",
  "cluster_name":"elasticsearch-cluster",
  "version":{
    "number":"1.4.0",
    "build_hash":"bc94bd81298f81c656893ab1ddddd30a99356066",
    "build_timestamp":"2014-11-05T14:26:12Z",
    "build_snapshot":false,
    "lucene_version":"4.10.2"
  },
  "tagline":"You Know, for Search"
}
```

On the off chance that you decided to introduce Marvel & set up, you can directly get to Marvel interface utilizing same SSH tunnel and entering the below given URL in browser:

http://localhost:9200/_plugin/marvel/

Troubleshooting

A deployment might fail to complete successfully for one of the following reasons:

1. Deployment Manager Inability to make a cloud resource, for example an instance, burden adjusting rule or disk.
2. Deployment organizer module failure to make a persistent disk (for those deployments that require and incorporate information on disks)
3. Software installation or configuration failure

The accompanying will help you recognize and address each of these issues if you experience them.

Deployment Manager Failure

On the off chance that a sending neglects to finish effectively, every module status is shown in the Developers Console, for example:

```
module: DEPLOYMENT_FAILED  
Replica module-1234 failed with status PERMANENTLY_FAILING: Replica State  
changed to PERMANENTLY_FAILING. Unexpected error: detailed message
```

On the off chance that the *detailed message* demonstrates a CPU quota sharing issue or other issues that can be amended, erase the deployment, rectify the issue, and attempt once more.

The quota for your cloud project can be checked at **Compute Engine Quotas**.

If the error message on the Developers Console does not give proper detail regarding the failure, for example:

```
module: DEPLOYMENT_FAILED  
Replica module-1234 failed with status PERMANENTLY_FAILING: Replica State  
changed to PERMANENTLY_FAILING. Replica was unhealthy 2 sequential times.
```

This message implies that a blunder happened on one or more Compute Engine instances that are an essential piece of the deployment.

Deployment coordinator or software installation failure

In the event that the module demonstrating failure is the Deployment facilitator module, then the blunder occurred is likely because of data disk creation. In the event that another module is showcasing failure, then the mistake is because of incorrect software setup or settings.

The detailed failure logs can be found on the virtual machine instances that were launched. On each instance, there will be a directory:

```
/gagent/metaOutput
```

The directory contains an arrangement of yield documents:

```
stdoutN.txt  
stderrN.txt  
exitcodeN.txt
```

To get a complete rundown of your Compute Engine instances, check the Compute Engine list of virtual machine instances in the **Developers Console** or run the given below command:

```
$ gcloud compute instances list
```

SSH to any specified/listed instance:

```
$ gcloud compute ssh --zone=<zone> <instance>
```

Deployment processing

The Google Cloud Deployment Manager deals with establishment & installation of Elasticsearch cluster. The Deployment Manager uses formats and related shell scripts to determine guidelines that are particular to the deployment of Elasticsearch.

Elasticsearch module

Amid the formation of the Elasticsearch cluster, Deployment Manager finishes these steps:

1. Downloads and installs required packages/software such as OpenJDK 1.7.
2. Downloads and installs Elasticsearch.
3. Downloads the specified software source code to /usr/src.
4. Creates a data disk and mounts it on /elasticsearch.
5. Configures every node of Elasticsearch with the cluster name, and the IP addresses of its associate connected instances.
6. Configures regular parameters to enhance Elasticsearch performance.
7. Downloads, installs and designs Marvel (if required).

Configure your virtual machine instances

Associate with your Elasticsearch instances and perform any framework setting that is required on every server.

- Use the SSH button on the deployment page or the SSH button in the console.
- You can likewise utilize `gcloud compute ssh <instance-name> --zone <zone> --project <project_id>` to connect to your instances. You can gaze upward your instances by using the `gcloud compute instances list --zone <zone> --project <project_id>` command.

4.6 Developing Front End using Angular JS

Coding:

```
<!doctype html>
<!--[if gt IE 8]><!--><html lang="en-US"><!--<![endif]-->
<head>

<!-- META TAGS -->

<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Find a Recipe</title>
<link rel="shortcut icon" href="images/rf2.jpg" />

<!-- Style Sheet-->

<link rel="stylesheet" href="style.css"/>
<link rel='stylesheet' id='bootstrap-css-css' href='css/bootstrap5152.css?ver=1.0' type='text/css'
media='all' />
<link rel='stylesheet' id='responsive-css-css' href='css/responsive5152.css?ver=1.0'
type='text/css' media='all' />
<link rel='stylesheet' id='pretty-photo-css-css'
href='js/prettyphoto/prettyPhotoaeb9.css?ver=3.1.4' type='text/css' media='all' />
<link rel='stylesheet' id='main-css-css' href='css/main5152.css?ver=1.0' type='text/css'
media='all' />
<link rel='stylesheet' id='custom-css-css' href='css/custom5152.html?ver=1.0' type='text/css'
media='all' />
<link rel="import" href="blue-skin.html">

</head>
<body>

<!-- Start of Header -->

<div class="header-wrapper">
<header>
<div class="container">
<div class="logo-container">

<!-- Website Logo -->

<a href="index.html" title="Recipe search">
</a>
<span class="tag-line"></span>
```



```

<div class='no-recipes' ng-hide='recipes.length'>No results</div>

<article class='recipe' ng-cloak ng-repeat='recipe in recipes'>

<h2>
<a ng-href='{{ recipe.url }}'>{{ recipe.name }}</a>
</h2>

<ul>

<li ng-repeat='ingredient in recipe.ingredients'>{{ ingredient }}</li>

</ul>

<p>
{{ recipe.description }}
<a ng-href='{{ recipe.url }}'>... more at {{ recipe.source }}</a>
</p>

</article>

<div class='load-more' ng-cloak ng-hide='allResults'>
  <a ng-click='loadMore()'>More...</a>
</div>
</section>
</div>
</div>

<script src="bower_components/angular/angular.js" type="text/javascript"></script>

<script src="bower_components/elasticsearch/elasticsearch.angular.js"
type="text/javascript"></script>

<script src="js/script.js" type="text/javascript"></script>

<!-- End of Search Wrapper -->
<!-- Start of Page Container -->
<!-- start of page content -->
<!-- Basic Home Page Template -->

<section class="span4 articles-list">
</div>
</div>

<!-- end of page content -->

```

```
<!-- start of sidebar -->
```

```
<aside class="span4 page-sidebar">  
</aside>
```

```
<!-- end of sidebar -->
```

```
</div>  
</div>  
</div>
```

```
<!-- End of Page Container -->
```

```
<!-- Start of Footer -->
```

```
<footer id="footer-wrapper">  
<div id="footer" class="container">  
<div class="row">  
<div class="span3">  
<section class="widget">  
<h3 class="title">How it works</h3>  
<div class="textwidget">  
<p>The search site is a specific recipe search which takes care of your taste buds,so if your tastes buds are tickling and you don't want to spend money buying it in a restaurant. Step in to the site we are here to help you calm your taste buds. </p>
```

```
</div>  
</section>  
</div>
```

```
<div class="span3">  
<section class="widget">  
<h3 class="title">Flickr Photos</h3>  
<div class="flickr-photos" id="basicuse">  
</div>  
</section>  
</div>
```

```
</div>  
</div>
```

```
<!-- end of #footer -->
```

```
<!-- Footer Bottom -->
```

```
<div id="footer-bottom-wrapper">
```

```
<div id="footer-bottom" class="container">
<div class="row">
<div class="span6">
<p class="copyright">
Copyright © 2015. Powered by UPES.
</p>
</div>
<div class="span6">
```

<!-- Social Navigation -->

```
<ul class="social-nav clearfix">
<li class="linkedin"><a target="_blank" href="#"></a></li>
<li class="stumble"><a target="_blank" href="#"></a></li>
<li class="google"><a target="_blank" href="#"></a></li>
<li class="deviantart"><a target="_blank" href="#"></a></li>
<li class="flickr"><a target="_blank" href="#"></a></li>
<li class="skype"><a target="_blank" href="skype:#?call"></a></li>
<li class="rss"><a target="_blank" href="#"></a></li>
<li class="twitter"><a target="_blank" href="#"></a></li>
<li class="facebook"><a target="_blank" href="#"></a></li>
</ul>
</div>
</div>
</div>
</div>
```

<!-- End of Footer Bottom -->

