Revista Nexos Científicos
Junio 2022 pp. 25-31
Volumen 1, Número 1
Fecha de recepción: marzo 2022

ISSN: 2773-7489
Correo: editor@istvidanueva.edu.ec
URL: http://nexoscientificos.vidanueva.edu.ec/index.php/ojs/index
Fecha de aceptación: mayo 2022

Aplicación de un sistema de gerencia de clientes y camarógrafos usando JDBC

Jorge Beltran [1]; Oscar Naranjo [2]
[1,2] *Instituto Superior Tecnológico Vida Nueva, Quito-Ecuador, jorge.beltran@istvidanueva.edu.ec, oscar.naranjo@istvidanueva.edu.ec*

Resumen: Java™ Database Connection (JDBC) es la especificación de JavaSoft de una interfaz de programación de aplicaciones (API) estándar que permite a los programas Java acceder a los sistemas de gestión de bases de datos. La API JDBC consiste en un conjunto de interfaces y clases escritas en el lenguaje de programación Java. Mediante estas interfaces y clases estándar, los programadores pueden escribir aplicaciones que se conectan a las bases de datos, envían consultas escritas en lenguaje de consulta estructurado (SQL) y procesan los resultados. Dado que JDBC es una especificación estándar, un programa Java que utilice la API JDBC puede conectarse a cualquier sistema de gestión de bases de datos (SGBD), siempre que exista un controlador para ese SGBD en particular. De esta forma, es posible trabajar de forma directa con nuestras bases de datos, pudiendo modificarlas de forma directa gracias a nuestra codificación.
Palabras clave: Tecnología, Bases de datos relacionales, JDBC, SQL, Consultas.

Implementation of a client and cameraman management system using JDBC

Abstract: Java™ Database Connection (JDBC) is JavaSoft's specification of a standard application programming interface (API) that allows Java programs to access database management systems. The JDBC API consists of a set of interfaces and classes written in the Java programming language. Using these standard interfaces and classes, programmers can write applications that connect to databases, send queries written in Structured Query Language (SQL) and process the results. Because JDBC is a standard specification, a Java program using the JDBC API can connect to any database management system (DBMS), provided that a driver exists for that particular DBMS. In this way, it is possible to work directly with our databases, being able to modify them directly thanks to our coding.
Keywords: Technology**,** Relational Databases**,** JDBC**,** SQL**,** Queries.

1. INTRODUCCIÓN

En la actualidad, todo sistema que maneje su propio grupo de clientes, necesita estar enlazado casi por obligación a una base de datos, preferiblemente una relacional, esto debido a que en reiteradas ocasiones, es necesario acceder a ciertos datos de los clientes de los cuales ya se tiene conocimiento, como por ejemplo, saber la dirección de un cliente con dificultades para desplazarse a consumar la sesión previamente definida, o bien cuando hubo un problema a la hora de ingresar los detalles de una sesión al sistema y es necesario comunicar tanto al cliente como al camarógrafo ya sea mediante sus números de teléfono fijo o móviles, o bien mediante su dirección de correo electrónico.

Para un mantenimiento adecuado de nuestra base de datos se inició creando un proyecto en Netbeans con modalidad de Java Application, en la sección de Java with Ant.

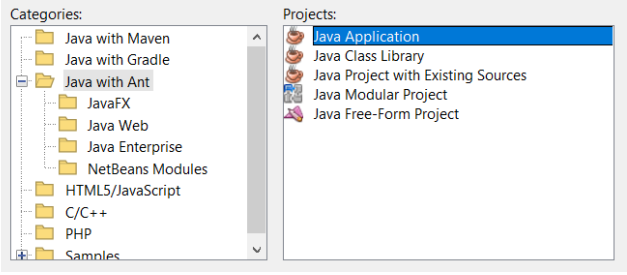

Ilustración 1. Java Application.

