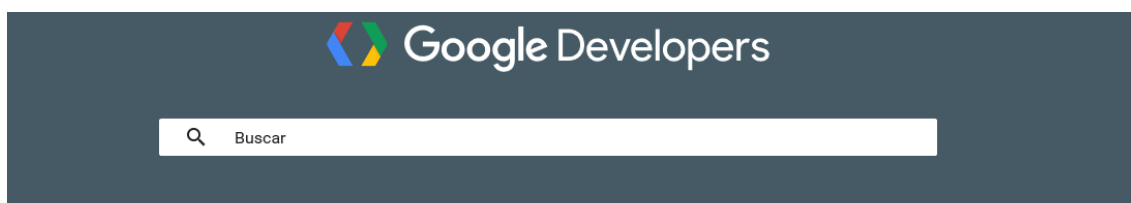


Google APIs con .NET

Por: Enrique Hernández González

Junio de 2016, ver. 1.0.0.

En Google Developers (<https://developers.google.com/>) existe una gran variedad de productos, los cuales podemos usar en el desarrollo de nuestras aplicaciones.



Plataformas y productos principales



Android



Cloud Platform



Firebase



Maps



Web



Todos los productos

¿Cómo determinamos que productos usar?

Lo primero que necesitamos es conocer los productos que nos ofrece Google, para que tecnologías están dirigidos, que limitaciones técnicas existen (hardware, software, costos, etcétera) y las limitantes del negocio (en caso de aplicar).

Una ventaja que nos ofrece Google, es que cada producto cuenta con información de soporte, la mayoría de los casos, con ejemplos y código fuente, para un mejor entendimiento.

Navega por orden alfabético

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z #

A

Admin API

AdMob

Ads

AdSense Custom Search Ads

AdSense Host API

Google AdWords

AdWords App Promotion

AdWords Scripts

Google AMP Cache

Analytics Core Reporting API

Admin SDK

AdMob by Google

AdSense

AdSense for Shopping

AdSense Management API

AdWords API

AdWords Remarketing Tag

AJAX Crawling (Deprecated)

Google Analytics

Analytics Embed API

Productos populares



Ads



Analytics



Android



Cast



Cloud Platform



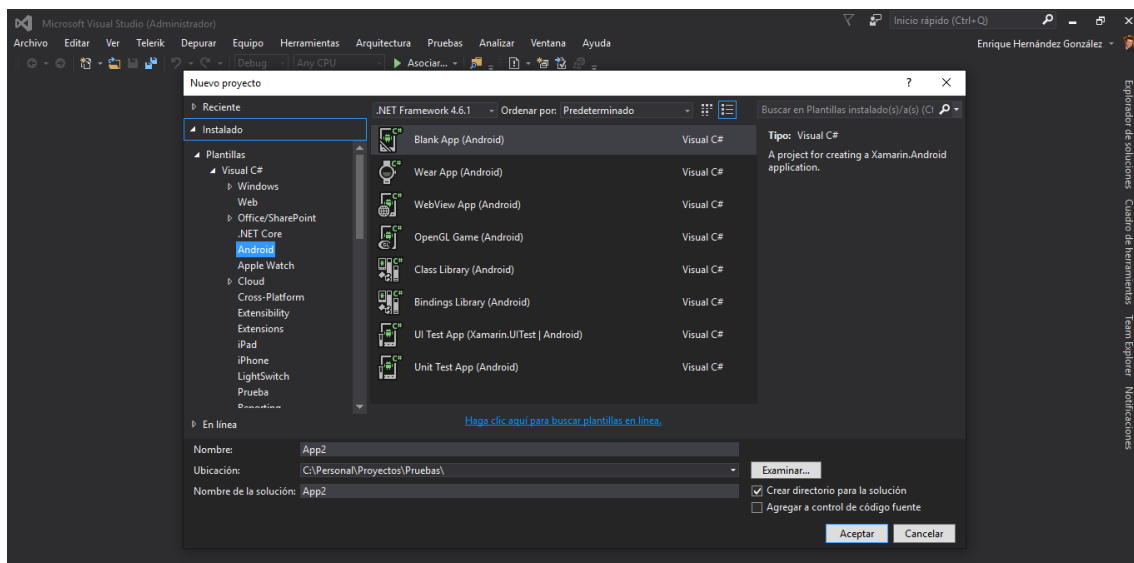
Chrome

¿Por qué usar Google con Visual Studio .NET?

