

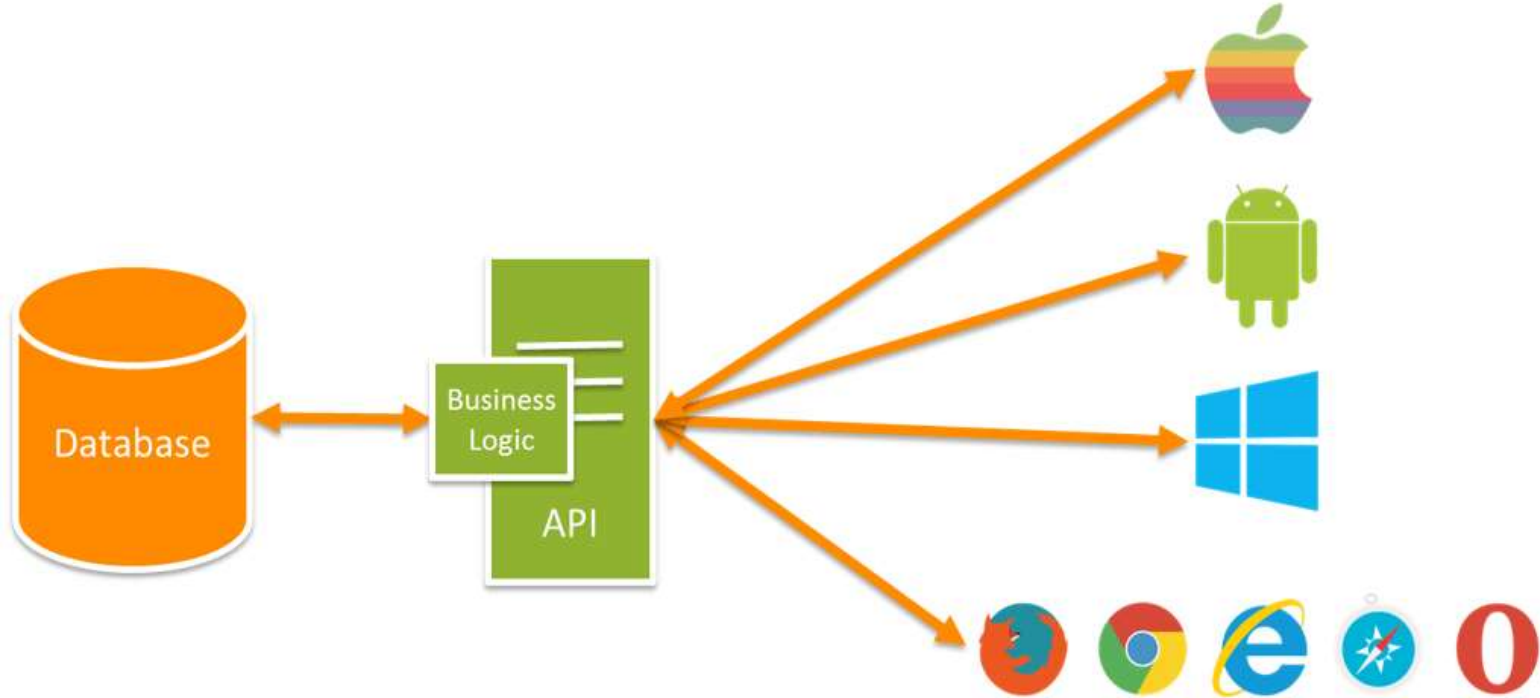


Web Api - Rest

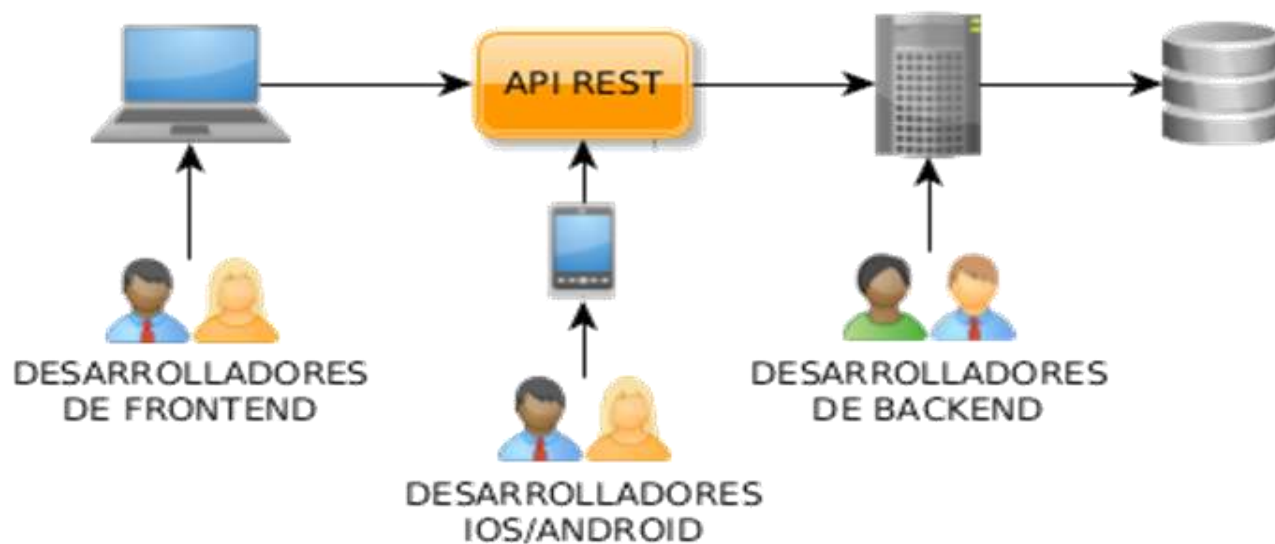


C#
Csharp





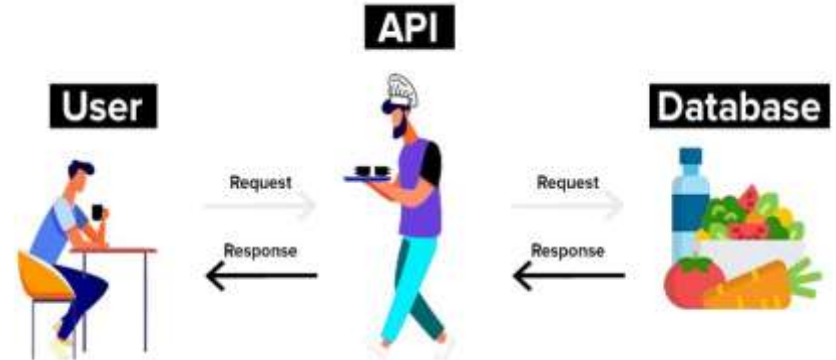
APIs





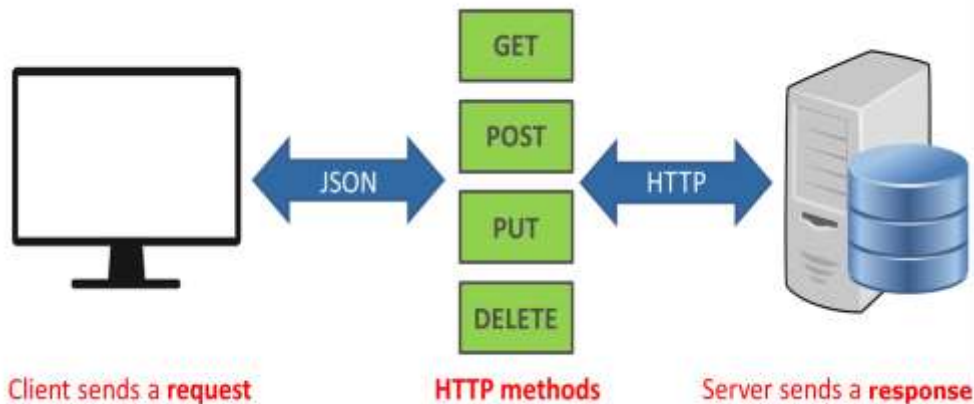
Qué es una API

El término API es una abreviatura de Application Programming Interfaces, que en español significa interfaz de programación de aplicaciones. Se trata de un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones, permitiendo la comunicación entre dos aplicaciones de software a través de un conjunto de reglas.





HTTP define métodos que son utilizados para diferentes circunstancias, por lo que trataremos de listar lo más relevantes y más utilizamos en la construcción de servicios REST, los métodos son los siguientes:



GET: Es utilizado únicamente para consultar información al servidor, muy parecidos a realizar un SELECT a la base de datos. No soporta el envío del payload

POST: Es utilizado para solicitar la creación de un nuevo registro, es decir, algo que no existía previamente, es decir, es equivalente a realizar un INSERT en la base de datos. Soporta el envío del payload.

PUT: Se utiliza para actualizar por completo un registro existente, es decir, es parecido a realizar un UPDATE a la base de datos. Soporta el envío del payload.

