

Protocolos de uso común en aplicaciones web

HTTP, FTP y STOP

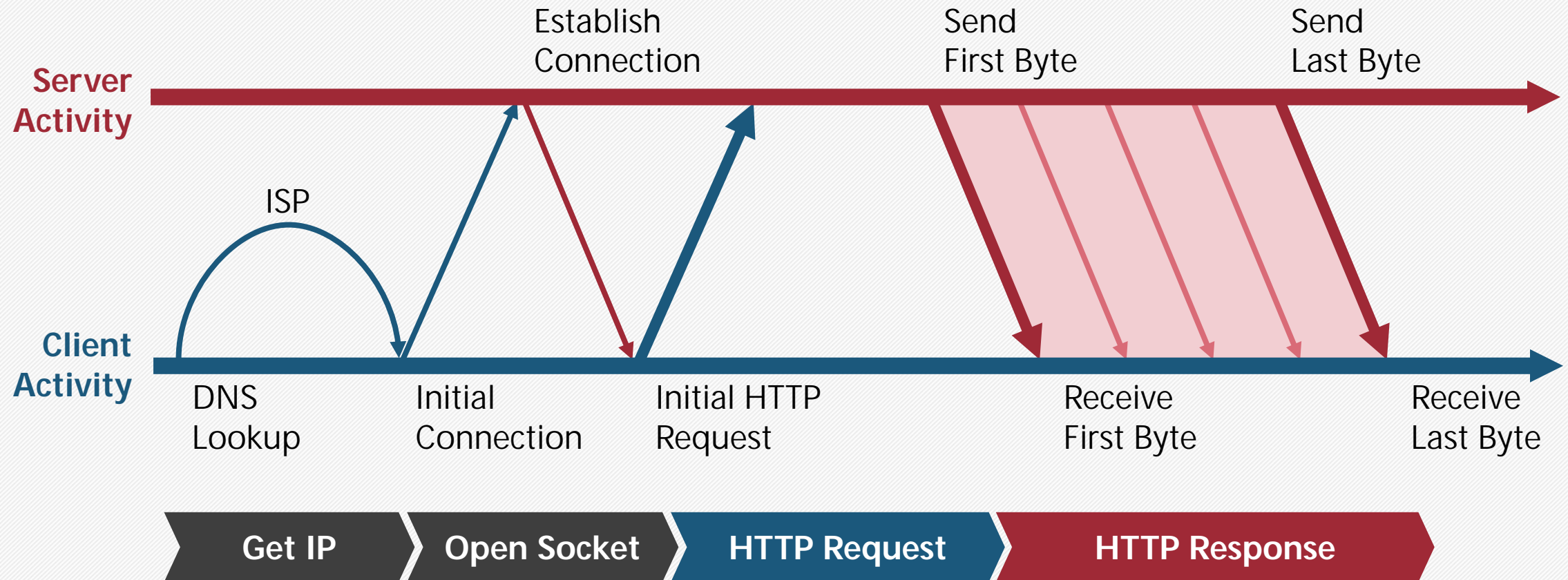


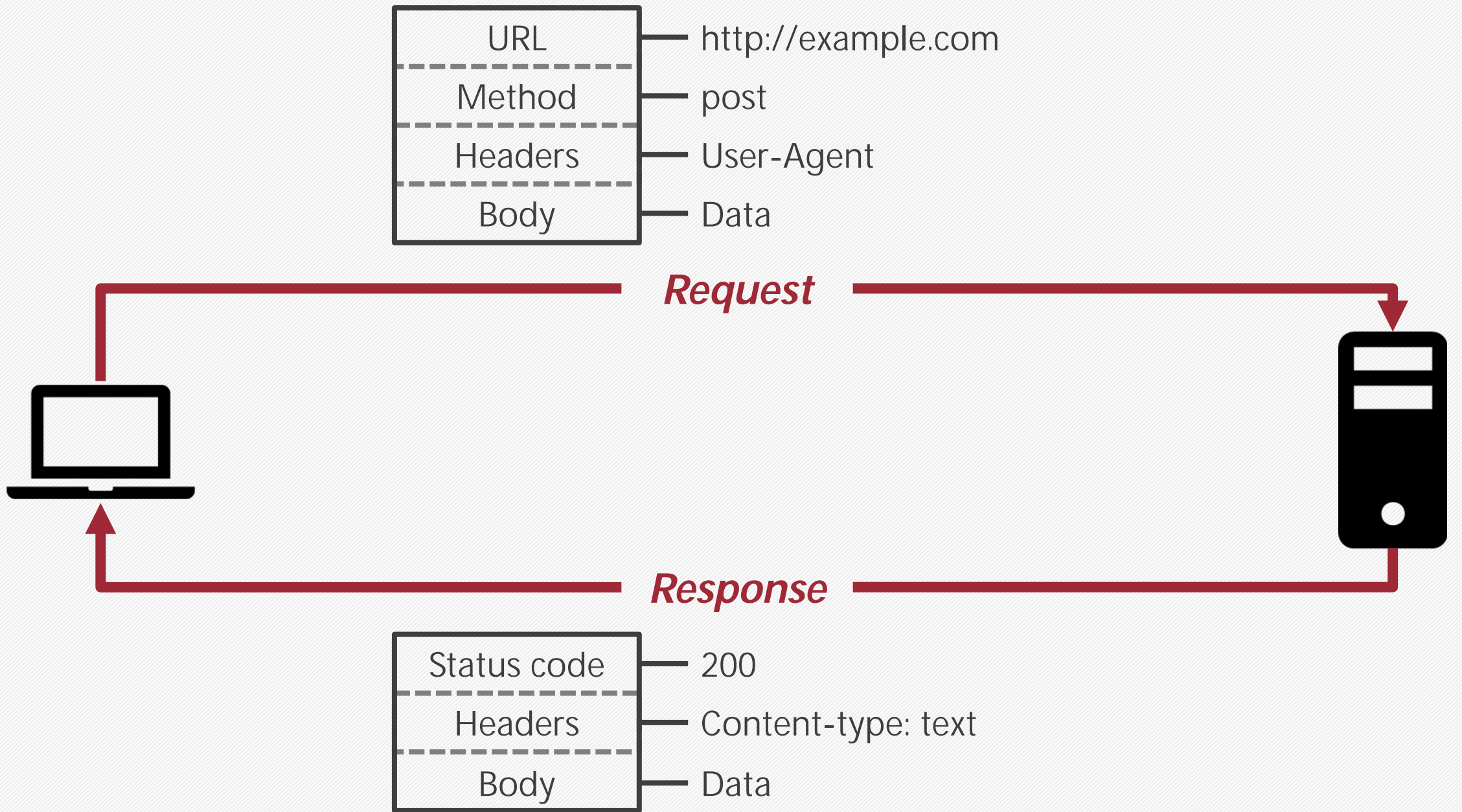
Modelo petición-respuesta HTTP

Model request-response HTTP

Esquema de **comunicación** entre un **cliente** y un **servidor**, el cual efectúa una petición mediante un localizador uniforme de recursos o **URL**

The HTTP Request





Modelo petición-respuesta HTTP

HTTP Request Message

		Method	Path	Protocol
		POST	/Home/index	HTTP/1.1
Message Header	Host: unitecti.net Accept: image/gif, image/jpg, */* Accept-Language: en-us Accept-Encoding: gzip, deflate User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT) Content-Length: length			
Message Body	id=555&content=Hola+Mundo			

Modelo petición-respuesta HTTP

HTTP Response Message

	Protocol	Code	Status
	HTTP/1.1	200	OK
Message Header	Date: Mon, 08 Feb 2018 01:01:01 GMT Server: IIS/8.3.29 (Win64) Last-Modified: Sun, 07 Feb 2018 Accept-Ranges: bytes Content-Length: 35 Connection: close Content-Type: text/html; charset=utf-8		
Message Body	<h1>Hola</h1> <h2>mundo</h2>		

