**Universidad Autónoma Gabriel René Moreno**

Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones

NODOS COGNITIVOS  


INTEGRANTES

* Alvarez Orias Genaro Mauricio……………………………………………215001567
* Oviedo Acosta Natalia…………………………………………………...…215029607

DOCENTE

* Ing. Veizaga Josue Obed

FECHA

* 17/08/2017

Santa Cruz – Bolivia

**Tabla de Contenidos**

[Perfil del proyecto 3](#_Toc491123578)

[1.1 Introducción 3](#_Toc491123579)

[1.2 Objetivos 4](#_Toc491123580)

[1.2.1 Objetivo general 4](#_Toc491123581)

[1.2.2 Objetivos específicos 4](#_Toc491123582)

[1.3 Antecedentes (Moocs) 5](#_Toc491123583)

[1.4 Justificación 7](#_Toc491123584)

[1.5 Descripción del problema 8](#_Toc491123585)

[1.6 Formulación del problema 9](#_Toc491123586)

[1.7 Alcance 10](#_Toc491123587)

[1.7.1 Módulo Informacion 10](#_Toc491123588)

[1.7.2 Módulo Parámetros 10](#_Toc491123589)

[1.7.3 Módulo Visor 11](#_Toc491123590)

[Diseño de la base de datos 13](#_Toc491123591)

[2.1 Diseño conceptual 13](#_Toc491123592)

[2.1.1 Lista De Clases y Descripción 14](#_Toc491123593)

[2.2 Diseño lógico 15](#_Toc491123594)

[2.3 Diseño físico 16](#_Toc491123595)

[2.3.1. Tablas de Volumen 16](#_Toc491123596)

[2.3.2 Script 18](#_Toc491123597)

[Bibliografia 22](#_Toc491123598)

# Perfil del proyecto

## Introducción

Por mucho tiempo la educación se limitó a memorizar; sin embargo, con el paso del tiempo surgieron nuevas metodologías de enseñanza que se centraban en el aprendizaje desde puntos de vista muy distintos.

El acelerado desarrollo de actual sociedad y las transformaciones realizadas en la educación, justifican la actualidad e importancia de los nodos cognitivos; este se vincula con el proceso de enseñanza-aprendizaje en el que se desarrolla la formación de los estudiantes; y se vincula con la gestión de los conocimientos dirigida al logro de una eficiente integración de los contenidos de los diferentes textos.

En el proceso enseñanza-aprendizaje deben alcanzarse mejores resultados; principalmente en la búsqueda de la integración de los conocimientos, teniendo en cuenta los nodos cognitivos, por lo beneficioso que es para todas las asignaturas, y porque favorece el desarrollo de la cultura integral de los estudiantes a partir de que pueden asimilar e integrar los contenidos con los elementos de la experiencia práctica y las vivencias

Hoy en día el conocimiento deja de ser algo inalcanzable, ya que hay diversas formas de lograr aprender. Llego a tal punto que se puede vender una forma de conocimiento. En otras palabras vender un sistema en el cual se pueda almacenar ese conocimiento de forma didáctica e interesante.

Los nodos cognitivos se enfocan en intentar comprender la naturaleza de los procesos mentales de los estudiantes y de la naturaleza del aprendizaje. Establece conexiones inteligentes, dinámicas de la información con otra u otra para dar solución a un problema o para la adquisición de nuevos conocimientos.

Los nodos se “activan” al lograr responder una pregunta de retroalimentación del nodo anterior de forma correcta. Si la respuesta es falsa, se retorna al nodo anterior para así seguir su aprendizaje hasta que la persona sea capaz de responder la retroalimentación con éxito.

El nodo base (primero) es libre y desde ahí va marcando la secuencia en la cual se pueden ir desbloqueando cada uno de los nodos y cada parte del conocimiento.

## 1.2 Objetivos

### 1.2.1 Objetivo general

Desarrollar un sistema de información para la visualización de nodos cognitivos en un sistema Android

### 1.2.2 Objetivos específicos

* Recolectar información acerca de los nodos cognitivos; su uso, estructura y manejo.
* Analizar la información que se ha recolectado y buscar diferentes componentes que ayuden a mejorar y ayudar en el proyecto.
* Diseñar un sistema apropiado para solucionar los problemas existentes en el visualizador al leer la base de datos.
* Implementar la base de datos, ésta se realizará en el gestor de base de datos de SQL Server.
* Implementar la interfaz del software acorde a los requerimientos del usuario, en la plataforma Android, lenguaje Java.
* Evaluar cada etapa del desarrollo del software con el Tutor, para corregir las falencias de manejo y así verificar si cumple con todas las exigencias.