```
</footer>
```

<!-- End of Footer -->

```
<a href="#top" id="scroll-top"></a>
```

```
<!-- script -->
<script type='text/javascript' src='js/jquery-1.8.3.min.js'></script>
<script type='text/javascript' src='js/jquery.easing.1.3.js'></script>
<script type='text/javascript' src='js/prettyphoto/jquery.prettyPhoto.js'></script>
<script type='text/javascript' src='js/jflickrfeed.js'></script>
<script type='text/javascript' src='js/jquery.liveSearch.js'></script>
<script type='text/javascript' src='js/jquery.form.js'></script>
<script type='text/javascript' src='js/jquery.validate.min.js'></script>
<script type='text/javascript' src='js/custom.js'></script>
```

```
</body>
</html>
```

4.7 Connecting Front End (AngularJS) with Elastic Search

Coding:

```
var fs = require('fs');
var es = require('elasticsearch');
var client = new es.Client({
  host: 'localhost:9200'
});

fs.readFile('recipeitems-latest.json', {encoding: 'utf-8'}, function(err, data) {
  if (err) { throw err; }

  // Build up a giant bulk request for elasticsearch.

  bulk_request = data.split('\n').reduce(function(bulk_request, line) {
    var obj, recipe;

    try {
      obj = JSON.parse(line);
    } catch(e) {
      console.log('Done reading');
      return bulk_request;
    }

    // Rework the data

    recipe = {
      id: obj._id.$oid, // Used for mongodb entry
      name: obj.name,
      source: obj.source,
      url: obj.url,
      recipeYield: obj.recipeYield,
      ingredients: obj.ingredients.split('\n'),
      prepTime: obj.prepTime,
      cookTime: obj.cookTime,
      datePublished: obj.datePublished,
      description: obj.description
    };

    bulk_request.push({index: {_index: 'recipes', _type: 'recipe', _id: recipe.id}});
    bulk_request.push(recipe);
    return bulk_request;
  }, []);
```

```
// Simulate synchronous insert
```

```
var busy = false;
var callback = function(err, resp) {
  if (err) { console.log(err); }

  busy = false;
};
```

```
// Recursively whittle away at bulk_request, 1000 at a time.
```

```
var perhaps_insert = function(){
  if (!busy) {
    busy = true;
    client.bulk({
      body: bulk_request.slice(0, 1000)
    }, callback);
    bulk_request = bulk_request.slice(1000);
    console.log(bulk_request.length);
  }

  if (bulk_request.length > 0) {
    setTimeout(perhaps_insert, 10);
  } else {
    console.log('Inserted all records.');
```

```
  }
};