La idea es aprovechar la infraestructura de Google, sus casos de éxito y buenas prácticas, para desarrollar software de forma más rápida y en muchos casos, usar productos sólidos que no tendríamos de forma nativa con Visual Studio .NET; uno de los ejemplos es Google Maps, el cual detallaremos más adelante.

¿Cómo integramos los productos de Google a Visual Studio .NET?

Debemos de tomar en cuenta, que con Visual Studio .NET se pueden desarrollar diferentes tipos de proyectos, como son de: Escritorio, Windows 10 Universal, Web, etcétera, por otro lado, es importante remarcar que en la actualidad se pueden desarrollar aplicaciones nativas para IOS y Android.



Como se mencionó anteriormente cada producto de Google está dirigido o desarrollado para ciertas tecnologías, por lo tanto, solamente se podrán usar los que sean compatibles con el tipo de proyecto seleccionado en Visual Studio .NET.

Existen una serie de APIs (<https://developers.google.com/api-client-library/dotnet/>), las cuales se integran directamente y proporcionan una forma fácil de acceder a los productos de Google.



Easily access Google APIs from .NET

Google services such as [Calendar](#), [Analytics](#), and [Google+](#) allow users to store personal information and manage their activities online.

Most Google services have web APIs that .NET developers can use to give their applications access to this information with the user's authorization.



Con estas librerías podemos usar la autenticación de Google, verificar los contactos, acceder al calendario, emplear el traductor, administrar YouTube y Gmail, entre muchas cosas más. Las APIs manipulan aproximadamente un 20% de los productos que actualmente existen (junio de 2016).

Caso Práctico

Antecedentes

La firma “Franquicias y Asociados S. A. de C. V.” cuenta con 10 empresas en la República Mexicana y cada una de ellas, tiene “n” número de sucursales.

La firma se propuso entre sus metas aumentar el número de franquicias en los próximos años, así como mantener las ya existentes.

Requerimientos del cliente

Contar con una herramienta para dar a conocer a sus clientes, sus diferentes empresas y sucursales, con la finalidad de obtener información detallada y de esta manera facilitar la comunicación entre ellos.

Propuesta técnica


Desarrollar una página en el portal web institucional de la firma denominada “Contáctenos”, la cual presentará un formulario con los datos del cliente y un botón para enviarlos vía correo, como funcionalidad adicional, contará con una lista de empresas, en donde al seleccionar una, se presentará la información de las diferentes sucursales que la integran.

Diagrama de Fase I del formulario de contacto. El formulario está dividido en dos secciones por una línea vertical punteada. A la izquierda, hay tres campos de texto etiquetados 'Nombre:', 'Correo:' y 'Comentario:', con un botón 'Enviar' debajo. A la derecha, hay dos columnas: 'Empresas' con tres líneas horizontales y 'Sucursales' con un cuadro vacío.

Fase I

Diagrama de Fase II del formulario de contacto. El formulario está dividido en dos secciones por una línea vertical punteada. A la izquierda, hay tres campos de texto etiquetados 'Nombre:', 'Correo:' y 'Comentario:', con un botón 'Enviar' debajo. A la derecha, hay dos campos de lista desplegable etiquetados 'Empresas:' y 'Sucursales:', y una sección 'Información' con dos líneas horizontales.

Fase II

Nombre: <input type="text"/>	Empresas: <input type="text"/>
Correo: <input type="text"/>	Información 
Comentario: <input type="text"/>	
<input type="button" value="Enviar"/>	