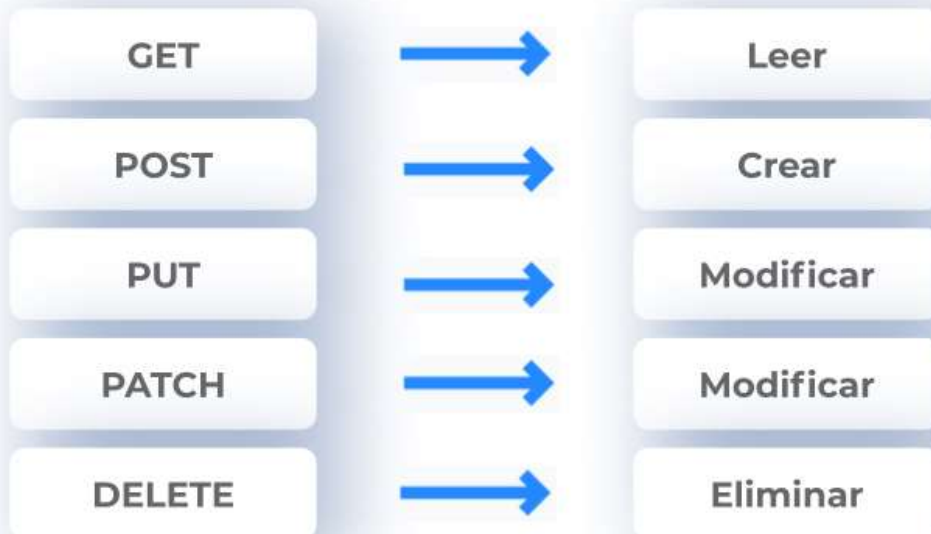
PATCH: Este método es similar al método PUT, pues permite actualizar un registro existente, sin embargo, este se utiliza cuando actualizar solo un fragmento del registro y no en su totalidad, es equivalente a realizar un UPDATE a la base de datos. Soporta el envío del payload

DELETE: Este método se utiliza para eliminar un registro existente, es similar a DELETE a la base de datos. No soporta el envío del payload.

Instructor: Adalberto Cárcamo A.



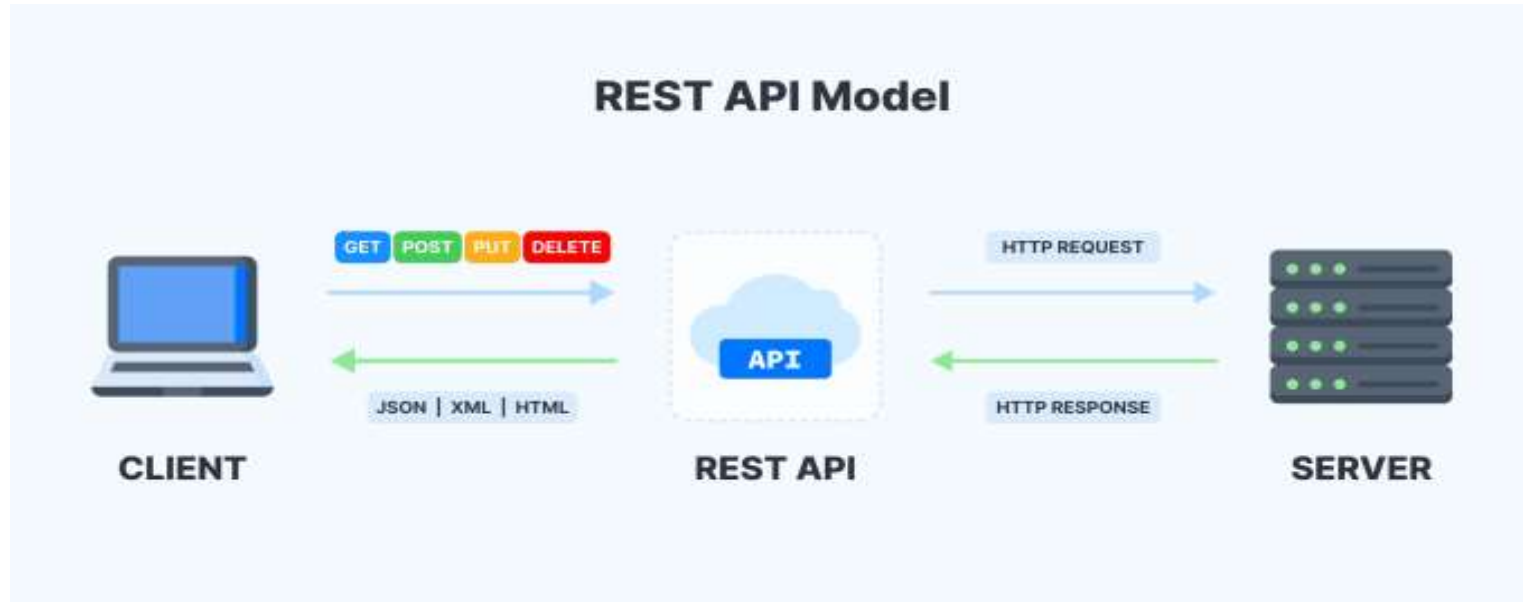
Métodos HTTP





Realizaremos un ejemplo práctico:

API Rest CRUD con el lenguaje de programación C# y SQL Server



Instructor: Adalberto Cárcamo A.



1. Creación de la base de datos:

```
CREATE DATABASE DB_API_DATOS
```

```
USE DB_API_DATOS
```



BD



2. Creamos la siguiente tabla de usuario:

```
create table USUARIO(  
IdU varchar(15) primary key,  
Nombres varchar(80),  
Telefono varchar(60),  
Correo varchar(40),  
Ciudad varchar(60),  
FechaIngreso datetime default getdate()  
)
```



3. Ingresamos 5 registros de prueba en la tabla de usuario:



```
INSERT USUARIO (IdU, Nombres, Telefono, Correo, Ciudad)
VALUES ( '545454', 'Sara', 3445654321, 'saracar@gmail.com', 'Cartagena' )
```



4. Creamos la capa de datos :

Iniciamos con los procedimientos almacenados

- a. Registrar
- b. Actualizar
- c. Eliminar
- d. Consultar
- e. Listar





a. Registrar Usuario



```
create procedure usp_registrar(  
@idu varchar(15),  
@nombres varchar(80),  
@telefono varchar(60),  
@correo varchar(40),  
@ciudad varchar(60)  
)  
as  
begin  
insert into USUARIO(IdU,Nombres,Telefono,Correo,Ciudad)  
values  
(  
@idu,  
@nombres,  
@telefono,  
@correo,  
@ciudad  
)  
end
```



b. Actualizar Usuario



```
create procedure usp_actualizar (  
@idu varchar(15),  
@nombres varchar(80),  
@telefono varchar(60),  
@correo varchar(40),  
@ciudad varchar(60)  
)  
as  
begin  
update USUARIO set  
Nombres = @nombres,  
Telefono = @telefono,  
Correo = @correo,  
Ciudad = @ciudad  
where IdU = @idu  
)  
end
```



c. Eliminar Usuario



```
create procedure usp_eliminar(  
@idu varchar(15)  
)  
as  
begin  
  
delete from usuario where IdU = @idusuario  
  
end
```



d. Consultar Usuario



```
create procedure usp_obtener
@idu varchar(15)

as
begin
    select * from usuario where IdU = @idu
end
```



e. Listar Usuario



```
create procedure usp_listar  
as  
begin  
  
select * from usuario  
end
```




Creamos ahora la aplicación Web vacía con la opción API web en Visual Studio /Aplicación Web

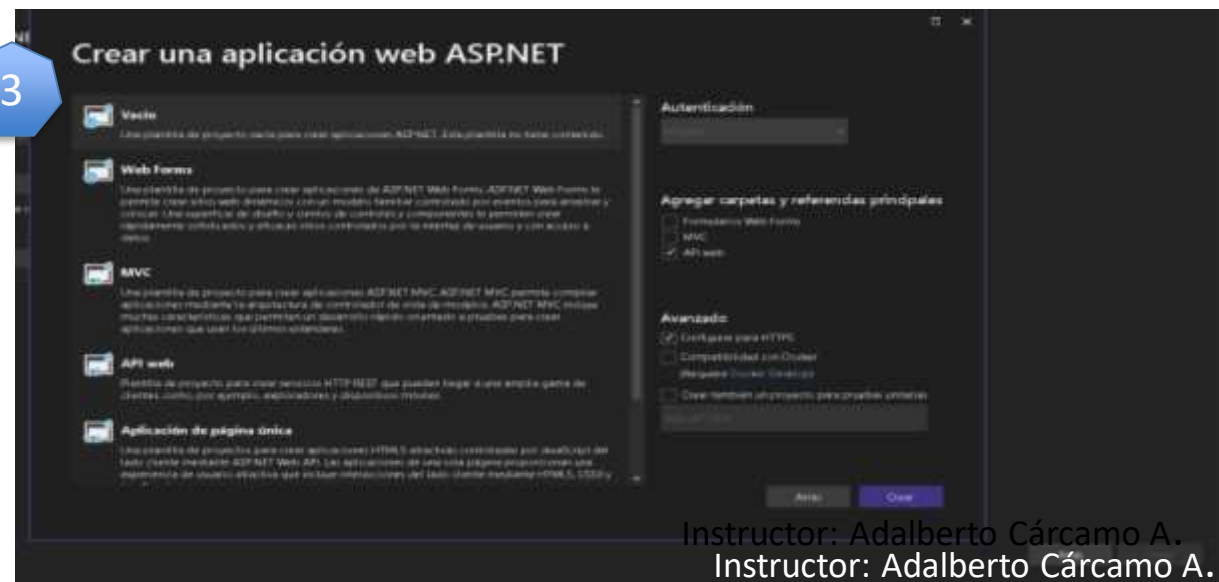
1



2



3



Instructor: Adalberto Cárcamo A.
Instructor: Adalberto Cárcamo A.