1. Estudiante

Revista Nexos Científicos
Junio 2022 pp. 25-31
Volumen 1, Número 1
Fecha de recepción: marzo 2022

ISSN: 2773-7489
Correo: editor@istvidanueva.edu.ec
URL: http://nexoscientificos.vidanueva.edu.ec/index.php/ojs/index
Fecha de aceptación: mayo 2022

Fuente: Autor

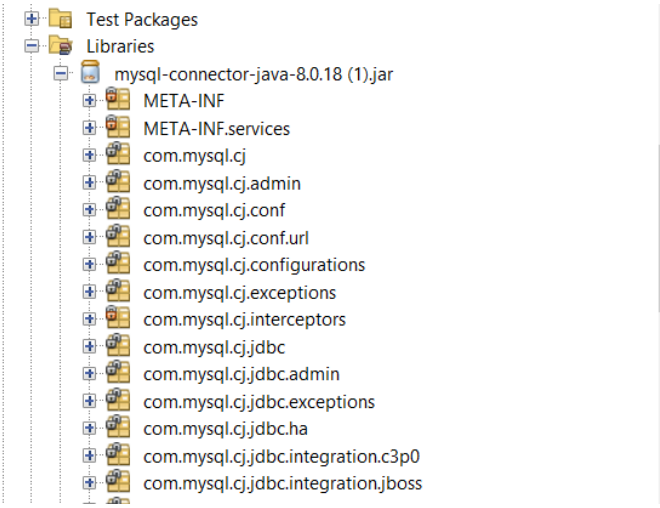

Ilustración 2. Archivo .jar para realizar la conexión a la base de datos.
Fuente: Autor

El desarrollo del proyecto se realiza en Netbeans, implementando la API de JDBC.

Una vez creado el proyecto es importante inicializar el componente con los siguientes parámetros:

```
/**
 * Creates new form Registro
 */
public Registro()
{
    initComponents();
}

int band =0;
public String driver = "com.mysql.cj.jdbc.Driver";
// Nombre de la base de datos
public String database = "registro";
// Host
public String hostname = "localhost";
// Puerto
public String port = "3306";
// Ruta de nuestra base de datos (desactivamos el uso de SSL con "?useSSL=false")
public String url = "jdbc:mysql://" + hostname + ":" + port + "/" + database + "?useSSL=false";
//public String url = "jdbc:mysql://" + hostname + ":/" + database;
// Nombre de usuario
public String username = "root";
// Clave de usuario
public String password = "";
```

Ilustración 3. InitComponents
Fuente: Autor

2. METODOLOGÍA

La metodología Extreme Programming XP también permite un desarrollo ágil y entregar funcionalidades del sistema parcial para mejorar en cada lanzamiento hasta obtener una versión estable que se pueda distribuir.

2.1.Planificación

Antes de empezar a trabajar en nuestro programa, es necesario crear nuestra base de datos por medio del programa XAMPP Control Panel.

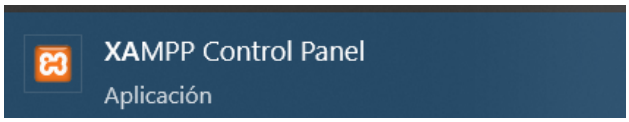

Ilustración 3. XAMPP.
Fuente: Autor

Una vez allí, creamos una base de datos llamado “registro”, dentro de la cual, crearemos otras 4 tablas: “persona”, “cliente”, “camarógrafo” y “sesiones”.

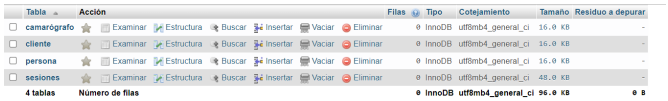

Ilustración 4. phpMyAdmin
Fuente: Autor

Y establecemos las columnas de cada tabla para poder empezar a trabajar:

Ilustración 5. Tabla persona.
Fuente: Autor

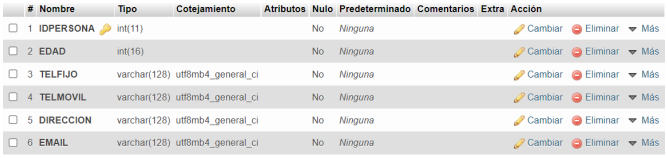

Ilustración 6. Tabla cliente
Fuente: Autor

Ilustración 7. Tabla camarógrafo
Fuente: Autor

Ilustración 8. Tabla sesiones
Fuente: Autor

2.2.Diseño

Revista Nexos Científicos
Junio 2022 pp. 25-31
Volumen 1, Número 1
Fecha de recepción: marzo 2022

ISSN: 2773-7489
Correo: editor@istvidanueva.edu.ec
URL: http://nexoscientificos.vidanueva.edu.ec/index.php/ojs/index
Fecha de aceptación: mayo 2022

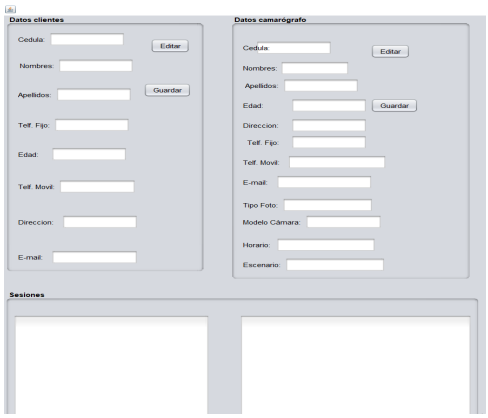

Ilustración 9. GUI.
Fuente: Autor

La interfaz gráfica de usuario escogida es aquella que permita tener acceso a todos los datos necesarios sin resultar demasiada compleja para aquel que vaya a probar el programa, esto siguiendo los estándares básicos de DCU (Diseño Centrado en el Usuario).

2.3.Desarrollo

Debido a que la variable cédula opera como nuestro identificador en la vida real, este será nuestro valor principal que modificaremos a la hora de ingresar nuevos datos.

```
private void
cedulaClActionPerformed(java.awt.event.ActionEvent
evt){
      char[] aux=cedulaCl.getText().toCharArray();
      if(evt.getKeyCode()==10)
      {
       if(aux.length==10)
       {
        try
        {
         Class.forName(driver);
        }
        catch(ClassNotFoundException e)
        {

System.out.println(e.getMessage());
                }
                try
                {
                    System.out.println(url);
                    Connection
conexion=DriverManager.getConnection(url, username,
password);
                    Statement
estatuto=conexion.createStatement();
                    ResultSet
rs=estatuto.executeQuery("select * from DatosClientes
where Cedula="+cedulaCl.getText()+);
                    while(rs.next())
                    {
                        band=1;
                        Object[] t=new Object[8];
                        for(int i=0;i<t.length;i++)
                        t[i]=rs.getObject(i+1);
                        nombresCl.setText(t[1]+"");
                        apellidosCl.setText(t[2]+"");
                        telffijoCl.setText(t[3]+"");
                        edadCl.setText(t[4]+"");
                        telfmovilCl.setText(t[5]+"");
                        direccionCl.setText(t[6]+"");
                        emailCl.setText(t[7]+"");
                    }

                    if(band==0)
                    {
                        est.setText("Nuevo Cliente -
Ingresar Datos");
                        nombresCl.setEditable(true);

apellidosCl.setEditable(true);
                        telffijoCl.setEditable(true);
                        edadCl.setEditable(true);
                       telfmovilCl.setEditable(true);
                       direccionCl.setEditable(true);
                        emailCl.setEditable(true);
                        nombresCl.setText("");
                        apellidosCl.setText("");
                        telffijoCl.setText("");
                        edadCl.setText("");
                        telfmovilCl.setText("");
                        direccionCl.setText("");
                        emailCl.setText("");
                        Guardar1.setEnabled(true);
                        Editar1.setEnabled(false);
                    }
                    else
                    {
                        est.setText("");
                        nombresCl.setEditable(false);
                      apellidosCl.setEditable(false);
                       telffijoCl.setEditable(false);
                        edadCl.setEditable(false);
                      telfmovilCl.setEditable(false);
                      direccionCl.setEditable(false);
                        emailCl.setEditable(false);
                        Editar1.setEnabled(true);
                        Guardar1.setEnabled(false);
                    }

                    est.close();
                    conexion.close();
                }
                catch(Exception e)
                {

System.out.println(e.getMessage());
                }
            }
            else
            {
javax.swing.JOptionPane.showMessageDialog(this,"¡Cédu
la Incorrecta!","Error",0);
        }
    }
```