Fase III

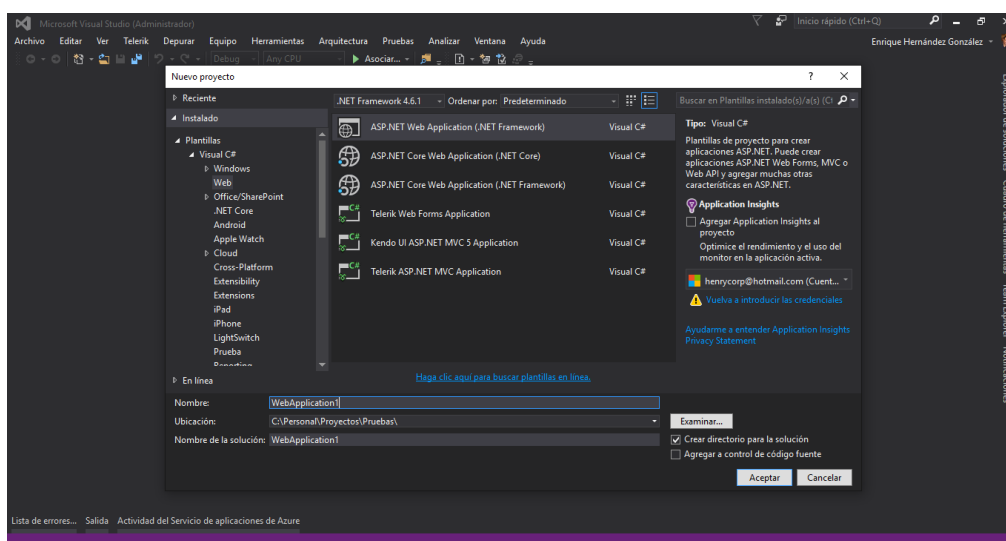
También se deberá desarrollar una aplicación (CRUD), la cual permitirá administrar la información de las empresas y sucursales, ya que en cualquier momento pueden cambiar sus datos, dejar de pertenecer al grupo, y según las metas planteadas por la firma, agregar nuevos integrantes.

Empresa: <input type="text"/>	Empresas: <input type="text"/>
Descripción: <input type="text"/>	Sucursal: <input type="text"/>
<input type="button" value="Actualizar"/>	
	Latitud: <input type="text"/>
	Longitud: <input type="text"/>
<input type="button" value="Agregar"/>	

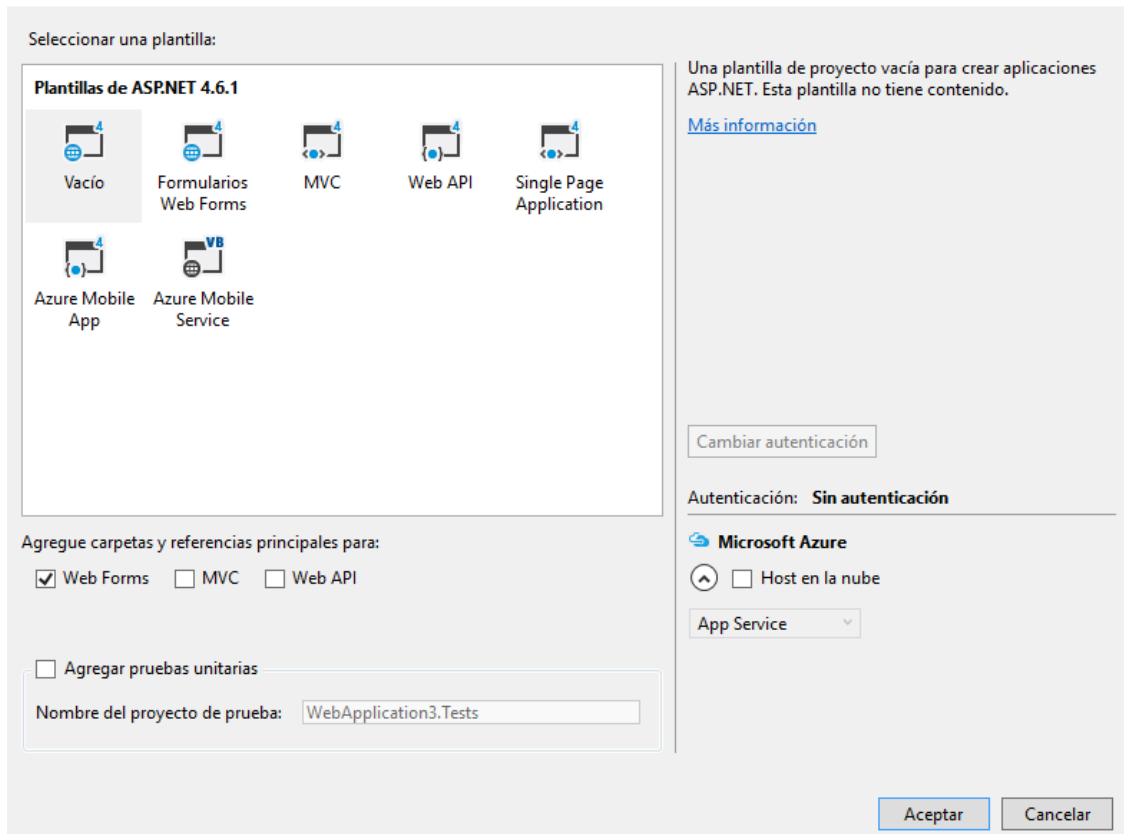
Para el caso práctico tomaremos en cuenta que los requerimientos técnicos de la firma condicionan el uso de Visual Studio .NET.