The screenshot shows the Visual Studio IDE with the 'App API' project open. The main window displays the 'ASP.NET' introduction page, which includes a sidebar on the left with 'Introducción', 'Servicios conectados', and 'Publicar'. The main content area has the heading 'ASP.NET' and a subheading 'Obtenga más información sobre la plataforma .NET, cree su primera aplicación y exténdala hasta la nube.' Below this are three cards: 'Compilar la aplicación', 'Conectarse a Azure', and 'Obtenga información sobre el IDE'. The right sidebar shows the 'Explorador de soluciones' (Solution Explorer) with a tree view of the project structure: 'App API' (1 of 1 projects) containing 'Connected Services', 'Properties', 'References', 'App_Data', 'App_Start', 'Controllers', 'Models', 'Global.asax', 'packages.config', and 'WebResource'. A red arrow points from a text box to the 'App API' project in the Solution Explorer.

Nos presenta la siguiente estructura

Instructor: Adalberto Cárcamo A.

The screenshot shows the Visual Studio IDE interface. The main window displays the 'ASP.NET' introduction page, which includes links to 'Compilar la aplicación', 'Conectarse a Azure', and 'Obtenga información sobre el IDE'. The right-hand pane shows the 'Explorador de soluciones' (Solution Explorer) with a tree view of the project files. A red arrow points to the 'Models' folder in the tree view.

En la carpeta **Models** (modelo) creamos una clase donde tendremos las propiedades.

Instructor: Adalberto Cárcamo A.



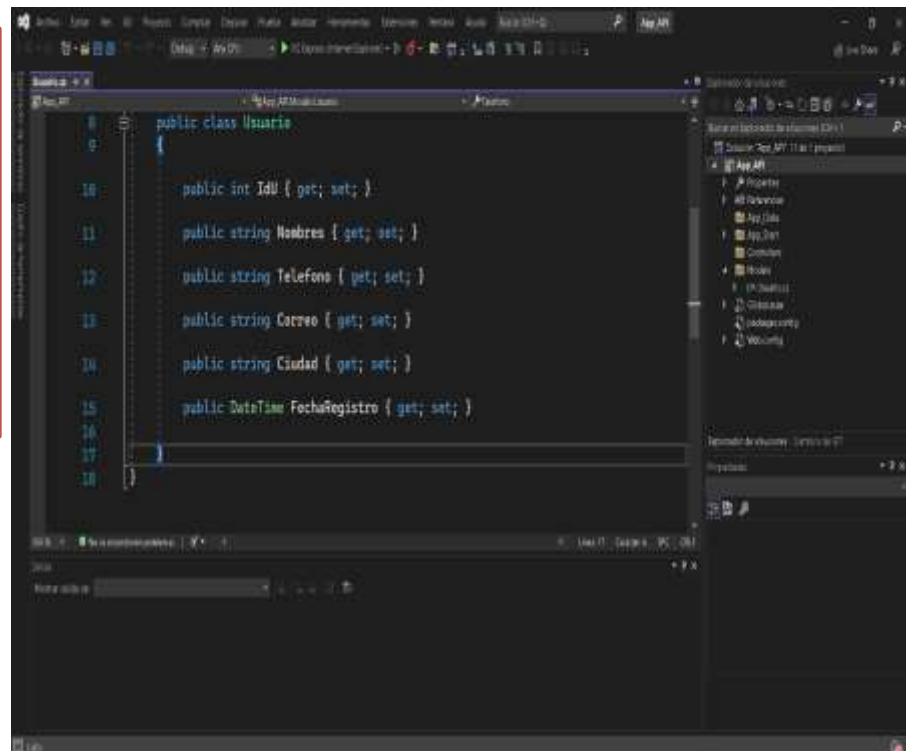
Nos presenta la siguiente estructura:

A screenshot of a code editor window. The title bar shows 'Usuario.cs' and 'App_API: Introducción'. The left sidebar shows the 'App_API' project structure. The main editor area displays the following C# code:

```
1 using System;
2     using System.Collections.Generic;
3     using System.Linq;
4     using System.Web;
5
6 namespace App_API.Models
7 {
8     public class Usuario
9     {
10     }
11 }
```

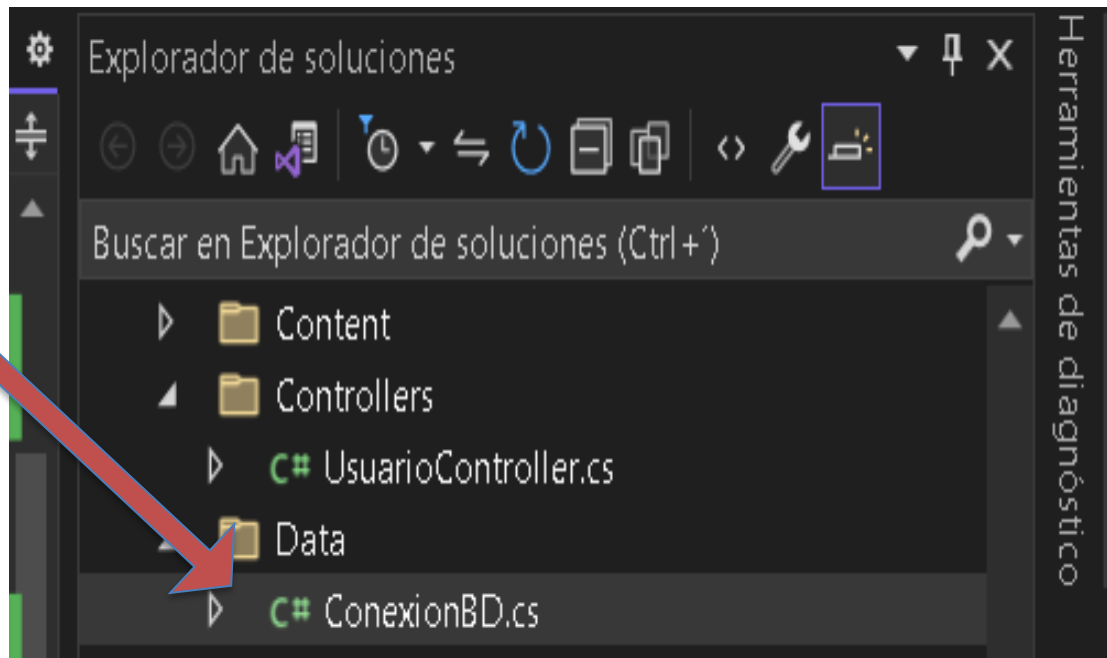


```
public class Usuario
{
    public int IdU { get; set; }
    public string Nombres { get; set; }
    public string Telefono { get; set; }
    public string Correo { get; set; }
    public string Ciudad { get; set; }
    public DateTime FechaIngreso { get; set; }
}
```





Creamos una carpeta llamada Data y luego asignamos una clase llamada ConexionBD que será compartida por el instructor.

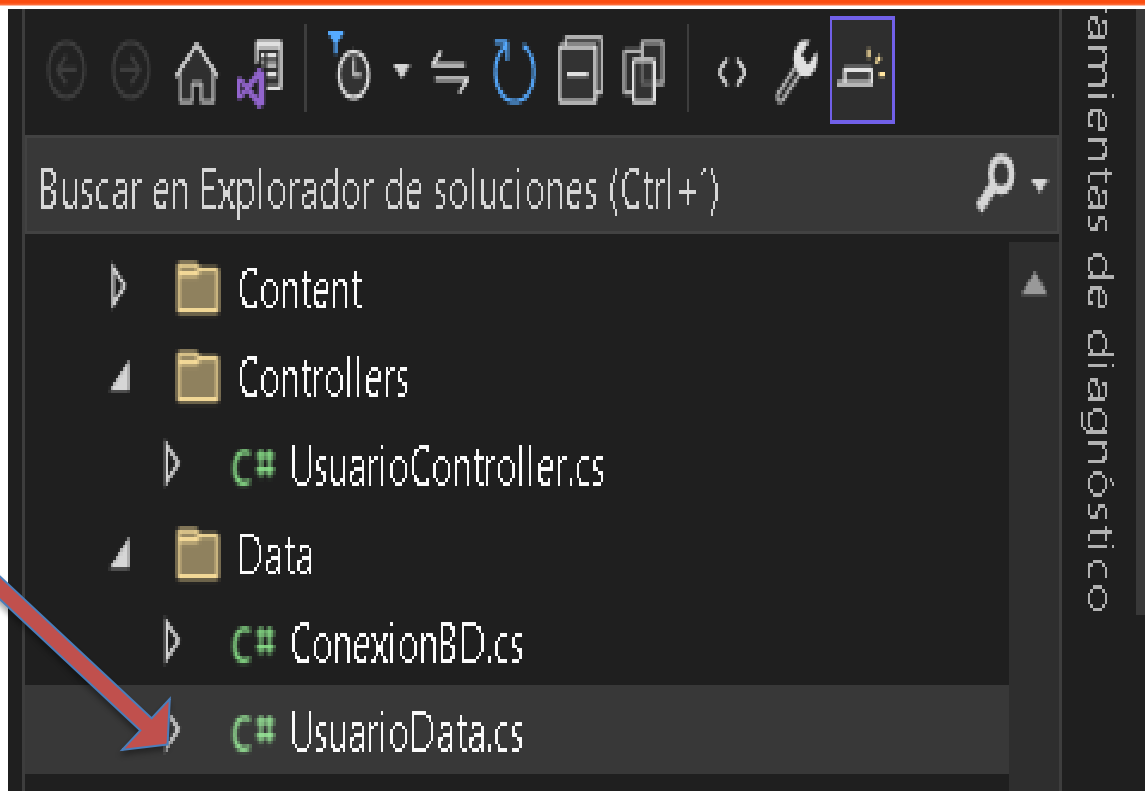




Creamos una clase llamada UsuarioData en la carpeta Data.