Sin embargo, resulta sumamente importante la necesidad de establecer un medio para actualizar esos datos con nueva información:

```
private void
Editar1ActionPerformed(java.awt.event.ActionEvent
evt) {

        nombresCl.setEditable(true);
```

Revista Nexos Científicos
Junio 2022 pp. 25-31
Volumen 1, Número 1
Fecha de recepción: marzo 2022
ISSN: 2773-7489
Correo: editor@istvidanueva.edu.ec
URL: http://nexoscientificos.vidanueva.edu.ec/index.php/ojs/index
Fecha de aceptación: mayo 2022

```
        apellidosCl.setEditable(true);

        telffijoCl.setEditable(true);

        edadCl.setEditable(true);

        telfmovilCl.setEditable(true);

        direccionCl.setEditable(true);

        emailCl.setEditable(true);

        Guardar1.setEnabled(true);

        Editar1.setEnabled(false);

    }
```

Y tampoco se puede obviar una función para colocar todos los datos que vayamos ingresando en la interfaz que hayamos establecido.

```
try
        {
            Class.forName(driver);
        }
        catch(ClassNotFoundException e)
        {
            System.out.println(e.getMessage());
        }
        try
        {
            Connection conexion =
DriverManager.getConnection(url, username, password);
            Statement estatuto =
conexion.createStatement();
            if(!estatuto.getText().equals(""))
            {
                String sql="insert into DatosClientes
values('"+cedulaCl.getText()+"','"+nombresCl.getText(
)+"','"+apellidosCl.getText()+"','"+telffijoCl.getTex
t()+"','"+edadCl.getText()+"','"+telfmovilCl.getText(
)+"','"+direccionCl.getText()+"','"+emailCl.getText()
)"')";
                System.out.println(sql);
                est.executeUpdate(sql);
            }
            else
            {
                est.executeUpdate("update
DatosClientes set Nombres='"+nombresCl.getText()+"'"
+
                ",Apellidos='"+apellidosCl.getText()
+"' " +

",TelefonoFijo='"+telffijoCl.getText() +"' " +
                ", Edad='"+edadCl.getText() +"' " +

",TelefonoMovil='"+telfmovilCl.getText() +"' " +
                ",Direccion='"+direccionCl.getText()
+"' " +
                ",E-mail='"+emailCl.getText() +"' " +
                "where
Cedula='"+cedulaCl.getText()+"'");
            }
            ResultSet
rs=estatuto.executeQuery("select * from DatosClientes
where Cedula='"+cedulaCl.getText()+"'");
            int band =0;
            while(rs.next())
            {
                band=1;
                Object[] t=new Object[8];
                for(int i=0;i<t.length;i++)
                t[i]=rs.getObject(i+1);
                nombresCl.setText(t[1]+"");
                apellidosCl.setText(t[2]+"");
                telffijoCl.setText(t[3]+"");
                edadCl.setText(t[4]+"");
                telfmovilCl.setText(t[5]+"");
                direccionCl.setText(t[6]+"");
                emailCl.setText(t[7]+"");
            }
            if(band==0)
            {
                est.setText("Error - Cliente no
Ingresado");
                nombresCl.setEditable(true);
                apellidosCl.setEditable(true);
                telffijoCl.setEditable(true);
                edadCl.setEditable(true);
                telfmovilCl.setEditable(true);
                direccionCl.setEditable(true);
                emailCl.setEditable(true);
                cedulaCl.setText("");
            }
        }
    }
```