Desarrollo de la aplicación:

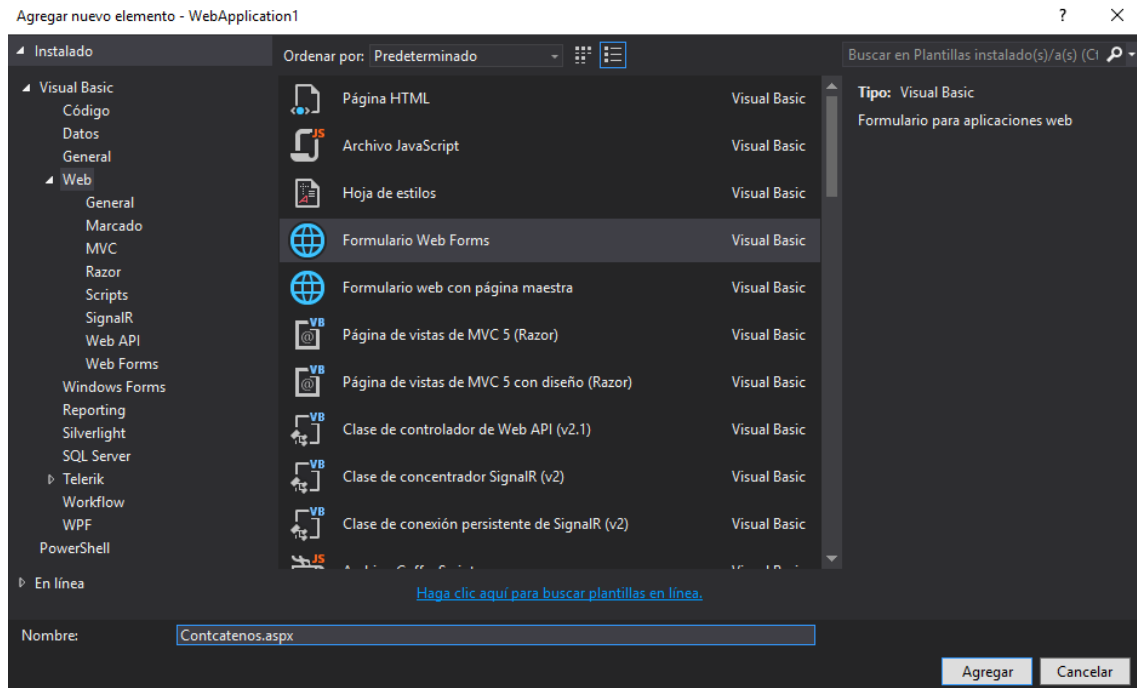
1.- Creamos un nuevo proyecto.



2.- Seleccionamos la plantilla a utilizar.



3.- Agregamos el formulario de "Contáctenos".



Un Formulario web de Visual Studio .NET (.aspx), realmente es un archivo HTML en su diseño, podemos usar casi cualquier objeto que funcione a este nivel (cliente), la diferencia es que los controles nativos ejecutarán sus procesos en el servidor.