En esta clase creamos la lógica de negocio de la Api

Donde desarrollaremos los métodos correspondientes.





Creamos el método Registrar Usuario

```
public static bool registrarUsuario(Usuario oUsuario)
{
    ConexionBD objEst = new ConexionBD();
    string sentencia;
    sentencia = "EXECUTE usp_registrar '" +
oUsuario.IdU + "','" + oUsuario.Nombres + "','" +
oUsuario.Telefono + "','" + oUsuario.Correo + "','" +
oUsuario.Ciudad + "'";

    if (!objEst.EjecutarSentencia(sentencia, false))
    {
        objEst = null;
        return false;
    }
    else
    {
        objEst = null;
        return true;
    }
}
```



Creamos el método Actualizar Usuario

```
public static bool actualizarUsuario(Usuario oUsuario)
{
    ConexionBD objEst = new ConexionBD();
    string sentencia;
    sentencia = "EXECUTE usp_actualizar '" +
oUsuario.IdU + "','" + oUsuario.Nombres + "','" +
oUsuario.Telefono + "','" + oUsuario.Correo + "','" +
oUsuario.Ciudad + "'";

    if (!objEst.EjecutarSentencia(sentencia, false))
    {

        objEst = null;
        return false;

    }
    else
    {
        objEst = null;
        return true;
    }
}
```



Creamos el método Eliminar Usuario

```
public static bool eliminarUsuario(string id)
{
    ConexionBD objEst = new ConexionBD();
    string sentencia;
    sentencia = "EXECUTE usp_Eliminar '" + id + "'";

    if (!objEst.EjecutarSentencia(sentencia, false))
    {

        objEst = null;
        return false;

    }
    else
    {
        objEst = null;
        return true;
    }
}
```



Creamos el método Listar Usuario

```
public static List<Usuario> Listar()
{
    List<Usuario> oListaUsuario = new List<Usuario>();
    ConexionBD objEst = new ConexionBD();
    string sentencia;
    sentencia = "EXECUTE usp_consultar";

    if (objEst.Consultar(sentencia, false))
    {
        SqlDataReader dr = objEst.Reader;
        while (dr.Read())
        {
            oListaUsuario.Add(new Usuario()
            {
                IdU = Convert.ToInt32(dr["IdU"]),
                Nombres = dr["Nombres"].ToString(),
                Telefono = dr["Telefono"].ToString(),
                Correo = dr["Correo"].ToString(),
                Ciudad = dr["Ciudad"].ToString(),
                FechaIngreso =
                Convert.ToDateTime(dr["FechaIngreso"].ToString())
            });
        }

        return oListaUsuario;
    }
    else
    {
        return oListaUsuario;
    }
}
```



Creamos el método
obtener Usuario



```
public static List<Usuario> Obtener(string id)
{
    List<Usuario> oListaUsuario = new List<Usuario>();
    ConexionBD objEst = new ConexionBD();
    string sentencia;
    sentencia = "EXECUTE usp_obtener '"+ id +"'";

    if (objEst.Consultar(sentencia, false))
    {

        SqlDataReader dr = objEst.Reader;
        while (dr.Read())
        {
            oListaUsuario.Add(new Usuario()
            {
                IdU = Convert.ToInt32(dr["IdU"]),
                Nombres = dr["Nombres"].ToString(),
                Telefono = dr["Telefono"].ToString(),
                Correo = dr["Correo"].ToString(),
                Ciudad = dr["Ciudad"].ToString(),
                FechaIngreso = Convert.ToDateTime(dr["FechaIngreso"].ToString())
            });
        }

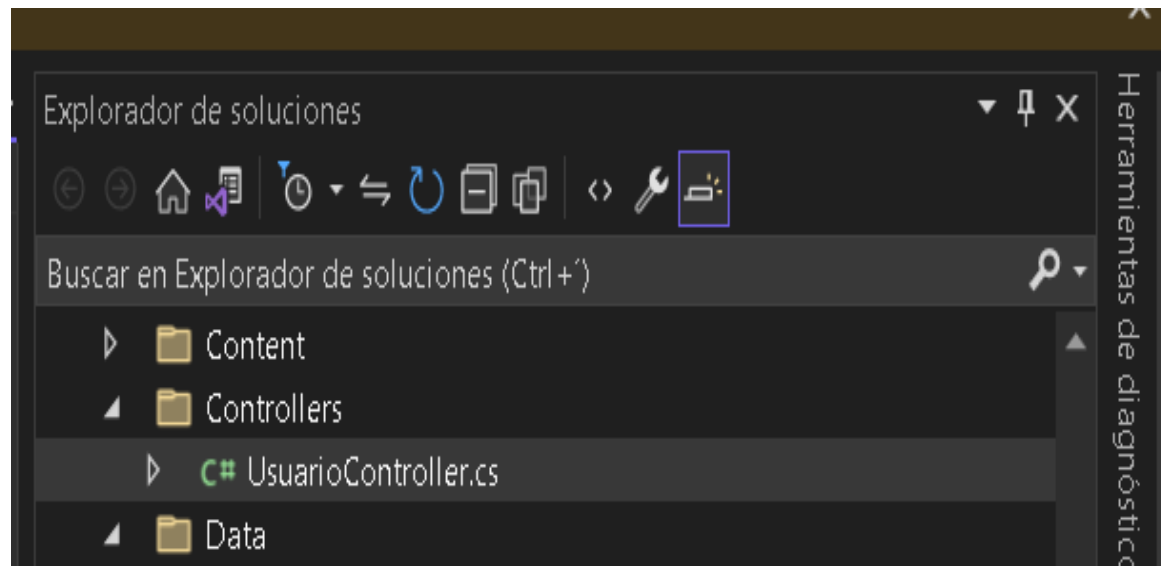
        return oListaUsuario;

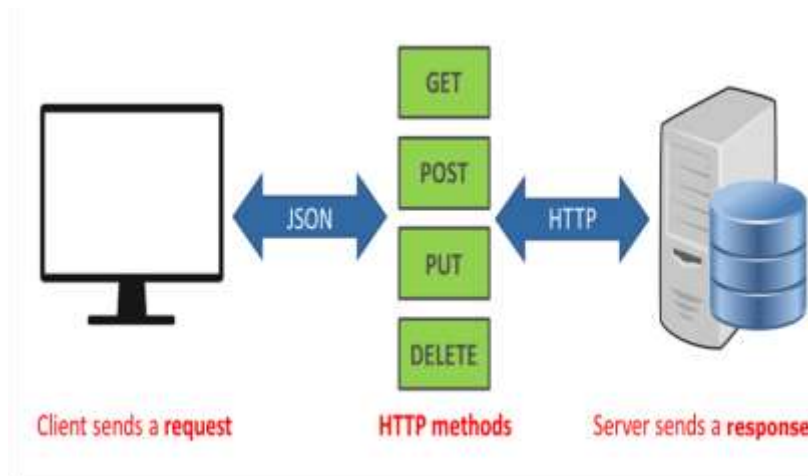
    }
    else
    {
        return oListaUsuario;
    }
}
```



Ahora creamos el controlador:
Dentro de la carpeta Controllers.