Códigos de respuesta

100 Mensajes

- 100-111 Conexión rechazada

200 Operación exitosa

- 201-203 Información no oficial
- 204 Sin Contenido
- 205 Contenido para recargar
- 206 Contenido parcial

300 Redirección

- 301 Mudado permanentemente
- 302 Encontrado
- 303 Vea otros
- 304 No modificado
- 305 Utilice un proxy
- 307 Redirección temporal

400 Error por parte del cliente (Solicitud incorrecta)

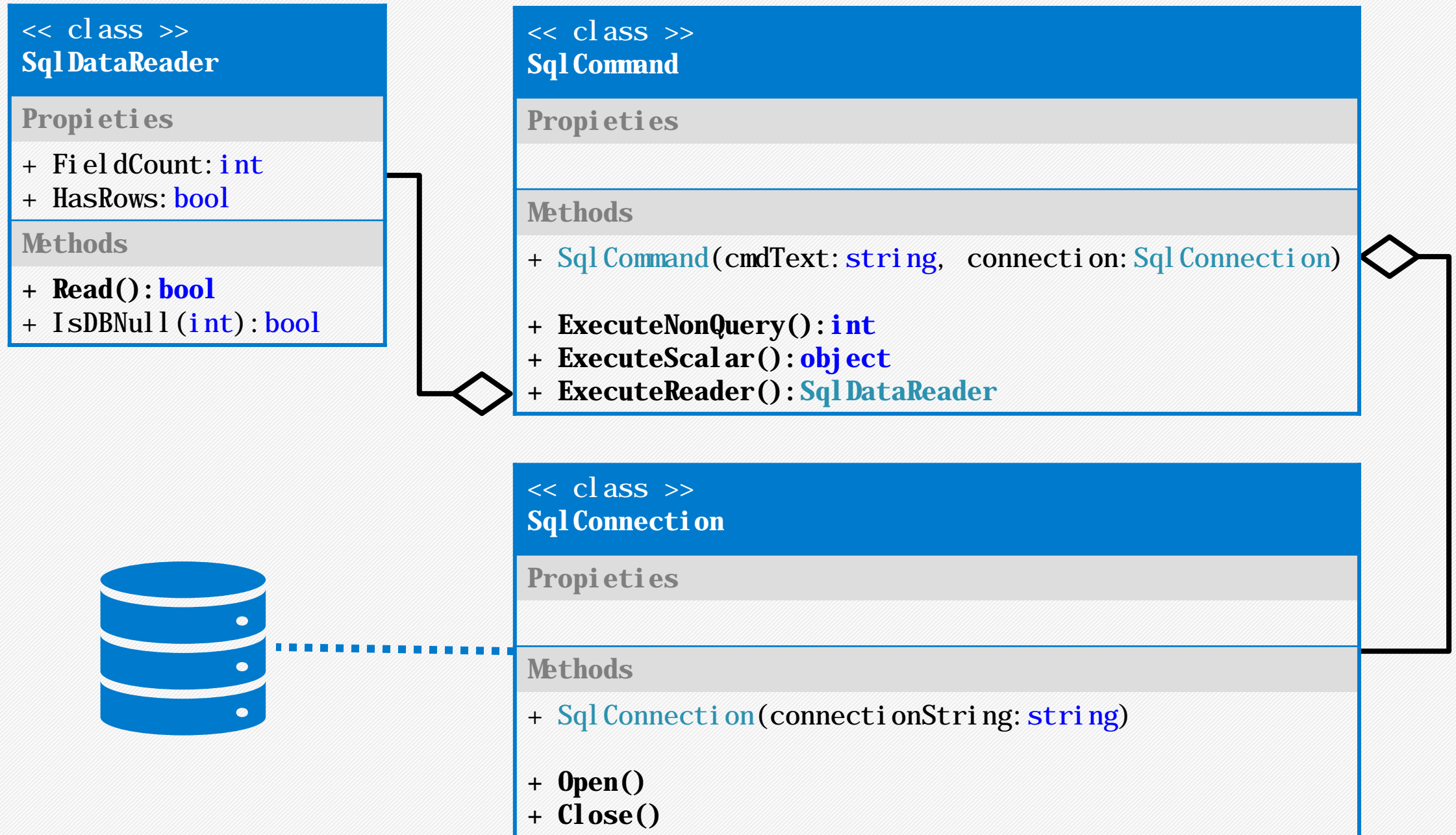
- 401 No autorizado
- 402 Pago requerido
- 403 Prohibido
- 404 No encontrado
- 409 Conflicto
- 410 Ya no disponible
- 412 Falló precondition