perhaps_insert();
});
```

5. Diagrams

5.1 Working of Elastic Search

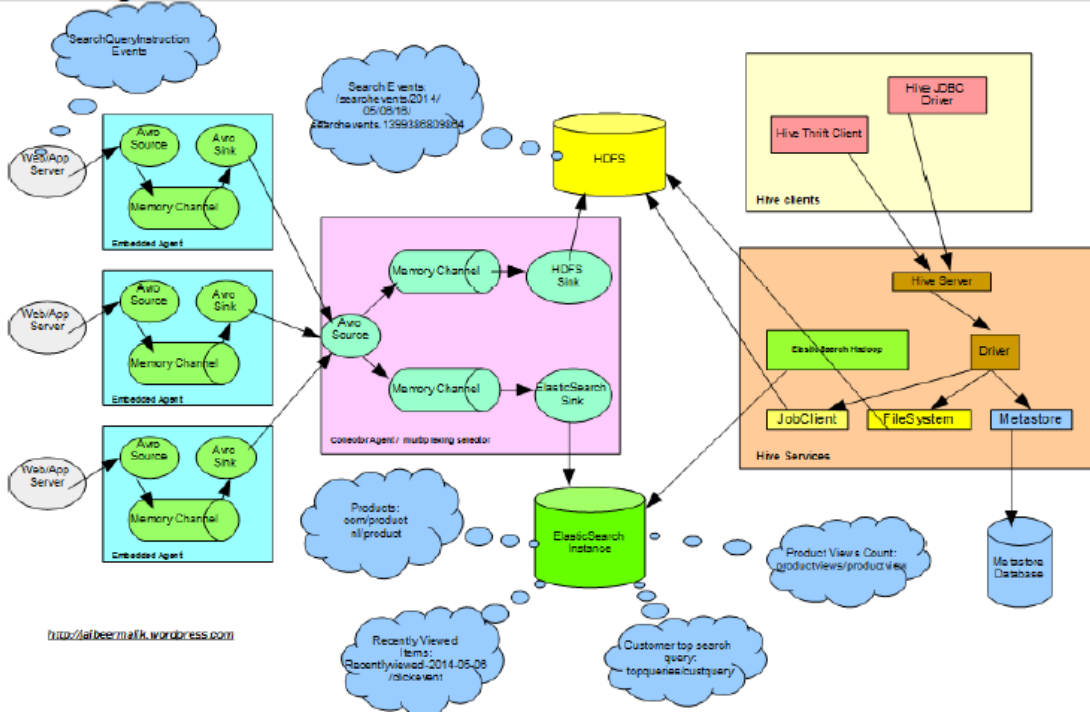


Fig 5.1 Working of ElasticSearch

5.2 DFD

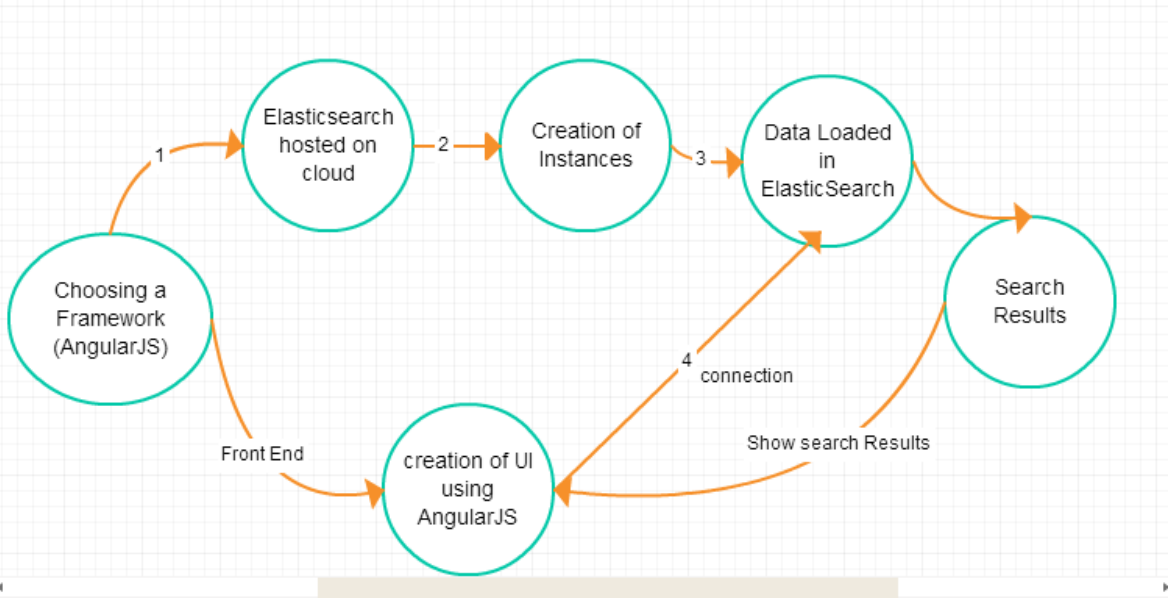


Fig 5.2 DFD (Data Flow Diagram)

5.3 Class Diagram

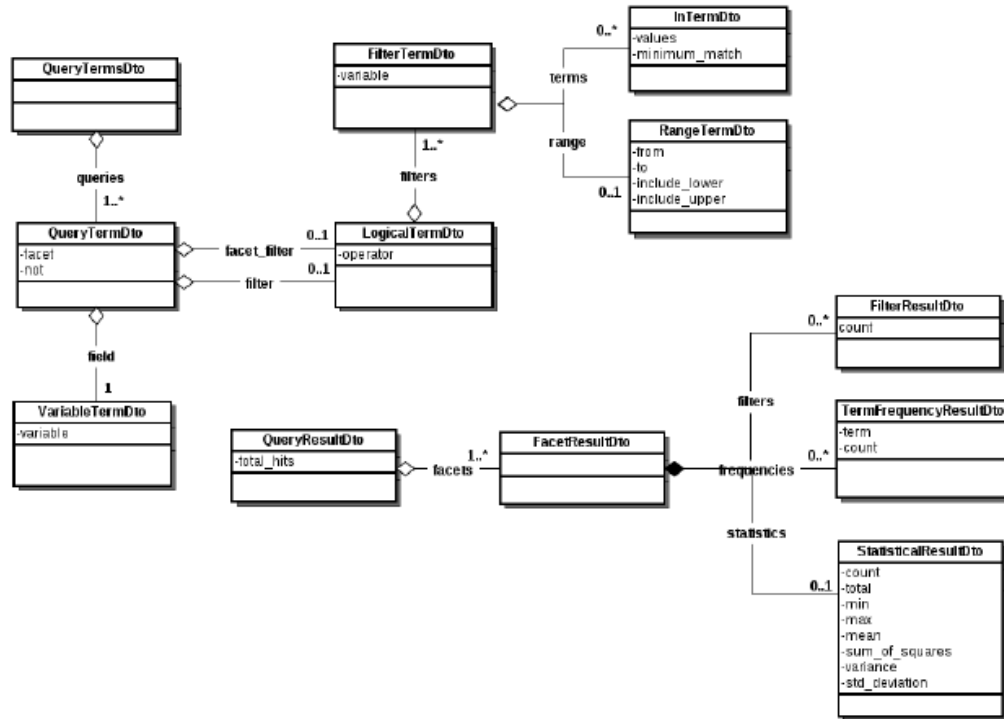


Fig 5.3 Class Diagram

5.4 Activity Diagram

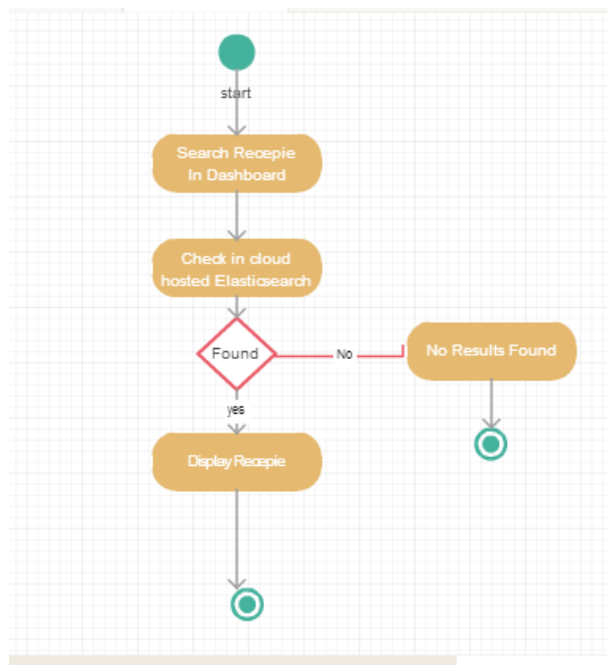
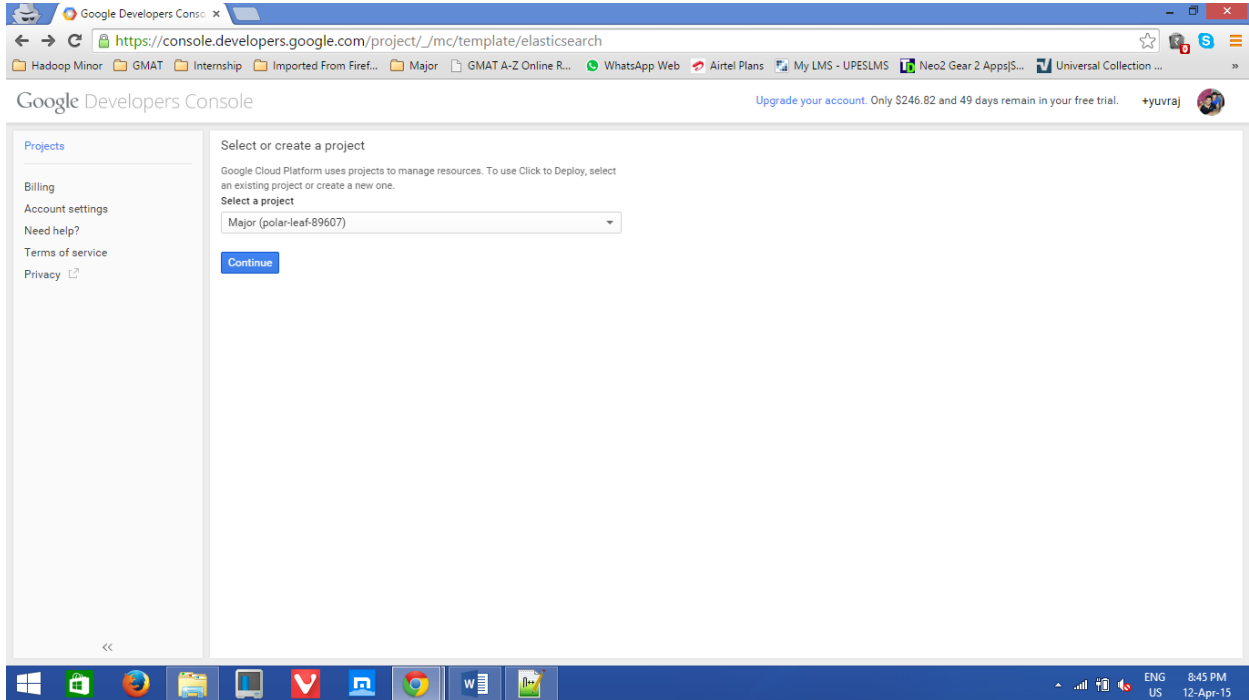


Fig 5.4 Activity Diagram