Con eso culminaríamos todas las labores de inserción de datos del lado de los clientes, faltarían los datos de camarógrafos, los cuales sería una codificación muy similar con excepción de que se cambian algunas variables y se añaden algunas nuevas. Empezamos con la cédula de parte de los camarógrafos:

```
char[] aux=cedulaCa.getText().toCharArray();
        if(evt.getKeyCode()==10)
        {
            if(aux.length==10)
            {
                try
                {
                    Class.forName(driver);
                }
                catch(ClassNotFoundException e)
                {

System.out.println(e.getMessage());
                }
                try
                {
                    System.out.println(url);
                    Connection
conexion=DriverManager.getConnection(url, username,
password);
                    Statement
estatuto=conexion.createStatement();
                    ResultSet
rs=estatuto.executeQuery("select * from
DatosCamarógrafo where Cedula="+cedulaCa.getText()+);
                    while(rs.next())
                    {
```

Revista Nexos Científicos
Junio 2022 pp. 25-31
Volumen 1, Número 1
Fecha de recepción: marzo 2022

ISSN: 2773-7489
Correo: editor@istvidanueva.edu.ec
URL: http://nexoscientificos.vidanueva.edu.ec/index.php/ojs/index
Fecha de aceptación: mayo 2022

```
                        band=1;
                        Object[] t=new Object[8];
                        for(int i=0;i<t.length;i++)
                        t[i]=rs.getObject(i+1);
                        nombresCa.setText(t[1]+"");
                        apellidosCa.setText(t[2]+"");
                        telffijoCl.setText(t[3]+"");
                        edadCl.setText(t[4]+"");
                        telfmovilCl.setText(t[5]+"");
                        direccionCl.setText(t[6]+"");
                        emailCl.setText(t[7]+"");
                    }

                    if(band==0)
                    {
                        estatuto.setText("Nuevo
Cliente - Ingresar Datos");
                        nombresCa.setEditable(true);
                       apellidosCa.setEditable(true);
                        edadCa.setEditable(true);
                       direccionCa.setEditable(true);
                        telffijoCa.setEditable(true);
                       telfmovilCa.setEditable(true);
                        tipofotoCa.setEditable(true);
                   modelocamaraCa.setEditable(true);
                        horarioCa.setEditable(true);
                       escenarioCa.setEditable(true);

                        nombresCa.setText("");
                        apellidosCa.setText("");
                        edadCa.setText("");
                        direccionCa.setText("");
                        telffijoCa.setText("");
                        telfmovilCa.setText("");
                        emailCa.setText("");
                        tipofotoCa.setText("");
                        modelocamaraCa.setText("");
                        horarioCa.setText("");
                        escenarioCa.setText("");
                        Guardar1.setEnabled(true);
                        Editar1.setEnabled(false);
                    }
                    else
                    {
                        estatuto.setText("");
                        nombresCa.setEditable(false);
                      apellidosCa.setEditable(false);
                        edadCa.setEditable(false);
                      direccionCa.setEditable(false);
                       telffijoCa.setEditable(false);
                      telfmovilCa.setEditable(false);
                        emailCa.setEditable(false);
                       tipofotoCa.setEditable(false);
                   modelocamaraCa.setEditable(false);
                        horarioCa.setEditable(false);
                      escenarioCa.setEditable(false);
                        Editar1.setEnabled(true);
                        Guardar1.setEnabled(false);
                    }

                    estatuto.close();
                    conexion.close();
                }
                catch(Exception e)
                {

System.out.println(e.getMessage());
                }
            }
            else
            {
javax.swing.JOptionPane.showMessageDialog(this,"¡Cédu
la Incorrecta!","Error",0);
```

```
private void
Editar2ActionPerformed(java.awt.event.ActionEvent
evt) {
        nombresCa.setEditable(true);
        apellidosCa.setEditable(true);
        edadCa.setEditable(true);
        direccionCa.setEditable(true);
        telffijoCa.setEditable(true);
        telfmovilCa.setEditable(true);
        emailCa.setEditable(true);
        tipofotoCa.setEditable(true);
        modelocamaraCa.setEditable(true);
        horarioCa.setEditable(true);
        escenarioCa.setEditable(true);
        Guardar2.setEnabled(true);
        Editar2.setEnabled(false);
    }
```

```
        }
    }
```