500 Error del servidor (Error interno)

- 501 No implementado
- 502 Pasarela incorrecta
- 503 Servicio no disponible
- 504 Tiempo de espera de la pasarela agotado
- 505 Versión de HTTP no soportada

Conexión a base de datos en una aplicación de consola

<https://gist.github.com/jlcarrillog/929747df5b8dff8e41abfec44e0e988e>



Connection

```
using System;  
using System.Data.SqlClient;
```

```
class Program
```

```
{
```

```
    static void Main(string[] args)
```

```
    {
```

```
        string _connectionString = "Data Source=... ";
```

```
        var con = new SqlConnection(_connectionString);
```

```
        con.Open();
```

```
        con.Close();
```

```
    }
```

```
}
```

```
Data Source=unitec-db.database.windows.net;  
Initial Catalog=im;  
User ID=AdminDB;  
Password=****;
```



Id	Nombre	Edad
e0267f65-b7e5-4942-9125-ace950f1363	Pepe Pepe	15
4f1dfcfa-3900-4b29-ab82-2cb462e4a902	Juan Juan	24
96f4ac67-ac12-4bfd-8262-040996603caf	Luis Luis	43

Queries: **Create**, **Alter**, **Drop**, **Insert**, **Update**, **Delete**, ...

Connection

```
using System;
using System.Data.SqlClient;

class Program
{
    static void Main(string[] args)
    {
        string _connectionString = "Data Source=... ";

        var con = new SqlConnection(_connectionString);

        try
        {
            con.Open();
        }
        catch (Exception)
        {
            throw;
        }
        finally
        {
            con.Close();
        }
    }
}
```

Data Source=unitec-db.database.windows.net;
Initial Catalog=im;
User ID=AdminDB;
Password=****;



Id	Nombre	Edad
e0267f65-b7e5-4942-9125-ace950f1363	Pepe Pepe	15
4f1dfcfa-3900-4b29-ab82-2cb462e4a902	Juan Juan	24
96f4ac67-ac12-4bfd-8262-040996603caf	Luis Luis	43

Queries: **Create, Alter, Drop, Insert, Update, Delete, ...**

```

using System;

namespace IM03a1
{
    class Program
    {
        static void Main(string[] args)
        {
            string _connectionString = "Data Source=... ";
            var _context = new EmpleadoDbContext(_connectionString);

            var list = _context.List();
            foreach (var e in list)
            {
                Console.WriteLine($"Id: {e.Id}, No: {e.Nombre}, Ed: {e.Edad}");
            }

            // ToDo
        }
    }
}

```

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;

public class EmpleadoDbContext
{
    private readonly string _connectionString;
    public EmpleadoDbContext(string connectionString)
    {
        _connectionString = connectionString;
    }

    public List<Empleado> List()
    {
        var data = new List<Empleado>();
        return data;
    }

    public Empleado Details(Guid id)
    {
        var data = new Empleado();
        return data;
    }

    public void Create(Empleado data)
    {
    }

    public void Edit(Empleado data)
    {
    }

    public void Delete(Guid id)
    {
    }
}

```

Base de Datos

Ejercicio

Producto (dbo)			
	Column Name	Data Type	Allow Nulls
⚡	Id	uniqueidentifier	<input type="checkbox"/>
	Nombre	nvarchar(128)	<input type="checkbox"/>
	Descripcion	nvarchar(512)	<input type="checkbox"/>
	Tipo	smallint	<input type="checkbox"/>
	Precio	float	<input type="checkbox"/>
	Cantidad	int	<input type="checkbox"/>
	Foto	varchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