6. Snapshots

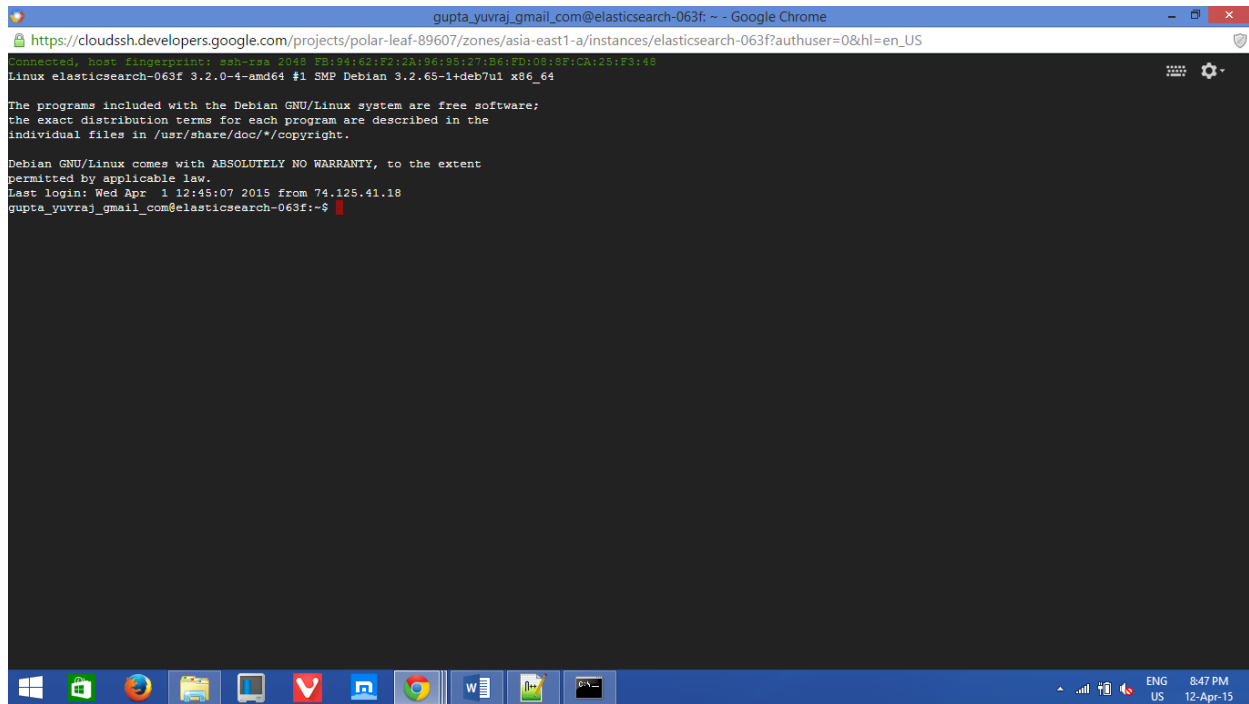
Snapshot of Google Cloud Compute Engine:



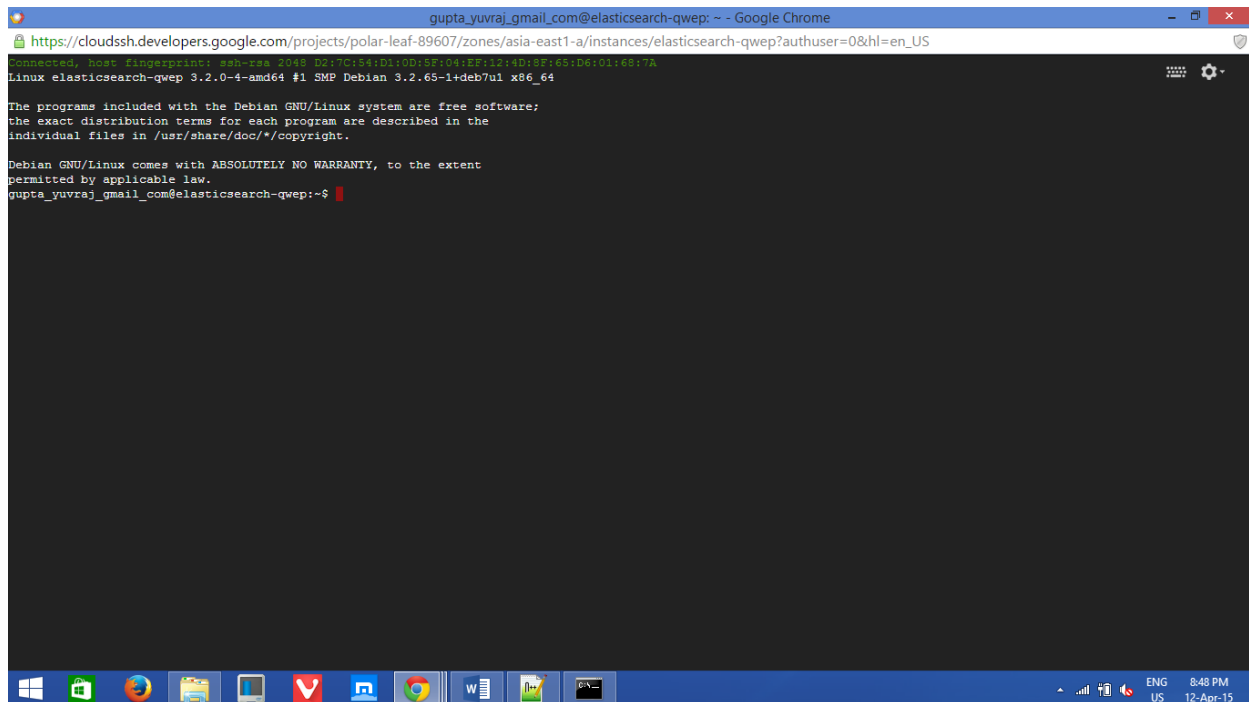
After Clicking Continue



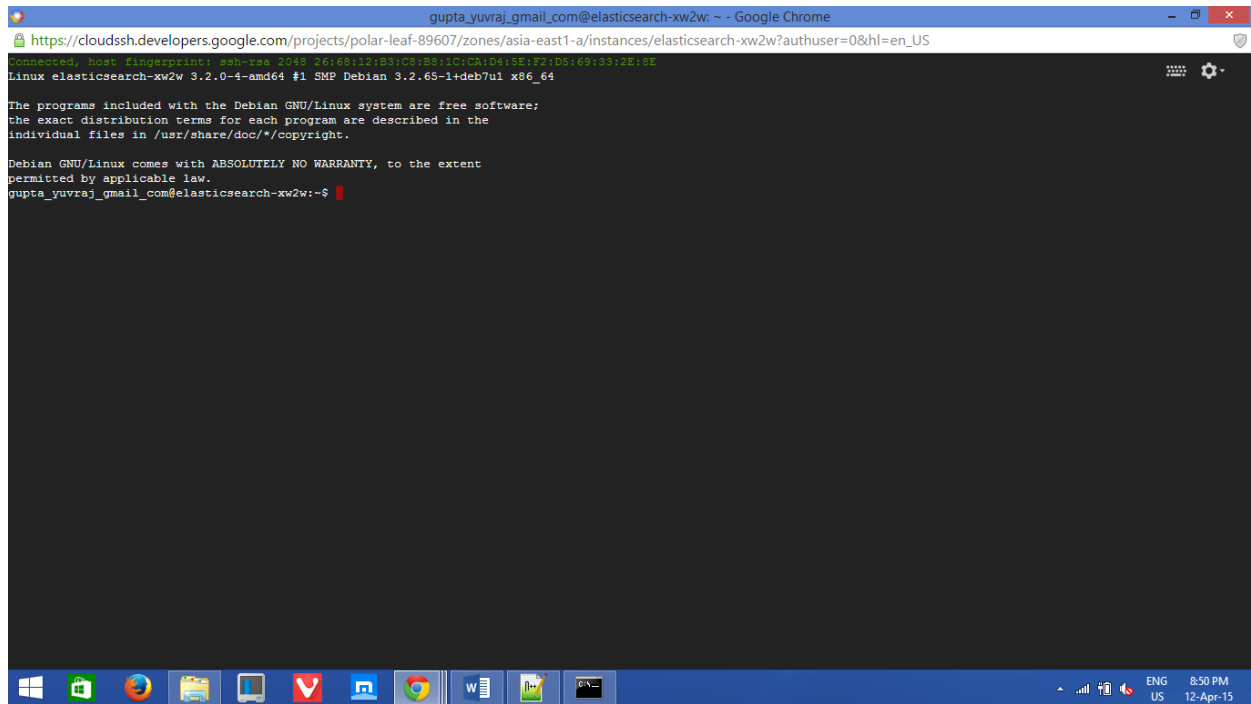
After Connecting Cluster Node of elasticsearch-063f via SSH in browser window



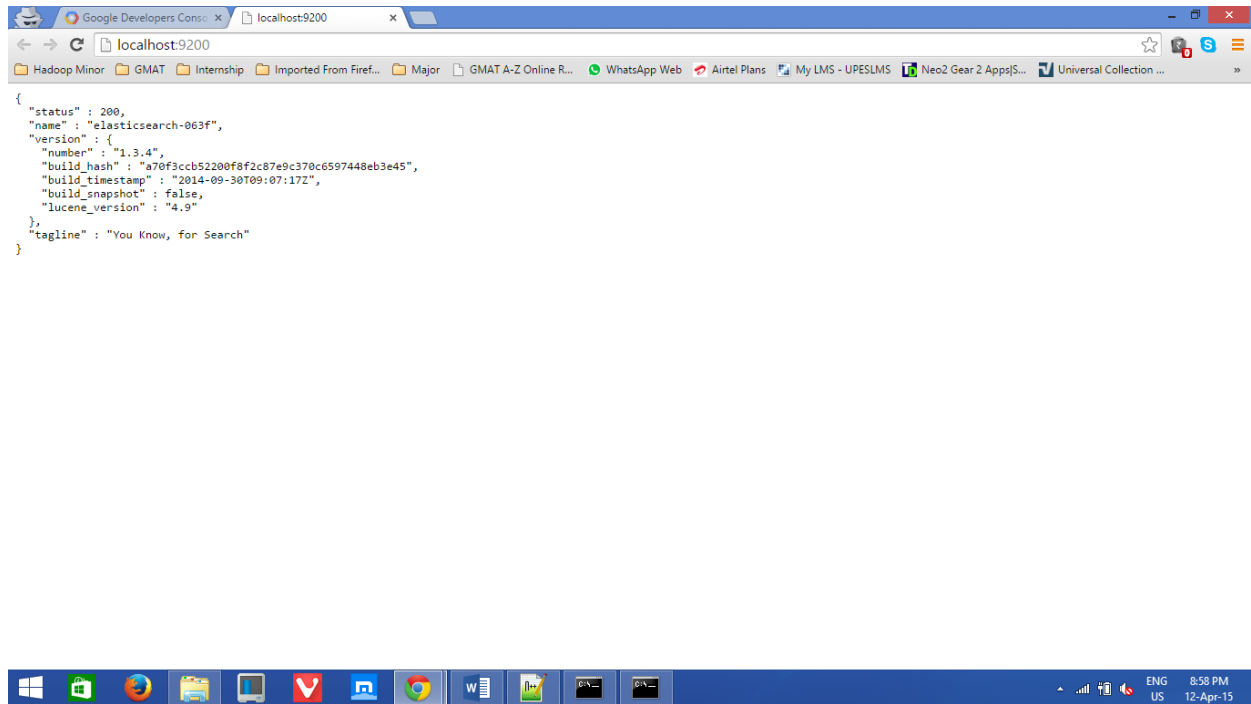
After Connecting Cluster Node of elasticsearch-qwep via SSH in browser window