1. Click derecho agregar nuevo elemento.
2. C# / web /web Api
3. Escogemos Clase de controlador web Api
4. Asignamos el nombre UsuarioController.





```
// GET api/<controller>
public List<Usuario> Get()
{
    return UsuarioData.Listar();
}

// GET api/<controller>/5
public List<Usuario> Get(string id)
{
    return UsuarioData.Obtener(id);
}

// POST api/<controller>
public bool Post([FromBody] Usuario oUsuario)
{
    return UsuarioData.registrarUsuario(oUsuario);
}

// PUT api/<controller>/5
public bool Put([FromBody] Usuario oUsuario)
{
    return UsuarioData.actualizarUsuario(oUsuario);
}

// DELETE api/<controller>/5
public bool Delete(string id)
{
    return UsuarioData.eliminarUsuario(id);
}
```



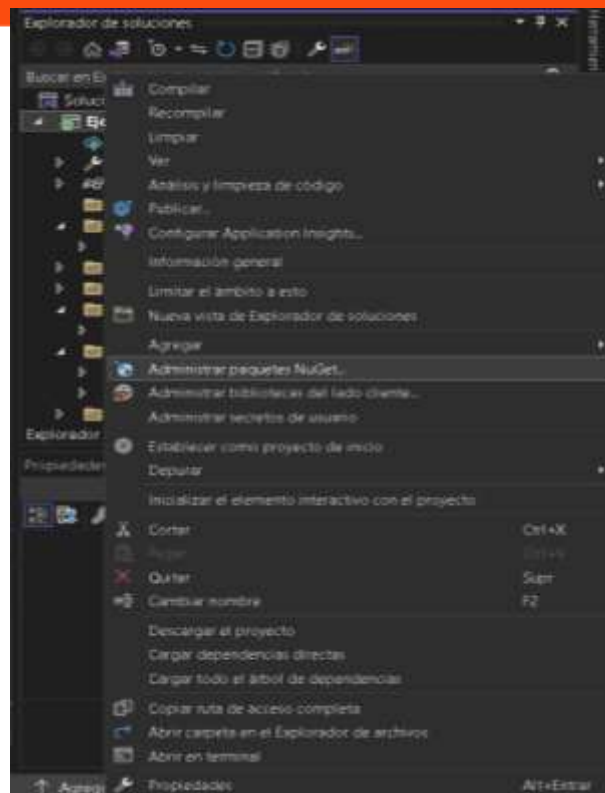
Para que los métodos de la Api puedan ser leídos desde la ruta que nos asigna la aplicación.

Debemos instalar el siguiente paquete:

Microsoft.AspNet.Web.WebApi.Cors



Click derecho en la solución y luego escogemos Administrador paquetes NuGet..



Instructor: Adalberto Cárcamo A.



En examinar buscamos el paquete y luego lo instalamos ,habilitando en la solución y el proyecto

NuGet: Ejemplo Api2

Administrador de paquetes NuGet: Ejemplo Api2

Origen del paquete: nuget.org

Paquete	Por	Descargas	Versiones
Microsoft.AspNetCore.WebApi.Cors	por Microsoft	42,1M descargas	5.2.9
Microsoft.AspNetCore.WebApi.Cors.ko	por Microsoft	11,8K descargas	5.2.9
Microsoft.AspNetCore.WebApi.Cors.zh-Hans	por Microsoft	16,5K descargas	5.2.9
Microsoft.AspNetCore.WebApi.Cors.ja	por Microsoft	27,1K descargas	5.2.9
Microsoft.AspNetCore.WebApi.Cors.de	por Microsoft	26,7K descargas	5.2.9

Microsoft.AspNetCore.WebApi.Cors (5.2.9)

Instalado: 5.2.9 Desinstalar

Versión: 5.2.9

Descripción: This package contains the components to enable Cross-Origin Resource Sharing (CORS) in ASP.NET Web API.

URL del proyecto: <https://www.asp.net/web-api>

Mostrar salida de: Depurar

```

11:55:55.000 [CLR v4.0.30319; /J:\MSVC\2\ROOF-1-13316555972563991]; "Anonymous Hosted DynamicMethods Assembly" cargado.
11:55:55.000 [CLR v4.0.30319; /J:\MSVC\2\ROOF-1-13316555972563991]; "C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Numerics\v4.0.0.0-17760"
11:55:55.000 [CLR v4.0.30319; /J:\MSVC\2\ROOF-1-13316555972563991]; "C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Data.resources\v4.0.0.0-17760"
El subproceso 84370 terminó con código 0 (0x0).
El subproceso 84370 terminó con código 0 (0x0).
El programa "[20754] 11:55:55.000" terminó con código 4294967296 (0xffffffff).
  
```

Lista de errores: Salida



En examinar buscamos el paquete y luego lo instalamos ,habilitando en la solución y el proyecto

The screenshot shows the Visual Studio interface with the NuGet Package Manager window open. The package list on the left includes several versions of **Microsoft.AspNet.WebApi.Core**. The right pane shows the details for version 5.2.9, which is installed. The command window at the bottom shows the output of a `dotnet publish` command, indicating that the application was successfully published to the specified output directory.

NuGet: Ejemplo Api2

Administrador de paquetes NuGet: Ejemplo Api2

Origen del paquete: nuget.org

Microsoft.AspNet.WebApi.Core por Microsoft 42,1M descargas

Instalado: 5.2.9 Desinstalar

Versión: 5.2.9

Descripción: This package contains the components to enable Cross-Origin Resource Sharing (CORS) in ASP.NET Web API.

Versión: 5.2.9

Autores: Microsoft

Licencia: Ver licencia

Fecha de publicación: martes, 31 de mayo de 2022 (11/05/2022)

URL del proyecto: <https://www.asp.net/web-api>

Mostrar salida de: Depurar

Output: [11/05/2022 14:00:00] "C:\Program Files\dotnet\cli\dotnet.exe" publish --self-contained --runtime win-x64 --output "C:\Users\user\source\repos\Ejemplo Api2\bin\Debug\Publish" --verbosity:quiet

El subproceso 83372 terminó con código 0 (0x0).

El subproceso 83372 terminó con código 0 (0x0).

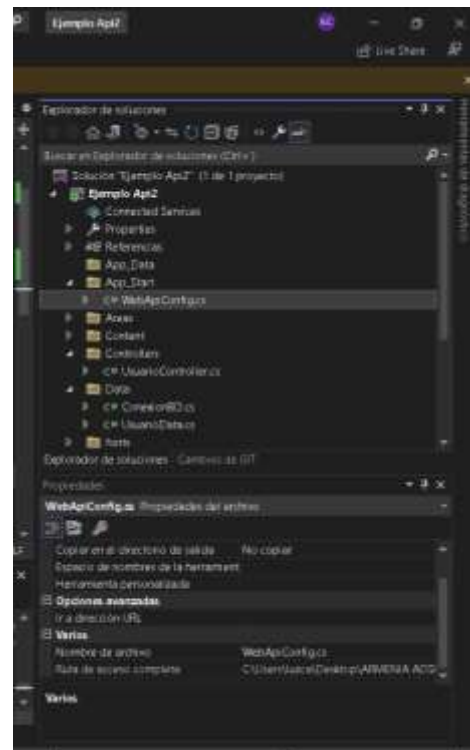
El programa "[20754] dotnet.exe" terminó con código 0 (0x0).



Abrimos el la clase WebConfig de la carpeta App_Start y asignamos el siguiente código que nos permitirá que la Api sea leída desde cualquier aplicación es importante primero usar los paquetes que instalamos :

```
using System.Web.Http;  
using System.Web.Http.Cors;
```

```
// Configuración y servicios de Web API  
var cors = new EnableCorsAttribute("*", "*",  
    "*");  
config.EnableCors(cors);
```



Instructor: Adalberto Cárcamo A.



Ejecutamos la Api y nos trae la url en este ejemplo seria el siguiente:

<http://localhost:49770/>

HTTP Error 403.14 - Forbidden

El servidor web está configurado para no mostrar una lista los contenidos de este directorio.

Causas más probables:

- No se ha configurado un documento predeterminado para la dirección URL solicitada y el examen de directorios no está habilitado en el servidor.

Qué puede intentar:

- Si no desea habilitar el examen de directorios, asegúrese de que se configura un documento predeterminado y de que el archivo existe.
- Habilite el examen de directorios.
 - Vaya al directorio de instalación de IIS Express.
 - Ejecute `appcmd set config /section:system.webServer/directoryBrowse/enabled:true` para habilitar el examen de directorios en el servidor.
 - Ejecute `appcmd set config ["NOMBRE_SITIO"] /section:system.webServer/directoryBrowse/enabled:true` para habilitar el examen de directorios en el sitio.
- Compruebe que el atributo `configuration/system.webServer/directoryBrowse/enabled` se establece como verdadero en el archivo de configuración del sitio o de la aplicación.

Detailed Error Information:

Module	DirectoryListingModule	Requested URL	http://localhost:49770/
Notification	ExecuteRequestHandler	Physical Path	C:\Users\juzo\Desktop\ARMENIA ADS\API EN C#Ejemplo APIEjemplo API2Ejemplo API2Ejemplo API2
Handler	StaticFile	Login Method	Anónimo
Error Code	0x00000000	Login User	Anónimo

More Information:

Este error se produce cuando un documento no está especificado en la dirección URL, no se ha especificado un documento predeterminado para la aplicación o el sitio web y no se ha habilitado la lista de directorios para la aplicación o el sitio web. En la configuración puede ser desactivada a propósito para proteger el contenido del servidor.

[View more information...](#)



Utilizaremos ahora una herramienta que nos permitirá hacer pruebas de las solicitudes de la API.



POSTMAN

Instructor: Adalberto Cárcamo A.
Instructor: Adalberto Cárcamo A.