```
CREATE TABLE [Producto](  
    [Id] [uniqueidentifier] PRIMARY KEY NOT NULL,  
    [Nombre] [nvarchar](128) NOT NULL,  
    [Descripcion] [nvarchar](512) NOT NULL,  
    [Tipo] [smallint] NOT NULL,  
    [Precio] [float] NOT NULL,  
    [Cantidad] [int] NOT NULL,  
    [Foto] [varchar](max) NULL  
);
```


TÓPICO PARA PROYECTO

Programa de Fomento a la
Investigación

BASES

EQUIPOS:

- Equipos de máximo 4 personas.
- Registro de Proyecto.
- Nombramiento de Líder de Proyecto.
- No está permitido proyectos individuales.
- Línea de investigación.
 - Sistemas computacionales para aplicaciones específicas.

CRITERIOS:

- Evaluación.
 - 10% avance al bimestre.
 - 10% Proyecto terminado.
- Funcionalidad.
- Presentación.
- Tecnologías de desarrollo implementadas.
- Documento final conforme a Rúbrica.

PRESENTACIÓN

PRESENTACIÓN GRUPO:

- Envío de la documentación antes de la presentación, incluir manuales operativos y de usuario.
- Presentación ejecutiva de los proyectos desarrollados en campus Semana 12.
- El mejor Proyecto seleccionado representará al campus en la presentación de proyectos de ZM.
- Todos los miembros del equipo deberán participar en la presentación del Proyecto.

PRESENTACIÓN A D.A:

- Presentación de mejores proyectos de los campus semana 13.
- Evaluadores profesores invitados.

ERP para la Gestión Hospitalaria:

- Desarrollar un sistema de gestión hospitalaria con soluciones en plataformas distribuidas y de alta disponibilidad, considerando metodologías para el desarrollo e implantación de software, seguridad, costo-beneficio y su valor agregado en la industria hospitalaria y que contribuya en su estrategia de negocio.

Sprint 1: Seguridad (1s)

Sprint 2: Productos (3s)

Sprint 3: Pacientes (3s)

Sprint 4: Citas (3s)

Sprint 5: Reportes (2s)

Sprint 6: Doctores (3s)

Sprint 7: Pagos (4s)



Check list Cumplimiento Evidencias

1. Generales

- 1.1 ¿Se encuentran Identificados los Procesos en el Sistema?
- 1.2 ¿Se controlan los Procesos?
- 1.3 ¿Se cuenta con un Modelado de los Procesos?
- 1.4 ¿Se cuenta con un Modelado de **Requerimientos**?
- 1.5 ¿Se cuenta con un Modelado de **Diseño**?
- 1.6 ¿Se contó con Estrategias de **Prueba** de Software?
- 1.7 ¿Se cuenta con una **Administración de la Configuración del Software**?
- 1.8 ¿Se cuenta con una **Estimación Económica** en el Desarrollo del Software?
- 1.9 ¿Se administran los **Riesgos** del Sistema?
- 1.10. ¿Pueden realizarse **Actualizaciones** y Mejoras de forma **ágil**?

Cumple **No Cumple**

2. Requisitos de Documentación

- 2.1 ¿Existen Procedimientos Documentados?
- 2.2 ¿El Manual incluye el Alcance del Sistema?
- 2.3 ¿El Manual incluye Exclusiones del Sistema?
- 2.4 ¿El Manual incluye la Interacción de los Procesos?
- 2.5 ¿Existe una Metodología Documentada para cada Proceso?

Notas

- 1.1 Se refiere a todos los procesos que involucra el Sistema, Gestión Activa, Gestión de Ingresos, Gestión Pacientes y Médica, etc.(los 5 procesos).
- 1.2 Se refiere al acceso del sistema y a la Interacción correcta de los datos entre procesos.

Documentación

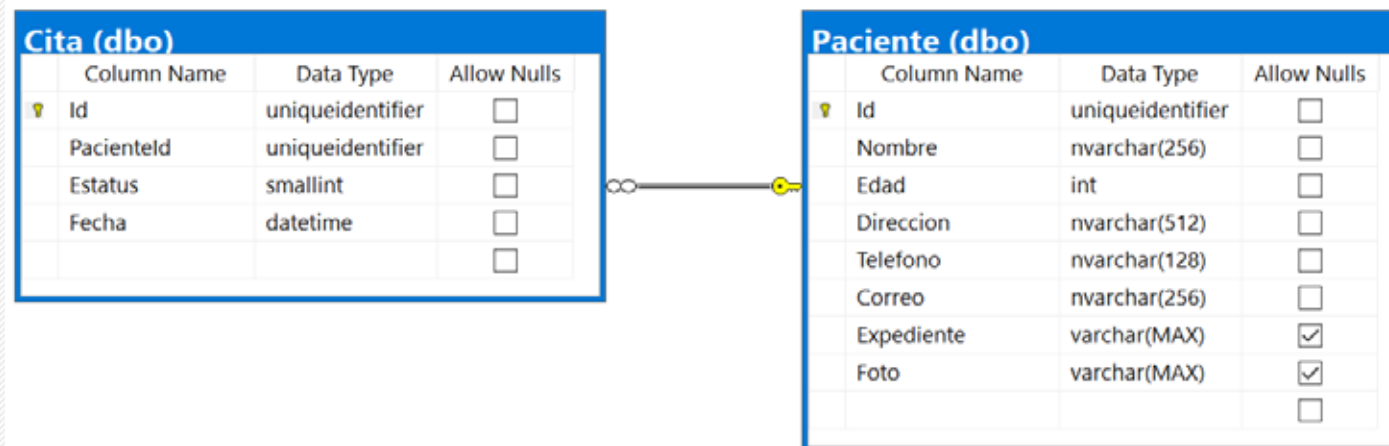
1. Registro_Proyecto_Investigacion_ISC_UNITEC

2. Programa de Fomento a la Investigación

https://myunitecedu-my.sharepoint.com/:f:/g/personal/juan_carrillog_my_unitec_edu_mx1/Eslq-tjCOplNgx_NpqjWNX0Bj6Q12FzHSXo61JBzZlnPrQ

Ejercicio

Producto (dbo)			
Column Name	Data Type	Allow Nulls	
Id	uniqueidentifier	<input type="checkbox"/>	
Nombre	nvarchar(128)	<input type="checkbox"/>	
Descripcion	nvarchar(512)	<input type="checkbox"/>	
Tipo	smallint	<input type="checkbox"/>	
Precio	float	<input type="checkbox"/>	
Cantidad	int	<input type="checkbox"/>	
Foto	varchar(MAX)	<input checked="" type="checkbox"/>	