After Connecting Cluster Node of elasticsearch-xw2w via SSH in browser window



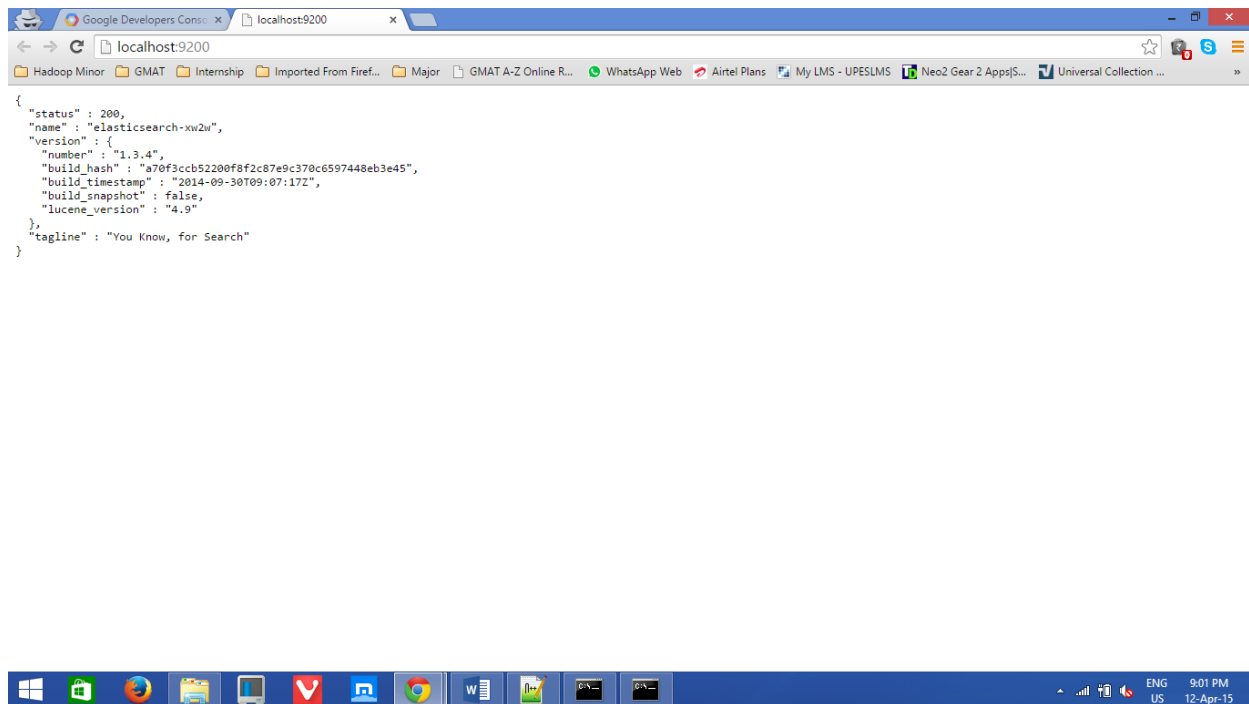
Access ElasticSearch Instance elasticsearch-063f over HTTP using tunneling



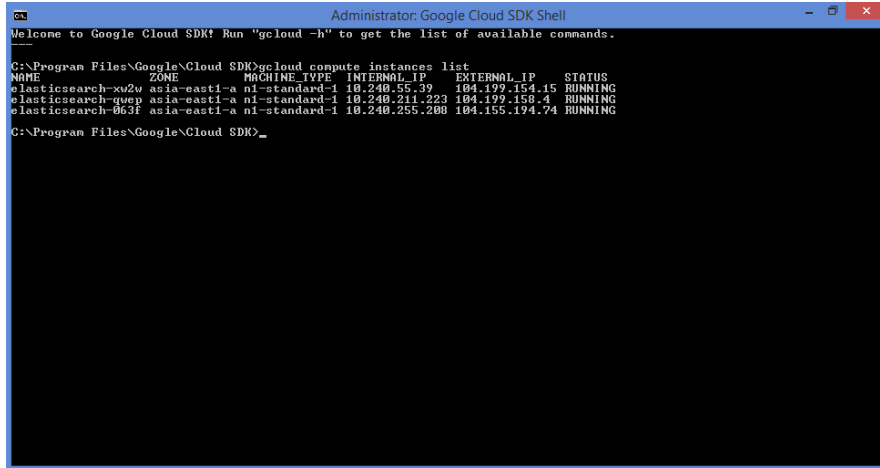
Access ElasticSearch Instance elasticsearch-qwep over HTTP using tunneling



Access ElasticSearch Instance elasticsearch-xw2w over HTTP using tunneling

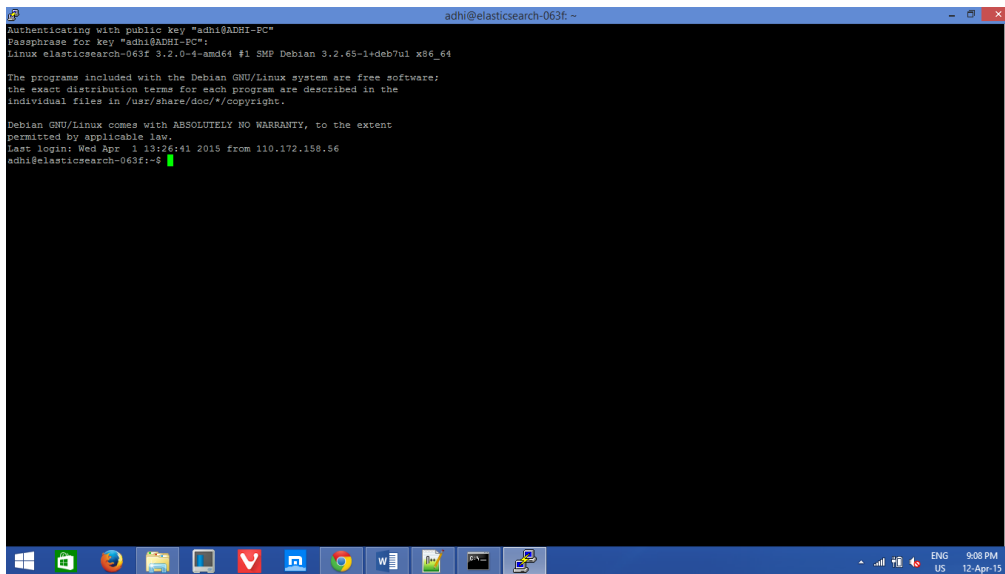


Listing of instances using gCloud command