GET https://jsonplaceholder.typicode.com/users

1 GET 2 https://jsonplaceholder.typicode.com/users 3 Send

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies (1) Headers (19) Test Results

4 Status: 200 OK Time: 784 ms Size: 6.27 KB

Pretty Raw Preview JSON 5

```
1 [
2   {
3     "id": 1,
4     "name": "Leanne Graham",
5     "username": "Bret",
6     "email": "Sincere@april.biz",
7     "address": {
8       "street": "Kulas Light",
9       "suite": "Apt. 556",
10      "city": "Gwenborough",
11      "zipcode": "92998-3874",
12      "geo": {
13        "lat": "-37.3159",
14        "lng": "81.1496"
15      }
16    },
17    "phone": "1-770-736-8031 x56442",
18    "website": "hildegard.org",
19    "company": {
20      "name": "Romaguera-Crona",
21      "catchPhrase": "Multi-layered client server neural-net",
22      "bs": "harness real-time e-markets"
23    }
24  },
25  {
26    "id": 2,
27    "name": "Ervin Howell",
28    "username": "Antonette",
29    "email": "Shanna@melissa.biz",
30    "address": {
31      "street": "Victor Plains",
32      "suite": "Suite 340",
33      "city": "Mediaset",
34      "zipcode": "92998-6593",
35      "geo": {
36        "lat": "41.7022",
37        "lng": "71.3371"
38      }
39    },
40    "phone": "1-704-285-9791 x6470",
41    "website": "elias.io",
42    "company": {
43      "name": "Corkery-Kraemer",
44      "catchPhrase": "Bespoke multi-channel initiative",
45      "bs": "e-enable e-enable e-enable"
46    }
47  },
48  {
49    "id": 3,
50    "name": "Clementine Bauch",
51    "username": "Samantha",
52    "email": "Nathan@yesenia.net",
53    "address": {
54      "street": "Douglas Extension",
55      "suite": "Apt. 9B",
56      "city": "Mediaset",
57      "zipcode": "92998-9132",
58      "geo": {
59        "lat": "48.8469",
60        "lng": "12.1372"
61      }
62    },
63    "phone": "1-646-564-2594 x3198",
64    "website": "kruke.com",
65    "company": {
66      "name": "Kuhikowske-Crona",
67      "catchPhrase": "Seamless multi-tasking",
68      "bs": "e-enable e-enable e-enable"
69    }
70  },
71  {
72    "id": 4,
73    "name": "Patricia Lebsack",
74    "username": "Karianne",
75    "email": "Julianne.OConor@kruke.com",
76    "address": {
77      "street": "Hoeger Mall",
78      "suite": "Apt. 692",
79      "city": "Port Jonathon",
80      "zipcode": "92998-8492",
81      "geo": {
82        "lat": "24.6460",
83        "lng": "-16.9659"
84      }
85    },
86    "phone": "1-994-395-1000 x3204",
87    "website": "conroy.com",
88    "company": {
89      "name": "Corkery-Paige",
90      "catchPhrase": "Profoundly virtual human centric",
91      "bs": "harness real-time e-markets"
92    }
93  },
94  {
95    "id": 5,
96    "name": "Cheryl Parker",
97    "username": "Bryan",
98    "email": "Bryan@marvin.com",
99    "address": {
100     "street": "Carter Street",
101     "suite": "Suite 569",
102     "city": "Mediaset",
103     "zipcode": "92998-6593",
104     "geo": {
105       "lat": "41.7022",
106       "lng": "71.3371"
107     }
108   },
109   "phone": "1-994-395-1000 x3204",
110   "website": "conroy.com",
111   "company": {
112     "name": "Corkery-Paige",
113     "catchPhrase": "Profoundly virtual human centric",
114     "bs": "harness real-time e-markets"
115   }
116 },
117 {
118   "id": 6,
119   "name": "Bret Davis",
120   "username": "Kurtis",
121   "email": "Kurtis@alyvia.biz",
122   "address": {
123     "street": "Nelson Plaza",
124     "suite": "Suite 569",
125     "city": "Mediaset",
126     "zipcode": "92998-6593",
127     "geo": {
128       "lat": "41.7022",
129       "lng": "71.3371"
130     }
131   },
132   "phone": "1-994-395-1000 x3204",
133   "website": "conroy.com",
134   "company": {
135     "name": "Corkery-Paige",
136     "catchPhrase": "Profoundly virtual human centric",
137     "bs": "harness real-time e-markets"
138   }
139 },
140 {
141   "id": 7,
142   "name": "Kurtis Davis",
143   "username": "Kurtis",
144   "email": "Kurtis@alyvia.biz",
145   "address": {
146     "street": "Nelson Plaza",
147     "suite": "Suite 569",
148     "city": "Mediaset",
149     "zipcode": "92998-6593",
150     "geo": {
151       "lat": "41.7022",
152       "lng": "71.3371"
153     }
154   },
155   "phone": "1-994-395-1000 x3204",
156   "website": "conroy.com",
157   "company": {
158     "name": "Corkery-Paige",
159     "catchPhrase": "Profoundly virtual human centric",
160     "bs": "harness real-time e-markets"
161   }
162 },
163 {
164   "id": 8,
165   "name": "Kurtis Davis",
166   "username": "Kurtis",
167   "email": "Kurtis@alyvia.biz",
168   "address": {
169     "street": "Nelson Plaza",
170     "suite": "Suite 569",
171     "city": "Mediaset",
172     "zipcode": "92998-6593",
173     "geo": {
174       "lat": "41.7022",
175       "lng": "71.3371"
176     }
177   },
178   "phone": "1-994-395-1000 x3204",
179   "website": "conroy.com",
180   "company": {
181     "name": "Corkery-Paige",
182     "catchPhrase": "Profoundly virtual human centric",
183     "bs": "harness real-time e-markets"
184   }
185 },
186 {
187   "id": 9,
188   "name": "Kurtis Davis",
189   "username": "Kurtis",
190   "email": "Kurtis@alyvia.biz",
191   "address": {
192     "street": "Nelson Plaza",
193     "suite": "Suite 569",
194     "city": "Mediaset",
195     "zipcode": "92998-6593",
196     "geo": {
197       "lat": "41.7022",
198       "lng": "71.3371"
199     }
200   },
201   "phone": "1-994-395-1000 x3204",
202   "website": "conroy.com",
203   "company": {
204     "name": "Corkery-Paige",
205     "catchPhrase": "Profoundly virtual human centric",
206     "bs": "harness real-time e-markets"
207   }
208 },
209 {
210   "id": 10,
211   "name": "Kurtis Davis",
212   "username": "Kurtis",
213   "email": "Kurtis@alyvia.biz",
214   "address": {
215     "street": "Nelson Plaza",
216     "suite": "Suite 569",
217     "city": "Mediaset",
218     "zipcode": "92998-6593",
219     "geo": {
220       "lat": "41.7022",
221       "lng": "71.3371"
222     }
223   },
224   "phone": "1-994-395-1000 x3204",
225   "website": "conroy.com",
226   "company": {
227     "name": "Corkery-Paige",
228     "catchPhrase": "Profoundly virtual human centric",
229     "bs": "harness real-time e-markets"
230   }
231 },
232 {
233   "id": 11,
234   "name": "Kurtis Davis",
235   "username": "Kurtis",
236   "email": "Kurtis@alyvia.biz",
237   "address": {
238     "street": "Nelson Plaza",
239     "suite": "Suite 569",
240     "city": "Mediaset",
241     "zipcode": "92998-6593",
242     "geo": {
243       "lat": "41.7022",
244       "lng": "71.3371"
245     }
246   },
247   "phone": "1-994-395-1000 x3204",
248   "website": "conroy.com",
249   "company": {
250     "name": "Corkery-Paige",
251     "catchPhrase": "Profoundly virtual human centric",
252     "bs": "harness real-time e-markets"
253   }
254 },
255 {
256   "id": 12,
257   "name": "Kurtis Davis",
258   "username": "Kurtis",
259   "email": "Kurtis@alyvia.biz",
260   "address": {
261     "street": "Nelson Plaza",
262     "suite": "Suite 569",
263     "city": "Mediaset",
264     "zipcode": "92998-6593",
265     "geo": {
266       "lat": "41.7022",
267       "lng": "71.3371"
268     }
269   },
270   "phone": "1-994-395-1000 x3204",
271   "website": "conroy.com",
272   "company": {
273     "name": "Corkery-Paige",
274     "catchPhrase": "Profoundly virtual human centric",
275     "bs": "harness real-time e-markets"
276   }
277 },
278 {
279   "id": 13,
280   "name": "Kurtis Davis",
281   "username": "Kurtis",
282   "email": "Kurtis@alyvia.biz",
283   "address": {
284     "street": "Nelson Plaza",
285     "suite": "Suite 569",
286     "city": "Mediaset",
287     "zipcode": "92998-6593",
288     "geo": {
289       "lat": "41.7022",
290       "lng": "71.3371"
291     }
292   },
293   "phone": "1-994-395-1000 x3204",
294   "website": "conroy.com",
295   "company": {
296     "name": "Corkery-Paige",
297     "catchPhrase": "Profoundly virtual human centric",
298     "bs": "harness real-time e-markets"
299   }
300 },
301 {
302   "id": 14,
303   "name": "Kurtis Davis",
304   "username": "Kurtis",
305   "email": "Kurtis@alyvia.biz",
306   "address": {
307     "street": "Nelson Plaza",