```
CREATE TABLE [Producto](
    [Id] [uniqueidentifier] PRIMARY KEY NOT NULL,
    [Nombre] [nvarchar](128) NOT NULL,
    [Descripcion] [nvarchar](512) NOT NULL,
    [Tipo] [smallint] NOT NULL,
    [Precio] [float] NOT NULL,
    [Cantidad] [int] NOT NULL,
    [Foto] [varchar](max) NULL
);
```

```
CREATE TABLE [Paciente](
    [Id] [uniqueidentifier] PRIMARY KEY NOT NULL,
    [Nombre] [nvarchar](256) NOT NULL,
    [Edad] [int] NOT NULL,
    [Direccion] [nvarchar](512) NOT NULL,
    [Telefono] [nvarchar](128) NOT NULL,
    [Correo] [nvarchar](256) NOT NULL,
    [Expediente] [varchar](max) NULL,
    [Foto] [varchar](max) NULL
);
CREATE TABLE [Cita](
    [Id] [uniqueidentifier] PRIMARY KEY NOT NULL,
    [PacienteId] [uniqueidentifier] NOT NULL,
    [Estatus] [smallint] NOT NULL,
    [Fecha] [datetime] NOT NULL,
    FOREIGN KEY([PacienteId]) REFERENCES [Paciente] ([Id])
);
```