```
Administrator: Google Cloud SDK Shell
Welcome to Google Cloud SDK! Run "gcloud -h" to get the list of available commands.
---
C:\Program Files\Google\Cloud SDK>gcloud compute instances list
NAME                                ZONE          MACHINE_TYPE  INTERNAL_IP    EXTERNAL_IP    STATUS
elasticsearch-xw2w                  asia-east1-a  n1-standard-1 10.240.45.39   104.199.154.15  RUNNING
elasticsearch-qwep                  asia-east1-a  n1-standard-1 10.240.211.223 104.199.150.4   RUNNING
elasticsearch-063f                  asia-east1-a  n1-standard-1 10.240.255.208 104.155.194.74  RUNNING
C:\Program Files\Google\Cloud SDK>_
```

Connecting Cluster elasticsearch-063f via gCloud command

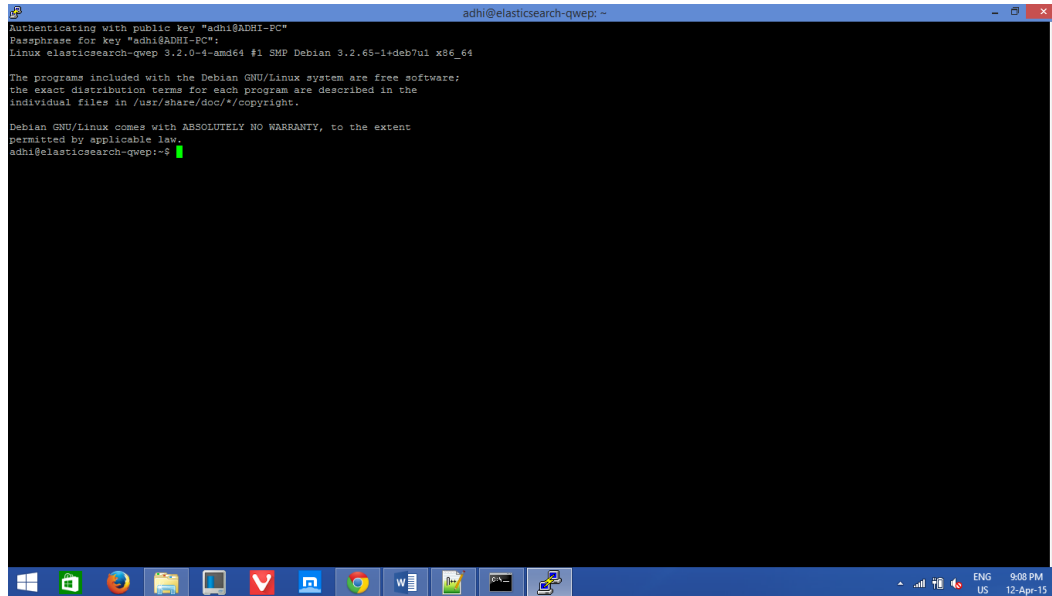