308     "suite": "Suite 569",
309     "city": "Mediaset",
310     "zipcode": "92998-6593",
311     "geo": {
312       "lat": "41.7022",
313       "lng": "71.3371"
314     }
315   },
316   "phone": "1-994-395-1000 x3204",
317   "website": "conroy.com",
318   "company": {
319     "name": "Corkery-Paige",
320     "catchPhrase": "Profoundly virtual human centric",
321     "bs": "harness real-time e-markets"
322   }
323 },
324 {
325   "id": 15,
326   "name": "Kurtis Davis",
327   "username": "Kurtis",
328   "email": "Kurtis@alyvia.biz",
329   "address": {
330     "street": "Nelson Plaza",
331     "suite": "Suite 569",
332     "city": "Mediaset",
333     "zipcode": "92998-6593",
334     "geo": {
335       "lat": "41.7022",
336       "lng": "71.3371"
337     }
338   },
339   "phone": "1-994-395-1000 x3204",
340   "website": "conroy.com",
341   "company": {
342     "name": "Corkery-Paige",
343     "catchPhrase": "Profoundly virtual human centric",
344     "bs": "harness real-time e-markets"
345   }
346 },
347 {
348   "id": 16,
349   "name": "Kurtis Davis",
350   "username": "Kurtis",
351   "email": "Kurtis@alyvia.biz",
352   "address": {
353     "street": "Nelson Plaza",
354     "suite": "Suite 569",
355     "city": "Mediaset",
356     "zipcode": "92998-6593",
357     "geo": {
358       "lat": "41.7022",
359       "lng": "71.3371"
360     }
361   },
362   "phone": "1-994-395-1000 x3204",
363   "website": "conroy.com",
364   "company": {
365     "name": "Corkery-Paige",
366     "catchPhrase": "Profoundly virtual human centric",
367     "bs": "harness real-time e-markets"
368   }
369 },
370 {
371   "id": 17,
372   "name": "Kurtis Davis",
373   "username": "Kurtis",
374   "email": "Kurtis@alyvia.biz",
375   "address": {
376     "street": "Nelson Plaza",
377     "suite": "Suite 569",
378     "city": "Mediaset",
379     "zipcode": "92998-6593",
380     "geo": {
381       "lat": "41.7022",
382       "lng": "71.3371"
383     }
384   },
385   "phone": "1-994-395-1000 x3204",
386   "website": "conroy.com",
387   "company": {
388     "name": "Corkery-Paige",
389     "catchPhrase": "Profoundly virtual human centric",
390     "bs": "harness real-time e-markets"
391   }
392 },
393 {
394   "id": 18,
395   "name": "Kurtis Davis",
396   "username": "Kurtis",
397   "email": "Kurtis@alyvia.biz",
398   "address": {
399     "street": "Nelson Plaza",
400     "suite": "Suite 569",
401     "city": "Mediaset",
402     "zipcode": "92998-6593",
403     "geo": {
404       "lat": "41.7022",
405       "lng": "71.3371"
406     }
407   },
408   "phone": "1-994-395-1000 x3204",
409   "website": "conroy.com",
410   "company": {
411     "name": "Corkery-Paige",
412     "catchPhrase": "Profoundly virtual human centric",
413     "bs": "harness real-time e-markets"
414   }
415 },
416 {
417   "id": 19,
418   "name": "Kurtis Davis",
419   "username": "Kurtis",
420   "email": "Kurtis@alyvia.biz",
421   "address": {
422     "street": "Nelson Plaza",
423     "suite": "Suite 569",
424     "city": "Mediaset",
425     "zipcode": "92998-6593",
426     "geo": {
427       "lat": "41.7022",
428       "lng": "71.3371"
429     }
430   },
431   "phone": "1-994-395-1000 x3204",
432   "website": "conroy.com",
433   "company": {
434     "name": "Corkery-Paige",
435     "catchPhrase": "Profoundly virtual human centric",
436     "bs": "harness real-time e-markets"
437   }
438 },
439 {
440   "id": 20,
441   "name": "Kurtis Davis",
442   "username": "Kurtis",
443   "email": "Kurtis@alyvia.biz",
444   "address": {
445     "street": "Nelson Plaza",
446     "suite": "Suite 569",
447     "city": "Mediaset",
448     "zipcode": "92998-6593",
449     "geo": {
450       "lat": "41.7022",
451       "lng": "71.3371"
452     }
453   },
454   "phone": "1-994-395-1000 x3204",
455   "website": "conroy.com",
456   "company": {
457     "name": "Corkery-Paige",
458     "catchPhrase": "Profoundly virtual human centric",
459     "bs": "harness real-time e-markets"
460   }
461 },
462 {
463   "id": 21,
464   "name": "Kurtis Davis",
465   "username": "Kurtis",
466   "email": "Kurtis@alyvia.biz",
467   "address": {
468     "street": "Nelson Plaza",
469     "suite": "Suite 569",
470     "city": "Mediaset",
471     "zipcode": "92998-6593",
472     "geo": {
473       "lat": "41.7022",
474       "lng": "71.3371"
475     }
476   },
477   "phone": "1-994-395-1000 x3204",
478   "website": "conroy.com",
479   "company": {
480     "name": "Corkery-Paige",
481     "catchPhrase": "Profoundly virtual human centric",
482     "bs": "harness real-time e-markets"
483   }
484 },
485 {
486   "id": 22,
487   "name": "Kurtis Davis",
488   "username": "Kurtis",
489   "email": "Kurtis@alyvia.biz",
490   "address": {
491     "street": "Nelson Plaza",
492     "suite": "Suite 569",
493     "city": "Mediaset",
494     "zipcode": "92998-6593",
495     "geo": {
496       "lat": "41.7022",
497       "lng": "71.3371"
498     }
499   },
500   "phone": "1-994-395-1000 x3204",
501   "website": "conroy.com",
502   "company": {
503     "name": "Corkery-Paige",
504     "catchPhrase": "Profoundly virtual human centric",
505     "bs": "harness real-time e-markets"
506   }
507 },
508 {
509   "id": 23,
510   "name": "Kurtis Davis",
511   "username": "Kurtis",
512   "email": "Kurtis@alyvia.biz",
513   "address": {
514     "street": "Nelson Plaza",
515     "suite": "Suite 569",
516     "city": "Mediaset",
517     "zipcode": "92998-6593",
518     "geo": {
519       "lat": "41.7022",
520       "lng": "71.3371"
521     }
522   },
523   "phone": "1-994-395-1000 x3204",
524   "website": "conroy.com",
525   "company": {
526     "name": "Corkery-Paige",
527     "catchPhrase": "Profoundly virtual human centric",
528     "bs": "harness real-time e-markets"
529   }
530 },
531 {
532   "id": 24,
533   "name": "Kurtis Davis",
534   "username": "Kurtis",
535   "email": "Kurtis@alyvia.biz",
536   "address": {
537     "street": "Nelson Plaza",
538     "suite": "Suite 569",
539     "city": "Mediaset",
540     "zipcode": "92998-6593",
541     "geo": {
542       "lat": "41.7022",
543       "lng": "71.3371"
544     }
545   },
546   "phone": "1-994-395-1000 x3204",
547   "website": "conroy.com",
548   "company": {
549     "name": "Corkery-Paige",
550     "catchPhrase": "Profoundly virtual human centric",
551     "bs": "harness real-time e-markets"
552   }
553 },
554 {
555   "id": 25,
556   "name": "Kurtis Davis",
557   "username": "Kurtis",
558   "email": "Kurtis@alyvia.biz",
559   "address": {
560     "street": "Nelson Plaza",
561     "suite": "Suite 569",
562     "city": "Mediaset",
563     "zipcode": "92998-6593",
564     "geo": {
565       "lat": "41.7022",
566       "lng": "71.3371"
567     }
568   },
569   "phone": "1-994-395-1000 x3204",
570   "website": "conroy.com",
571   "company": {
572     "name": "Corkery-Paige",
573     "catchPhrase": "Profoundly virtual human centric",
574     "bs": "harness real-time e-markets"
575   }
576 },
577 {
578   "id": 26,
579   "name": "Kurtis Davis",
580   "username": "Kurtis",
581   "email": "Kurtis@alyvia.biz",
582   "address": {
583     "street": "Nelson Plaza",
584     "suite": "Suite 569",
585     "city": "Mediaset",
586     "zipcode": "92998-6593",
587     "geo": {
588       "lat": "41.7022",
589       "lng": "71.3371"
590     }
591   },
592   "phone": "1-994-395-1000 x3204",
593   "website": "conroy.com",
594   "company": {
595     "name": "Corkery-Paige",
596     "catchPhrase": "Profoundly virtual human centric",
597     "bs": "harness real-time e-markets"
598   }
599 },
600 {
601   "id": 27,
602   "name": "Kurtis Davis",
603   "username": "Kurtis",
604   "email": "Kurtis@alyvia.biz",
605   "address": {
606     "street": "Nelson Plaza",
607     "suite": "Suite 569",
608     "city": "Mediaset",
609     "zipcode": "92998-6593",
610     "geo": {
611       "lat": "41.7022",
612       "lng": "71.3371"
613     }
614   },
615   "phone": "1-994-395-1000 x3204",
616   "website": "conroy.com",
617   "company": {
618     "name": "Corkery-Paige",
619     "catchPhrase": "Profoundly virtual human centric",
620     "bs": "harness real-time e-markets"
621   }
622 },
623 {
624   "id": 28,
625   "name": "Kurtis Davis",