Ahora, procedemos con la función de Editar.
Y por último, la función de Guardar:

```
private void
Guardar2ActionPerformed(java.awt.event.ActionEvent
evt) {
        try
        {
            Class.forName(driver);
        }
        catch(ClassNotFoundException e)
        {
            System.out.println(e.getMessage());
        }
        try
        {
            Connection
conexion=DriverManager.getConnection(url, username,
password);
            Statement est=conexion.createStatement();
            if(!est.getText().equals(""))
            {
                String sql="insert into
DatosCamarógrafo
values('"+cedulaCa.getText()+"','"+nombresCa.getText(
)+"','"+apellidosCa.getText()+"','"+edadCa.getText()+
"','"+direccionCa.getText()+"','"+telffijoCa.getText(
)+"','"+telfmovilCa.getText()+"','"+emailCa.getText()
)"','"+tipofotoCa.getText())"','"+modelocamaraCa.getT
ext())"','"+horarioCa.getText())"','"+escenarioCa.get
Text())"')";
                System.out.println(sql);
                est.executeUpdate(sql);
            }
            else
            {
                est.executeUpdate("update
DatosClientes set Nombres='"+nombresCa.getText()+"'"
+
                ",Apellidos='"+apellidosCa.getText()
+"' " +
                ",Edad='"+edadCa.getText() +"' " +
                ",Direccion='"+direccionCa.getText()
+"' " +
              ",TelefonoFijo='"+telffijoCa.getText()
+"' " +
```

Revista Nexos Científicos
Junio 2022 pp. 25-31
Volumen 1, Número 1
Fecha de recepción: marzo 2022
ISSN: 2773-7489
Correo: editor@istvidanueva.edu.ec
URL: http://nexoscientificos.vidanueva.edu.ec/index.php/ojs/index
Fecha de aceptación: mayo 2022

```
",TelefonoMovil='"+telfmovilCa.getText() +"' " +
                ",E-mail='"+emailCa.getText() +"' " +
                ",TipoFoto='"+tipofotoCa.getText()
+"' " +

",ModeloCamara='"+modelocamaraCa.getText() +"' " +
                ",Horario='"+horarioCa.getText() +"'
" +
                ",Escenario='"+escenarioCa.getText()
+"' " +
                "where
Cedula='"+cedulaCa.getText()+"'");
            }
            Result Set rs=est.executeQuery("select *
from DatosCamarógrafo where
Cedula='"+cedulaCa.getText()+"'");
            int band =0;
            while(rs.next())
            {
                band=1;
                Object[] t=new Object[8];
                for(int i=0;i<t.length;i++)
                t[i]=rs.getObject(i+1);
                nombresCa.setText(t[1]+"");
                apellidosCa.setText(t[2]+"");
                edadCa.setText(t[3]+"");
                direccionCa.setText(t[4]+"");
                telffijoCa.setText(t[5]+"");
                telfmovilCa.setText(t[6]+"");
                emailCa.setText(t[7]+"");
                tipofotoCa.setText(t[7]+"");
                modelocamaraCa.setText(t[7]+"");
                horarioCa.setText(t[7]+"");
                escenarioCa.setText(t[7]+"");
            }
            if(band==0)
            {
                est.setText("Error - Cliente no
Ingresado");
                nombresCa.setEditable(true);
                apellidosCa.setEditable(true);
                edadCa.setEditable(true);
                direccionCa.setEditable(true);
                telffijoCa.setEditable(true);
                telfmovilCa.setEditable(true);
                emailCa.setEditable(true);
                tipofotoCa.setEditable(true);
                modelocamaraCa.setEditable(true);
                horarioCa.setEditable(true);
                escenarioCa.setEditable(true);
                cedulaCa.setText("");
            }
        }
    }
```

3. RESULTADOS Y DISCUSIÓN

3.1.Identificación de contenido.