```
adhi@elasticsearch-063f: ~
Authenticating with public key "adhi@ADHI-PC"
Passphrase for key "adhi@ADHI-PC":
Linux elasticsearch-063f 3.2.0-4-amd64 #1 SMP Debian 3.2.65-1+deb7ui x86_64

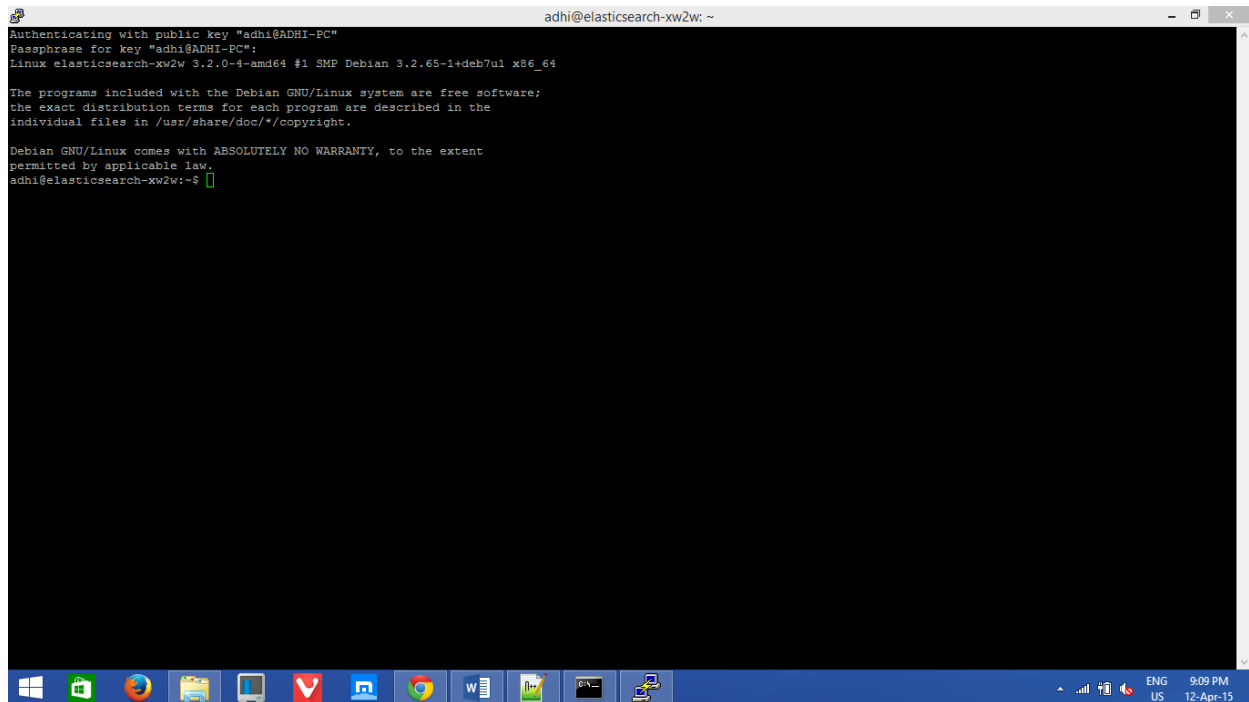
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Apr 1 13:26:41 2015 from 110.172.158.56
adhi@elasticsearch-063f:~$
```

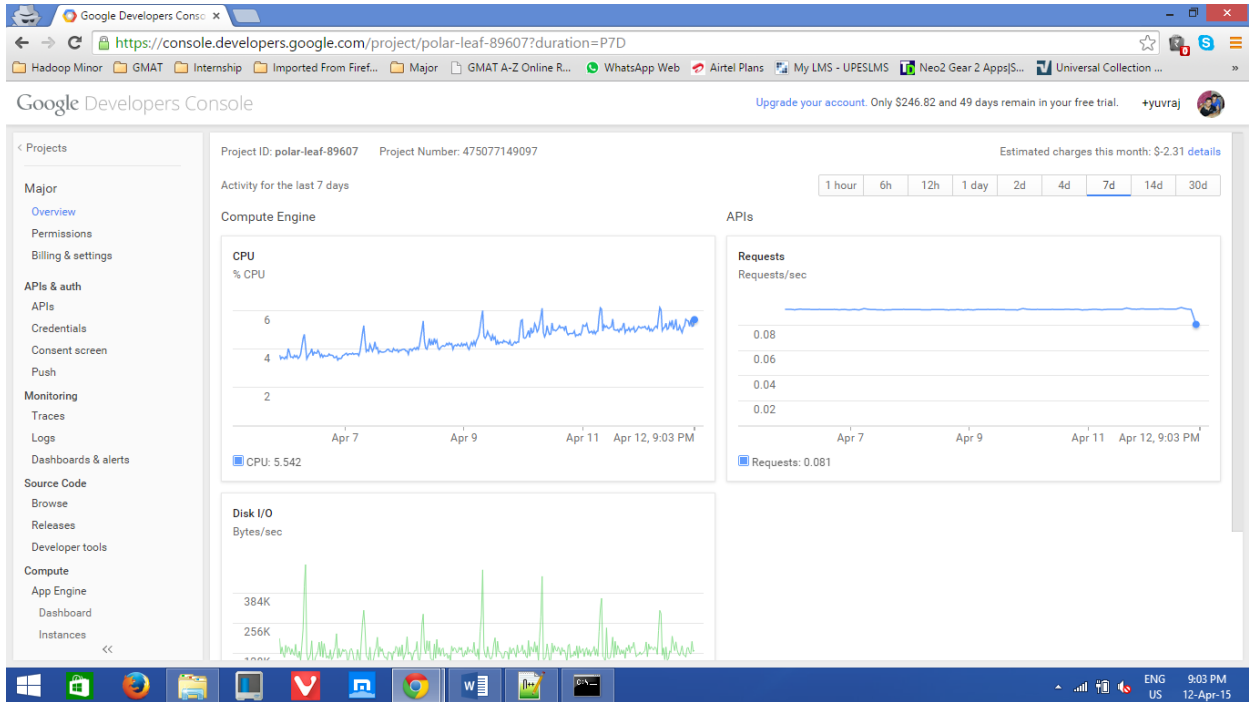
Connecting Cluster elasticsearch-qwep via gCloud command



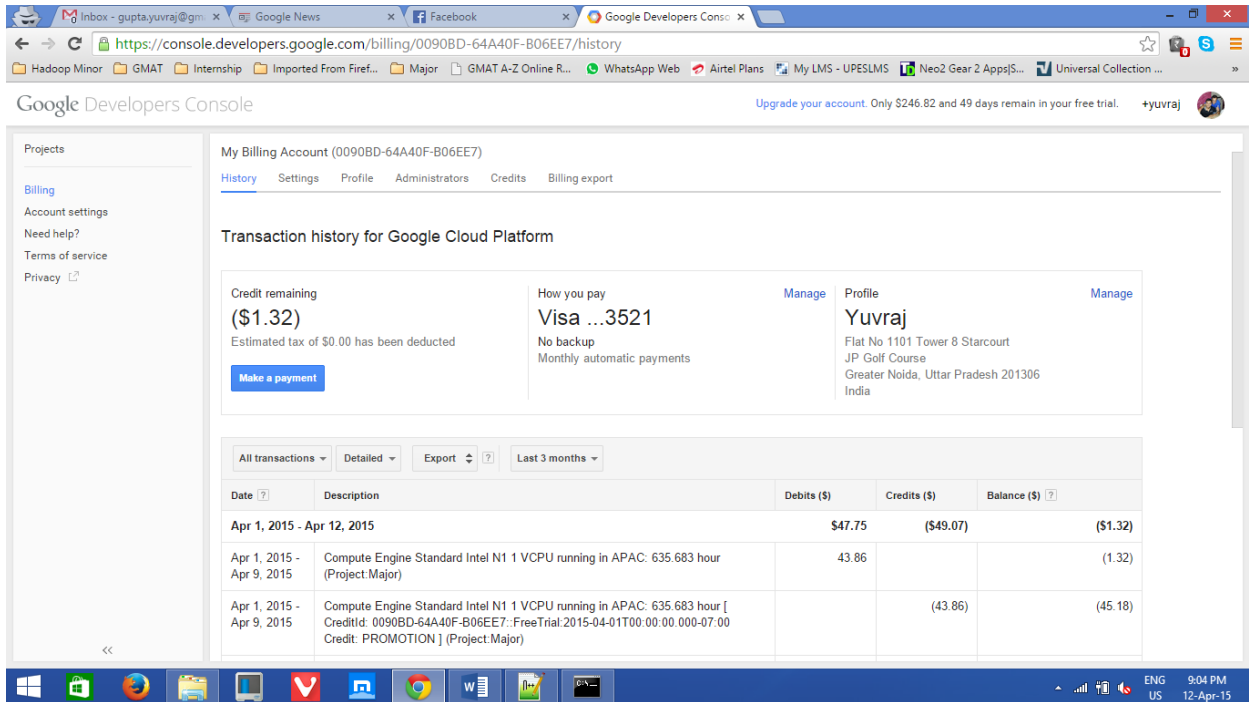
Connecting Cluster elasticsearch-xw2w via gCloud command



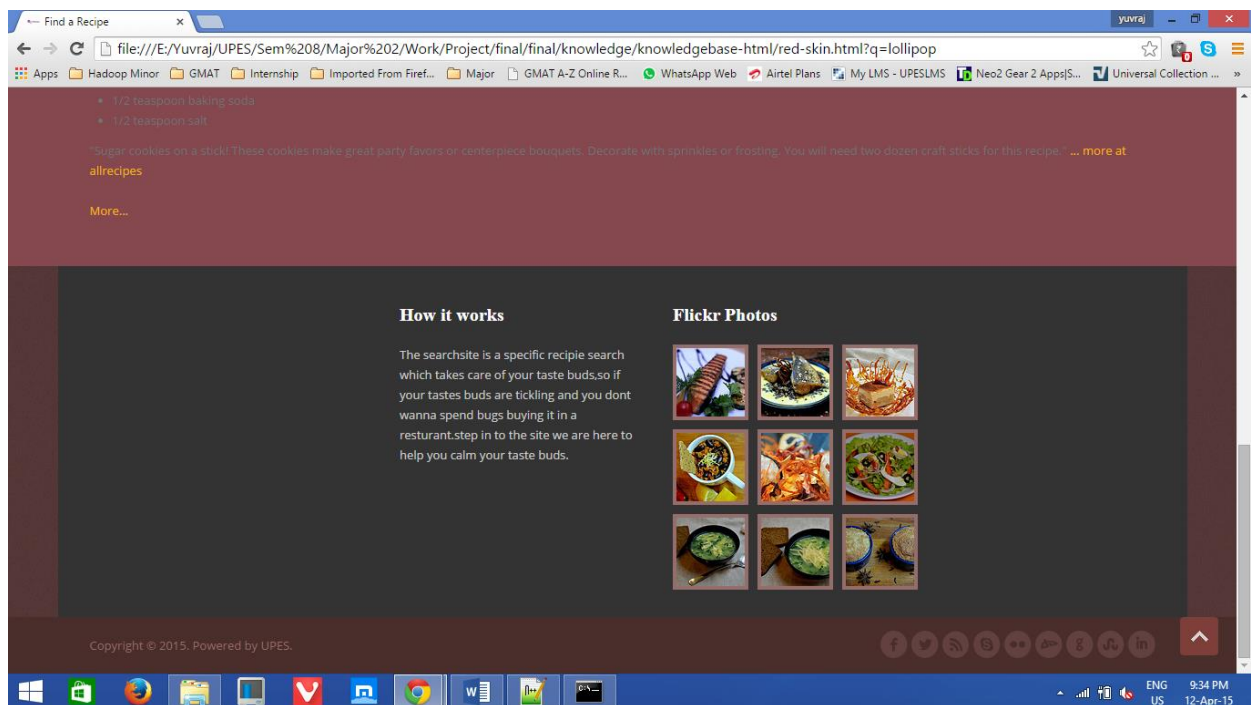
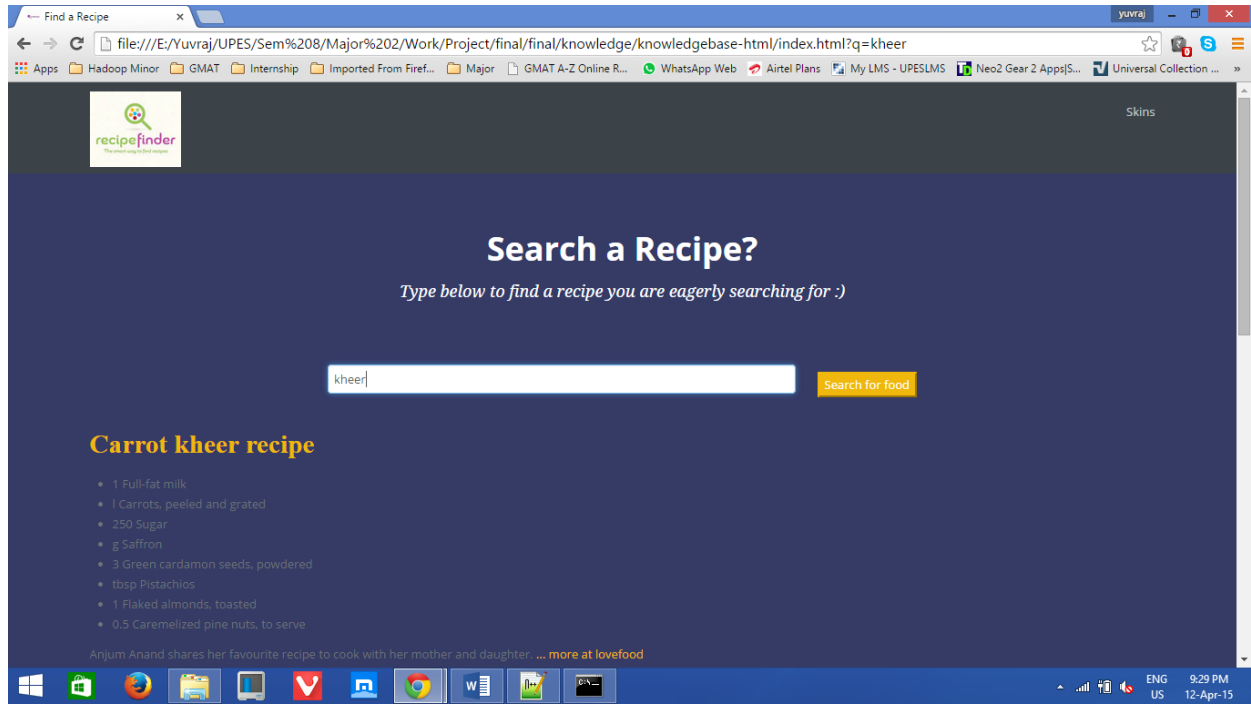
Google Compute Engine Developer Console Overview



Google Compute Engine Developer Console Billing



Snapshot of The Front-End



Find a Recipe x yuvraj

file:///E:/Yuvraj/UPES/Sem%208/Major%202/Work/Project/final/final/knowledge/knowledgebase-html/blue-skin.html?q=palak%20paneer

Apps Hadoop Minor GMAT Internship Imported From Firef... Major GMAT A-Z Online R... WhatsApp Web Airtel Plans My LMS - UPESLMS Neo2 Gear 2 AppzS... Universal Collection ...

recipefinder

Skins

Search a Recipe?

Type below to find a recipe you are eagerly searching for :)

palak paneer Search for food

Palak Paneer (Spinach Curry)

- 2 tablespoons vegetable oil, divided
- 2 cups chopped fresh spinach
- 1 teaspoon ground cumin
- 3/4 cup chopped onion
- 2 green chile peppers, chopped
- 2 teaspoons chopped garlic
- 2 tomatoes, chopped
- 1/2 cup water
- 2 teaspoons ground coriander
- 1 to 2 non-spicy mild chutneys

Windows Taskbar: File Explorer, Internet Explorer, VLC, Word, PowerPoint, OneDrive, System tray: Network, Volume, Power, ENG US, 9:34 PM 12-Apr-15

Find a Recipe x yuvraj

file:///E:/Yuvraj/UPES/Sem%208/Major%202/Work/Project/final/final/knowledge/knowledgebase-html/blue-skin.html?q=palak%20paneer

Apps Hadoop Minor GMAT Internship Imported From Firef... Major GMAT A-Z Online R... WhatsApp Web Airtel Plans My LMS - UPESLMS Neo2 Gear 2 AppzS... Universal Collection ...

Palak Paneer (Indian Spinach and Paneer)

- cooking spray
- 1/2 pound paneer, cubed
- 2 tablespoons canola oil, divided
- 1 teaspoon cumin seeds
- 1 1/2 cups water
- 1 (10 ounce) package fresh spinach
- 1 large tomato, diced
- 3 cloves garlic
- 2 dried red chile peppers, or to taste
- 1 tablespoon chopped fresh ginger root
- 1 teaspoon curry powder
- 1 teaspoon ground coriander
- 1 teaspoon ground turmeric
- 1 pinch salt to taste

"This is my mom's recipe. It's easy to make and tasty to enjoy." ... [more at allrecipes](#)

Palak Paneer: A Must-Try Indian Recipe!

Palak Paneer is one of my all time favorite Indian foods, not only because it tastes out of the world but also because it's one of those dishes that is rich but you still you don't feel guilty eating it. Palak, which is the Hindi name for spinach, doesn't need any introduction. Paneer is fresh Indian cheese that is rich in protein. ... [more at tastykitchen](#)

More...

Windows Taskbar: File Explorer, Internet Explorer, VLC, Word, PowerPoint, OneDrive, System tray: Network, Volume, Power, ENG US, 9:35 PM 12-Apr-15

7. Conclusion

In the wake of doing an eight week old project when we at long closed it ,the undertaking of project gave us a definite thought of the Angular JS framework which we picked over alternate frameworks like the HTML5,JQuery,Backbone,Knockout etc. We chose AngularJS due to the properties provided such as:

- Reusable Components
- Lightweight
- Extensibility & Maintainability
- Like jQuery, it doesn't act as a burden
- Begin in couple of minutes
- Large & rapidly developing community
- Modern (REST, SPAs)
- Open Source

At first we had no clue whether to take the project into the cloud space or not yet at the end of the project we moved the project over to the cloud providing us valuable knowledge to the latest cloud hosting, technology which was an addition plus for students specializing in Cloud domain. The following in addition to of the project was to implement distributed computing which is the foundation of any tremendous new framework.