626   "username": "Kurtis",
627   "email": "Kurtis@alyvia.biz",
628   "address": {
629     "street": "Nelson Plaza",
630     "suite": "Suite 569",
631     "city": "Mediaset",
632     "zipcode": "92998-6593",
633     "geo": {
634       "lat": "41.7022",
635       "lng": "71.3371"
636     }
637   },
638   "phone": "1-994-395-1000 x3204",
639   "website": "conroy.com",
640   "company": {
641     "name": "Corkery-Paige",
642     "catchPhrase": "Profoundly virtual human centric",
643     "bs": "harness real-time e-markets"
644   }
645 },
646 {
647   "id": 29,
648   "name": "Kurtis Davis",
649   "username": "Kurtis",
650   "email": "Kurtis@alyvia.biz",
651   "address": {
652     "street": "Nelson Plaza",
653     "suite": "Suite 569",
654     "city": "Mediaset",
655     "zipcode": "92998-6593",
656     "geo": {
657       "lat": "41.7022",
658       "lng": "71.3371"
659     }
660   },
661   "phone": "1-994-395-1000 x3204",
662   "website": "conroy.com",
663   "company": {
664     "name": "Corkery-Paige",
665     "catchPhrase": "Profoundly virtual human centric",
666     "bs": "harness real-time e-markets"
667   }
668 },
669 {
670   "id": 30,
671   "name": "Kurtis Davis",
672   "username": "Kurtis",
673   "email": "Kurtis@alyvia.biz",
674   "address": {
675     "street": "Nelson Plaza",
676     "suite": "Suite 569",
677     "city": "Mediaset",
678     "zipcode": "92998-6593",
679     "geo": {
680       "lat": "41.7022",
681       "lng": "71.3371"
682     }
683   },
684   "phone": "1-994-395-1000 x3204",
685   "website": "conroy.com",
686   "company": {
687     "name": "Corkery-Paige",
688     "catchPhrase": "Profoundly virtual human centric",
689     "bs": "harness real-time e-markets"
690   }
691 },
692 {
693   "id": 31,
694   "name": "Kurtis Davis",
695   "username": "Kurtis",
696   "email": "Kurtis@alyvia.biz",
697   "address": {
698     "street": "Nelson Plaza",
699     "suite": "Suite 569",
700     "city": "Mediaset",
701     "zipcode": "92998-6593",
702     "geo": {
703       "lat": "41.7022",
704       "lng": "71.3371"
705     }
706   },
707   "phone": "1-994-395-1000 x3204",
708   "website": "conroy.com",
709   "company": {
710     "name": "Corkery-Paige",
711     "catchPhrase": "Profoundly virtual human centric",
712     "bs": "harness real-time e-markets"
713   }
714 },
715 {
716   "id": 32,
717   "name": "Kurtis Davis",
718   "username": "Kurtis",
719   "email": "Kurtis@alyvia.biz",
720   "address": {
721     "street": "Nelson Plaza",
722     "suite": "Suite 569",
723     "city": "Mediaset",
724     "zipcode": "92998-6593",
725     "geo": {
726       "lat": "41.7022",
727       "lng": "71.3371"
728     }
729   },
730   "phone": "1-994-395-1000 x3204",
731   "website": "conroy.com",
732   "company": {
733     "name": "Corkery-Paige",
734     "catchPhrase": "Profoundly virtual human centric",
735     "bs": "harness real-time e-markets"
736   }
737 },
738 {
739   "id": 33,
740   "name": "Kurtis Davis",
741   "username": "Kurtis",
742   "email": "Kurtis@alyvia.biz",
743   "address": {
744     "street": "Nelson Plaza",
745     "suite": "Suite 569",
746     "city": "Mediaset",
747     "zipcode": "92998-6593",
748     "geo": {
749       "lat": "41.7022",
750       "lng": "71.3371"
751     }
752   },
753   "phone": "1-994-395-1000 x3204",
754   "website": "conroy.com",
755   "company": {
756     "name": "Corkery-Paige",
757     "catchPhrase": "Profoundly virtual human centric",
758     "bs": "harness real-time e-markets"
759   }
760 },
761 {
762   "id": 34,
763   "name": "Kurtis Davis",
764   "username": "Kurtis",
765   "email": "Kurtis@alyvia.biz",
766   "address": {
767     "street": "Nelson Plaza",
768     "suite": "Suite 569",
769     "city": "Mediaset",
770     "zipcode": "92998-6593",
771     "geo": {
772       "lat": "41.7022",
773       "lng": "71.3371"
774     }
775   },
776   "phone": "1-994-395-1000 x3204",
777   "website": "conroy.com",
778   "company": {
779     "name": "Corkery-Paige",
780     "catchPhrase": "Profoundly virtual human centric",
781     "bs": "harness real-time e-markets"
782   }
783 },
784 {
785   "id": 35,
786   "name": "Kurtis Davis",
787   "username": "Kurtis",
788   "email": "Kurtis@alyvia.biz",
789   "address": {
790     "street": "Nelson Plaza",
791     "suite": "Suite 569",
792     "city": "Mediaset",
793     "zipcode": "92998-6593",
794     "geo": {
795       "lat": "41.7022",
796       "lng": "71.3371"
797     }
798   },
799   "phone": "1-994-395-1000 x3204",
800   "website": "conroy.com",
801   "company": {
802     "name": "Corkery-Paige",
803     "catchPhrase": "Profoundly virtual human centric",
804     "bs": "harness real-time e-markets"
805   }
806 },
807 {
808   "id": 36,
809   "name": "Kurtis Davis",
810   "username": "Kurtis",
811   "email": "Kurtis@alyvia.biz",
812   "address": {
813     "street": "Nelson Plaza",
814     "suite": "Suite 569",
815     "city": "Mediaset",
816     "zipcode": "92998-6593",
817     "geo": {
818       "lat": "41.7022",
819       "lng": "71.3371"
820     }
821   },
822   "phone": "1-994-395-1000 x3204",
823   "website": "conroy.com",
824   "company": {
825     "name": "Corkery-Paige",
826     "catchPhrase": "Profoundly virtual human centric",
827     "bs": "harness real-time e-markets"
828   }
829 },
830 {
831   "id": 37,
832   "name": "Kurtis Davis",
833   "username": "Kurtis",
834   "email": "Kurtis@alyvia.biz",
835   "address": {
836     "street": "Nelson Plaza",
837     "suite": "Suite 569",
838     "city": "Mediaset",
839     "zipcode": "92998-6593",
840     "geo": {
841       "lat": "41.7022",
842       "lng": "71.3371"
843     }
844   },
845   "phone": "1-994-395-1000 x3204",
846   "website": "conroy.com",
847   "company": {
848     "name": "Corkery-Paige",
849     "catchPhrase": "Profoundly virtual human centric",
850     "bs": "harness real-time e-markets"
851   }
852 },
853 {
854   "id": 38,
855   "name": "Kurtis Davis",
856   "username": "Kurtis",
857   "email": "Kurtis@alyvia.biz",
858   "address": {
859     "street": "Nelson Plaza",
860     "suite": "Suite 569",
861     "city": "Mediaset",
862     "zipcode": "92998-6593",
863     "geo": {
864       "lat": "41.7022",
865       "lng": "71.3371"
866     }
867   },
868   "phone": "1-994-395-1000 x3204",
869   "website": "conroy.com",
870   "company": {
871     "name": "Corkery-Paige",
872     "catchPhrase": "Profoundly virtual human centric",
873     "bs": "harness real-time e-markets"
874   }
875 },
876 {
877   "id": 39,
878   "name": "Kurtis Davis",
879   "username": "Kurtis",
880   "email": "Kurtis@alyvia.biz",
881   "address": {
882     "street": "Nelson Plaza",
883     "suite": "Suite 569",
884     "city": "Mediaset",
885     "zipcode": "92998-6593",
886     "geo": {
887       "lat": "41.7022",
888       "lng": "71.3371"
889     }
890   },
891   "phone": "1-994-395-1000 x3204",
892   "website": "conroy.com",
893   "company": {
894     "name": "Corkery-Paige",
895     "catchPhrase": "Profoundly virtual human centric",
896     "bs": "harness real-time e-markets"
897   }
898 },
899 {
900   "id": 40,
901   "name": "Kurtis Davis",
902   "username": "Kurtis",
903   "email": "Kurtis@alyvia.biz",
904   "address": {
905     "street": "Nelson Plaza",
906     "suite": "Suite 569",
907     "city": "Mediaset",
908     "zipcode": "92998-6593",
909     "geo": {
910       "lat": "41.7022",
911       "lng": "71.3371"
912     }
913   },
914   "phone": "1-994-395-1000 x3204",
915   "website": "conroy.com",
916   "company": {
917     "name": "Corkery-Paige",
918     "catchPhrase": "Profoundly virtual human centric",
919     "bs": "harness real-time e-markets"
920   }
921 },
922 {
923   "id": 41,
924   "name": "Kurtis Davis",
925   "username": "Kurtis",
926   "email": "Kurtis@alyvia.biz",
927   "address": {
928     "street": "Nelson Plaza",
929     "suite": "Suite 569",
930     "city": "Mediaset",
931     "zipcode": "92998-6593",
932     "geo": {
933       "lat": "41.7022",
934       "lng": "71.3371"
935     }
936   },
937   "phone": "1-994-395-1000 x3204",
938   "website": "conroy.com",
939   "company": {
940     "name": "Corkery-Paige",
941     "catchPhrase": "Pro
```



GRACIAS