**Cost Resources Analysis**

For the development of the project, we will need:

* Operation System
* **Windows operating systems** developed by Microsoft. Each version of **Windows** includes a graphical user interface, with a desktop that allows users to view files and folders in **windows**.
* **Mac** OS is an **operating system** that was designed for the Apple **Macintosh** computer. It was developed by Apple Inc.
* **Apache HHTP Server**: is an open-source HTTP server for modern operation system including UNIX, Windows OS and Mac OS. It provides a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards.
* **MySQL:**  is an open source relational database management system. Information in a **MySQL** database is stored in the form of related tables. **MySQL** databases are typically used for web application development (often accessed using PHP).

**Host: hosting** is the business of housing, serving, and maintaining files for one or more websites. In a sense, you rent space on a computer to hold your website.

**000Webhost **

Free Web Hosting with PHP, MySQL, free Website Builder, cPanel and no ads. Almost unlimited free website hosting and free domain hosting.

**GoDaddy **

**GoDaddy** Inc. is a publicly traded Internet domain registrar and web **hosting** company.

**Development tools (softwares and services)**

* **Sublime Text:** is a propriety cross-platform source code editor with Python application programming interface (API). It natively supports many programming languages (such as Java, Javascript, Json, Htlm, CSS) and markup languages. Can be extended by users with plugins (such as JSHint), typically community-built and maintained under free-software licences.
* **HeidiSQL:** previously known as MySQL-Front, is a free and open source client, or front-end for MYSQL as well as Microsoft SQL Server and PostgreSQL.
* **Bootstrap Themes & Templates:** is an HTML5 & CSS3 framework designed to help you kickstart the development of webapps and sites, providing the layout of the page with an appropriate graphic of presentation to the client.
* **JQuery**: is a concise and fast JavaScript library that can be used to simplify event handling, HTML document traversing, Ajax interactions and animation for speedy website development. jQuery simplifies the HTML's client-side scripting, thus simplifying Web 2.0 applications development.
* **Google Fonts:** as a library of more than 600 font families from Google that are avaible to Web designers.
* **BACKUP SOLUTION?**

**Technologies**

**Google Maps API**  
Google Maps is a desktop web mapping service developed by Google. I offers satellite imagery, street maps, 360° panoramic, views of street (Street View), real-time traffic conditions (Google Traffic and route planning for travelling by foot, car, bicycle, or public transportation.

On this application we will be using the geolocation, search (destination) and marker's functions (Restaurants).  
  
 Google Maps ***Geolocation***  
 The Google Maps Geolocation API returns a location and accuracy radius based on information about cell towers and WIFI nodes that the mobile client can detect. This document describes the protocol used to send this data to the server and to return a response to the client.  
 Communication is done over HTTPS using POST. Both request and response are formatted as JSON, and the content type of both is application/JSON.  
  
Throughout the development this function was used as a base, which was already offered by Google.  
following:

if (navigator.geolocation) {  
          navigator.geolocation.getCurrentPosition(function(position) {  
            var pos = {  
              lat: position.coords.latitude,  
              lng: position.coords.longitude  
            };  
  
            infoWindow.setPosition(pos);  
            infoWindow.setContent('Location found.');  
            map.setCenter(pos);  
          }, function() {  
            handleLocationError(true, infoWindow, map.getCenter());  
          });  
        } else {  
          // Browser doesn't support Geolocation  
          handleLocationError(false, infoWindow, map.getCenter());  
        }  
      }  
  
      function handleLocationError(browserHasGeolocation, infoWindow, pos) {  
        infoWindow.setPosition(pos);  
        infoWindow.setContent(browserHasGeolocation ?  
                              'Error: The Geolocation service failed.' :  
                              'Error: Your browser doesn\'t support geolocation.');  
      }

Based on this, it was necessary to adjust our system with our latitudes and longitudes according to main location (Ireland). We will be detailing further along the project.

**lat/lon restaurant breakdown**

In this function we are registering the "Restaurant" and then saving in the Database with their respective identities, which are they:

***username*** - > nickname and also will work as a "primary key"  
***password*** -- > access password  
***name*** -- > name of the Restaurant  
***address*** --> Informing the address of the Restaurant and then it will be converted to latitude and longitude  
*https://maps.google.com/maps/api/geocode/json?address=%27.$prepAddr.%27&sensor=false%27);*

{

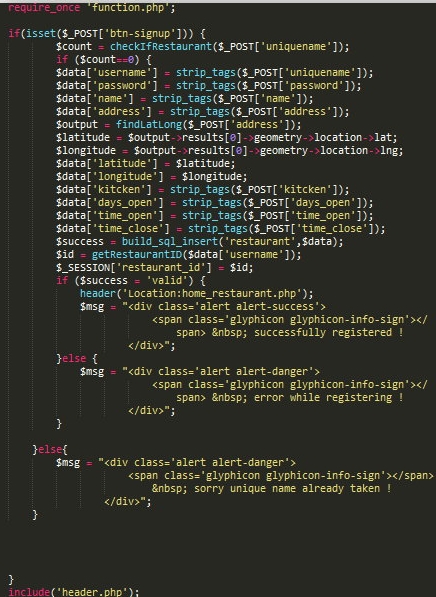
"results" : [],

"status" : "ZERO\_RESULTS"

}



***Kitcthen*** --> Type of cuisine ***Day's Open*** --> How many days are open  
***Time Open*** --> Opening hours  
***Time Close*** --> Opening hours

****

When already saved in the database with their respective identities this function "selectRestaurant" will make the connection with the database and get the address data to then define the location of the Restaurant on the map.



**Trip Advisor API Integration**

We are using this integration as a form of customer information and visualization already infomed by Trip Advisor.. When you enter the point on the map, you will obtain the qualifications already acquired. This will tell you how many stars and only a brief view of the comments of the restaurant gets. Our System will not have the availability to include or change any functions given by this integration, it will only work with visualization system.  
  