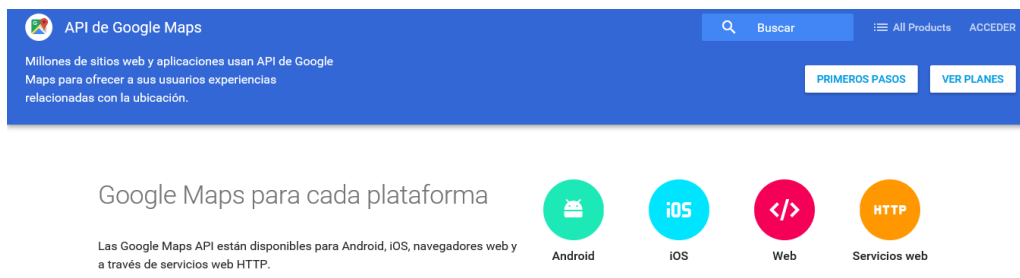
```
<%@ Page Language="vb" AutoEventWireup="false" CodeBehind="Contactenos.aspx.vb"
Inherits="WebApplication1.Contactenos" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>

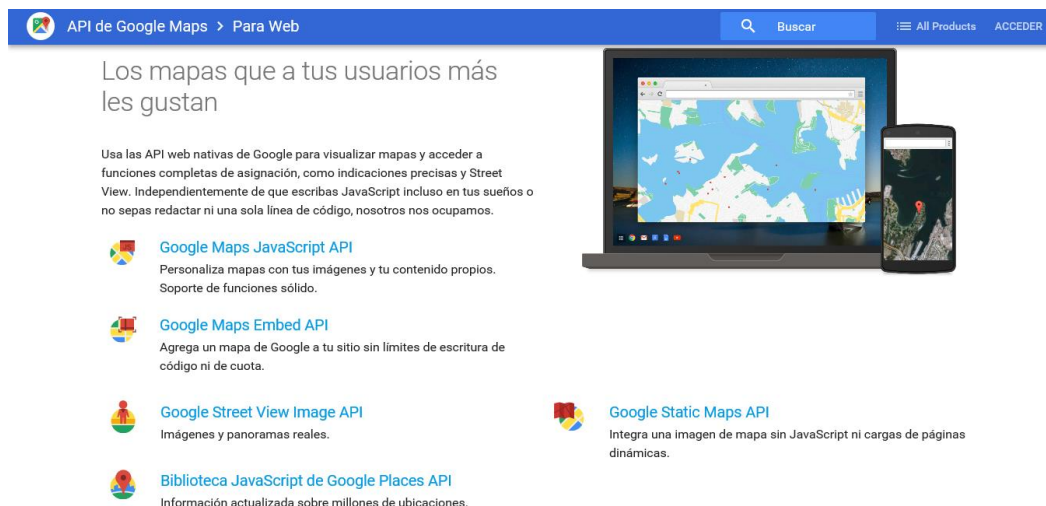
</div>
</form>
</body>
</html>
```

4.- Agregamos los objetos que integrarán el formulario de Contáctenos, considerando, que la tecnología que estamos usando no tiene un control nativo para visualizar y manipular un mapa, por lo tanto, tendremos que usar un producto de terceros y uno de los más sólidos y completos para esta tarea, es sin duda Google Maps (<https://developers.google.com/maps/>).



Como se comentó al principio, es importante conocer que productos podemos usar, de acuerdo a la tecnología y aspectos técnicos, las opciones que nos pueden servir para cumplir con el requerimiento de este caso práctico son:

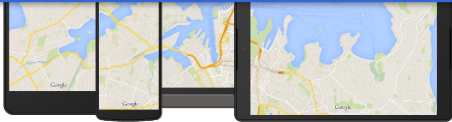
Web











Servicios web

API de Google Maps > Servicios web

Accede a interfaces HTTP, que proporcionan datos geográficos como geocodificación, indicaciones, elevación, sitio e información de zonas horarias.