All on account of the PaaS Google Compute Engine for providing us with the instances so we can replicate data using the replication factor of three on three instances, hence providing us same data at three different geographical areas, if there should be an occurrence of the essential instance breakdown we had two auxiliary instances for the unwavering quality purposes hence furnishing with zero downtime.

Amid the project we even learnt about most recent open source product ie ElasticSearch which stands out amongst the most productive in search tool which furnished our project with productivity for fathoming the mind boggling and massive 2 lakh records which we picked as JSON records for the project. We even learnt the framework how to function with model, view and control which is the heart of any application development. The last however not the minimum in the thing which in the project we learnt was to associate the remote host to the local host. All the execution which was done at the cloud was tunneled at the local host and the outcomes were showcased at the local host.

8. Definitions

UML: Unified Modelling Language (UML) is an institutionalized, broadly useful modelling language in the field of software engineering.

Use Case Diagram: A use case diagram at its least complexity is a representation of a user's interaction with the system and depict the determinations of a use case.

Class Diagram: In software engineering, a class diagram in the Unified Modeling Language (UML) is a kind of static structure diagram that depicts the structure of a system by demonstrating the system's classes, their properties, operations (or methods), and the connection among objects.

Collaboration Diagram: The UML Collaboration diagram is utilized to model how objects included in a situation communicate, with every other object instantiating a specific class in the system. Objects are joined by links, each link representing an instance of a relationship between the separate classes included. The connection shows messages sent between the objects, and the kind of message passed (synchronous, asynchronous, straight forward, balking, and timeout).

Data Flow diagram: A data flow diagram (DFD) is a graphical representation of the "flow" of information through a data system, displaying its process perspectives. Regularly they are a preliminary step used to make a diagram of the system which can later be elaborated. DFDs can likewise be utilized for the visualization of data processing (structured design).

Constraint: A constraint is something that fills the role of a physical, social or financial restriction. It is a derived form of the intransitive verb form constrained.

9. Bibliography

- Banon, Shay. "The Future of Compass & ElasticSearch". Archived from the original on 2013-08-27.
- Banon, Shay (2010-02-08). "You Know, for Search". Archived from the original on 2013-01-16.
- <http://techcrunch.com/2014/06/05/elasticsearch-scores-70m-in-series-c-to-fund-growth-spurt/>"percolate at elasticsearch.org reference". Elasticsearch.org. Retrieved 2014-02-04.
- "elasticsearch Guide: Gateway". elasticsearch. Retrieved April 19, 2013.
- "ElasticSearch as database". Karussell.wordpress.com. Retrieved 2014-02-04.
- "No transaction support". Elasticsearch-users.115913.n3.nabble.com. 2010-07-08. Retrieved 2014-02-04.
- PetarDjekic. "Architecture behind our new Search and Explore experience". Backstage.soundcloud.com. Retrieved 2014-02-04.
- "A Whole New Code Search 路 GitHub". Github.com. 2013-01-23. Retrieved 2014-02-21.
- What Is Angular? .Web. 12 February 2013.
- AngularJS: Developer Guide: Scopes. Web. 3 January 2014.
- 5 Awesome AngularJS Features. Web. 13 February 2013.
- angular/angularjs-batarang (GitHub). Web. 12 October 2014.
- Ford, Brian. Introducing the AngularJS Batarang. *AngularJS Blog*. Web. 10 December 2014
- <https://cloud.google.com/solutions/elasticsearch/click-to-deploy>
- <https://cloud.google.com/solutions/elasticsearch/>
- <https://www.elastic.co/about/partners/google-compute-engine>