Here we have some Common Elements and API Endpoints used for it.

http[s]://**{subdomain}**.ean.com/ean-services/rs/hotel/v3/**{request name}**?&**{request element 1}&{request element 2}&...**

A fixed set of endpoints, each defined by an ean.com subdomain, is used for all protocols.

Common request elements are used to provide authentication details, desired currencies and locales, and to provide customer session and usage details for each API service.

***TripAdvisor*** *Integration Requirements & Guidelines*  **Part 1 – Coverage Requirements**  
You must show TripAdvisor’s Licensed Content (as set out below) on at least 80% of the overall properties listed on your site.   
**Part 2 - TripAdvisor Brand Guidelines**

**DISPLAY OF LICENSED CONTENT ON DESKTOP AND MOBILE**

**I. All displays of Licensed Content on Licensee Site will be subject to approval by EAN before launch.**

**II. Spelling of “TripAdvisor”**

Licensee must spell “TripAdvisor” correctly in all cases (i.e. as a single word with no space between ‘Trip’ and ‘Advisor’, with a capital ‘T’ and a capital ‘A’).

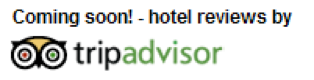
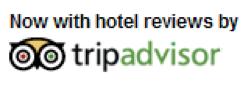
**III. TripAdvisor logo on Licensee Site home page**

Within one month of the Effective Date and for the duration of the Agreement, Licensee will include on the Partner Site homepage a TripAdvisor logo with an introductory sentence, e.g. “<Logo TripAdvisor> reviews – coming soon!”. The logo must be placed above the fold.   
Licensee shall use the below logo. Licensee will not store and locally serve TripAdvisor logos.



*Example mock-ups:*

Before the Launch Date:                                          After the Launch Date:

**IV. TripAdvisor logo on Licensee Site property pages**

All displays of Licensed Content on Licensee Site property pages must be accompanied by a TripAdvisor logo. The logo must be positioned as close as possible to the Licensed Content. When referencing TripAdvisor as the source of content, refer to it as “Powered by TripAdvisor"  
*Example:*  


**V. TripAdvisor Ratings Bar for hotels**

The TripAdvisor Ratings Bar must be used with the owl eyes and five green concentric circles for the overall rating of each property. The bar always has five circles whether the score is 5 out of 5 or 1 out of 5.   
owlcircles   
3 out of 5 is shown thus:  
3circs   
Sizing: 118 x 20 pixels

The TripAdvisor Ratings Bar must be shown for each property with the text “TripAdvisor Traveler Rating” and indicate the number of reviews on which it is based. This number is dynamic and must be kept updated.

*Example:*



TripAdvisor ratings bars must be served direct from TripAdvisor urls. Partner will not store and locally serve ratings bars.

**The Licensee will make all displays of Licensed Content non-indexable by search engines.**

**VI.** **TripAdvisor Content Policies**

Licensee will comply with TripAdvisor’s content policies at URL <http://help.tripadvisor.com/articles/200614797-Our-guidelines-for-traveler-reviews> (as such policies may be updated from time to time by TripAdvisor) to ensure the integrity of Licensed Content

**FACEBOOK API INTEGRATION**

**brief**

Facebook is a popular free social networking website that allows registered users to create profiles, upload photos and video, send messages and keep in touch with friends, family and colleagues.  
A ideia principal vai ser o usuário utilizar esta ferramenta como forma de acesso (com dados já existentes) ao nosso aplicativo web.  
  
how it works?

We provide a button as login called "Login with Facebook" for the user to access into it and durant the process will send a request for the user's email automatic, where the profile authenticity will be checked.

to make this button work, we need to create an App ID with Facebook and then the implement login with the following steps:

1. [**Check the login status**](https://developers.facebook.com/docs/facebook-login/web#checklogin) to see if someone's already logged into your app. During this step, you also should check to see if someone has previously logged into your app, but is not currently logged in.
2. [**Log people in**](https://developers.facebook.com/docs/facebook-login/web#logindialog), either with the login button or with the login dialog, and ask for a set of data permissions.
3. [**Log people out**](https://developers.facebook.com/docs/facebook-login/web#logout).

***check the login status***

The first step when loading your web page is figuring out if a person is already logged into your app with Facebook login. You start that process with a call to [FB.getLoginStatus](https://developers.facebook.com/docs/reference/javascript/FB.getLoginStatus). That function will trigger a call to Facebook to get the login status and call your callback function with the results.

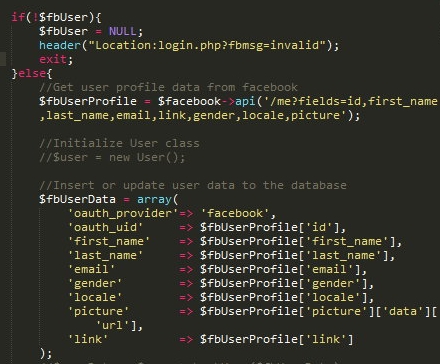
***Log People In***

If people using your app aren't logged into your app or not logged into Facebook, you can use [the Login dialog](https://developers.facebook.com/docs/facebook-login/overview/#logindialog) to prompt them to do both. Various versions of the dialog are shown below.  
If they aren't logged into Facebook, they'll first be prompted to log in and then move on to logging in to your app. The JavaScript SDK automatically detects this, so you don't need to do anything extra to enable this behavior.

***Log People Out***

You can log people out of your app by attaching the JavaScript SDK function FB.logout to a button or a link, as follows:



******

https://developers.facebook.com/docs/reference/php/ ???   
where all this code??????

did you downloade?