-  **Google Maps Geocoding API**
Realiza conversiones de direcciones a coordenadas geográficas.
-  **Google Places API Web Service**
Implementa el autocompletado y agrega información actualizada sobre millones de ubicaciones a tu sitio o aplicación.
-  **Google Maps Elevation API**
Datos de elevación para cualquier punto en el mundo.
-  **Google Maps Distance Matrix API**
Calcula el tiempo de viaje y la distancia para varios destinos.
-  **Google Maps Roads API**
Habilita la funcionalidad snap-to-road para rastrear de forma precisa rutas de navegación de GPS.
-  **Google Maps Time Zone API**
Proporciona datos de zonas horarias para cualquier lugar del mundo.
-  **Google Maps Geolocation API**
Encuentra una ubicación según información de torres celulares y nodos WiFi.
-  **Google Maps Directions API**
Calcula indicaciones entre varias ubicaciones.

El producto que integra la funcionalidad de todas las APIs es Google Maps JavaScript.
(<https://developers.google.com/maps/documentation/javascript>).

La integración inicial se haría de la siguiente manera:

```
<%@ Page Language="vb" AutoEventWireup="false" CodeBehind="Contactenos.aspx.vb"
Inherits="WebApplication1.Contactenos" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title></title>
<style>
<ul>
<li>
<ul>
<li>
<ul>
<li>Nombre</li>
<li>
<asp:TextBox ID="txtNombre" runat="server"></asp:TextBox>
</li>
<li>Correo</li>
<li>
<asp:TextBox ID="txtCorreo" runat="server" TextMode="Email"></asp:TextBox>
</li>
<li>Comentario</li>
<li>
<asp:TextBox ID="txtComentario" runat="server" TextMode="MultiLine"></asp:TextBox>
</li>
<li>&nbsp;</li>
<li>
<asp:Button ID="btnEnviar" runat="server" Text="Enviar" />
</li>
<li>Empresas</li>
<li>
<asp:DropDownList ID="ddlEmpresas" runat="server" AutoPostBack="True"></asp:DropDownList>
</li>
<li>Sucursales</li>
</ul>
</li>
</ul>
</div>
</form>
```

```

</li>
<asp:ListBox ID="libSucursales" runat="server"></asp:ListBox>
</li>
<li>&nbsp;</li>
<li>
  <div id="map"></div>
</li>
</ul>
</div>

<script>
function initMap() {
var map = new google.maps.Map(document.getElementById('map'), {
zoom: 14,
center: { lat: 17.980151, lng: -92.918084 }
});
}
</script>
<script src="https://maps.googleapis.com/maps/api/js?key=****&signed_in=true&callback=initMap" async
defer"></script>
</form>
</body>
</html>

```

5.- Nuestro siguiente paso será hacer toda la funcionalidad necesaria, diseño, acceso a datos, geocodificación, presentar en el mapa cada sucursal, etcétera.

Consideraciones

Por último, necesitamos tomar en cuenta el licenciamiento, Google Maps no es completamente gratis, esta información la podemos encontrar en <https://developers.google.com/maps/pricing-and-plans/>.

API de Google Maps		
	requisitos	
iOS	ESTÁNDAR	PREMIUM
Google Maps SDK for iOS	Unbegrenzte kostenlose Nutzung	Volumenbasierter Preis
Google Places API for iOS	1.000 solicitudes gratuitas por día predeterminadas, aumentadas a 150.000 solicitudes gratuitas por día después de la validación de la tarjeta de crédito Extras gratis para aplicaciones que cumplen con los requisitos	---
Web	ESTÁNDAR	PREMIUM
Google Maps JavaScript API	Gratis hasta exceder 25.000 cargas de mapas por día durante 90 días consecutivos	Volumenbasierter Preis
Google Static Maps API	USD 0,50 / 1000 cargas de mapas adicionales por encima de 25.000 por día después de alcanzar la carga de mapa 25.000 / límite de uso de 90 días, hasta 1.000.000 diariamente	Funciones Premium: • Sin anuncios, garantizado • Tamaño de imagen hasta de 2048 x 2048 píxeles
Google Street View Image API		
Google Maps Embed API	Unbegrenzte kostenlose Nutzung	---



<http://www.xnetcorporativo.com>



EnriqueHG1972



@EnriqueHG1972

La adquisición de cualquier conocimiento es siempre útil al intelecto, que sabrá descartar lo malo y conservar lo bueno.
Leonardo Da Vinci