Como se mencionó anteriormente, las bases de datos permiten tener una gran cobertura a la hora de tener acceso a grandes cantidades de información, he ahí la razón del por qué el lenguaje Java no se queda atrás en cuanto a manejo de base de datos mediante APIs y Frameworks como JDBC y JPA.

3.2.Resultados.

Gracias a los conceptos aplicados, podemos concluir que es posible operar en conjunto con bases de datos, sin importar el tipo de programa que querramos desarrollar en Netbeans usando el lenguaje Java, adicional a ello, podemos dar por hecho, la importancia de JPA y JDBC a la hora de trabajar con bases de datos relacionales.

Cabe recalcar que hubieron serias complicaciones por problemas de puertos, principalmente porque otros software ya lo estaban empleando y se tuvo que asignar un puerto aleatorio compuesto de 4 dígitos en las configuraciones de XAMPP Control Panel.

4. CONCLUSIONES

- Es posible trabajar tanto con bases de datos relaciones como no relacionales, incluyendo todas las funciones que éstas ofrecen como Vistas, Llaves primarias, Llaves foráneas, entre otros.
- Es importante tener precaución con los valores declarados al inicio asociadas a la conexión de la base de datos porque pueden repercutir de forma negativa, por ejemplo, representando un problema al conectarse a la base de datos por error de puertos.
- Se tuvo que cambiar la metodología a usar que fue planteada al inicio, puesto que resultaría más factible, en asuntos de versatilidad, trabajar con Netbeans que con Visual Studio, asimismo el producto final esperado tuvo algunas variaciones.

REFERENCIAS

- Herrmann, N. (1996). The Whole Brain Business Book: Unblocking the Power of Whole Brain Thinking in Organizations and Individuales. New York- McGrawHill
- Herrmann International (2009). Understanding the Herrmann Whole Brain Model, HBDI Profile Package.
- Holiday, R. (2014). Growth hacker marketing: a primer on the future of PR, marketing,and advertising. Penguin.

- Innovation (2018). Imnovation: Qué es el blockchain y cómo funciona.Consultado el 03 de febrero https://www.imnovation-hub.com/es/transformacion-digital/que-esblockchain-y-como-funciona-esta-tecnologia/
- Instituto de Ingeniería del conocimiento. (2018). Big data: Aprovecha los datos para extraer información de valor para tu negocio. Consultado el 31 de enero del 2021 en https://www.iic.uam.es/big-data/ Jaramillo, C.,
- Ocampo, E., Ríos, P., & Estrada, E. (2020). Metodologías para la innovación. Jara, I., & Ochoa, J.
- M. (2020). Usos y efectos de la inteligencia artificial en educación. Sector Social división educación. Documento para discusión número IDB-DP-00-776. BID. doi: http://dx. doi. org/10.18235/000238 0.
- Knox, J. (2020). Artificial intelligence and education in China. Learning, Media and Technology, 45(3), 298-311.
- Marbaise, M., & 50Minutes.fr (2015). Business model canvas: Élaborer une stratégie de développement. ProQuest Ebook Central https://www.mdconsult.internacional.edu.ec:2095
- Nanyang Technological University. (s.f.) https://www.ntu.edu.sg/. Obtenido de https://www3.ntu.edu.sg/home/ehchua/programming/java/JDBC_Basic.html
- IBM. (2020). https://www.ibm.com/docs/en. Obtenido de https://www.ibm.com/docs/en: https://www.ibm.com/docs/en/informix-servers/12.10?topic=started-what-is-jdbc