## 1.3 Antecedentes (Moocs)

La educación a los niños debe de ser hecha por medio de un proceso de instrucción, ensayo y retroalimentación, antes de hacerlo con una simple instrucción. Albert Bandura hizo hincapié en que la auto eficiencia, definida como la confianza que el individuo tiene sobre sus propias habilidades para desempeñar diversas conductas, es importante para aprender y desarrollar comportamientos socialmente aceptados. Así, el desarrollo de habilidades no sólo se convierte en una cuestión de comportamiento externo, sino de cualidades internas.

Teniendo nuevas formas de aprendiza que abrieron paso al aprendizaje de la forma de nodos cognitivos.



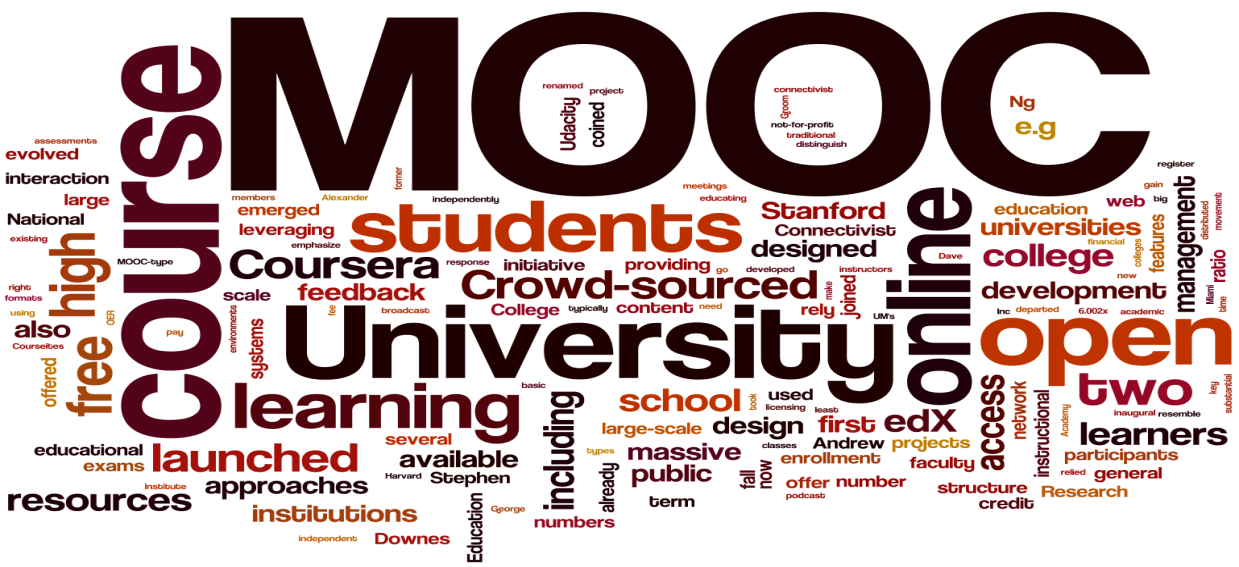
El Micro aprendizaje es una estrategia educativa que se caracteriza por la brevedad de las lecciones de aprendizaje.

El material o contenido que apoya las lecciones de microaprendizaje puede presentarse en diferentes formas (actividades, juegos, videos, preguntas, diapositivas, discusiones, etc.



Un sitio web destinado al aprendizaje gratuito de idiomas a la vez que una plataforma crowdsourcing de traducción de textos. El servicio está diseñado de tal forma que a medida que el usuario avanza en su aprendizaje ayuda a traducir páginas web y otros documentos.

Con un sistema fácil de utilizar. Ayuda a entender más el idioma utilizando preguntas y las respuestas que ayudan a controlar en qué nivel están o si ya están listos para preguntas y razonamiento más complicado.



A simple vista, los cursos MOOC no se diferencian mucho de los cursos en línea tradicionales que existen desde hace años, ya que al igual que un curso en línea tradicional, un curso MOOC posee un temario o programa; unos materiales, que normalmente son un conjunto de vídeos, pero que también pueden incluir lecturas; unas actividades que se pueden evaluar de diferentes formas (autoevaluación, evaluación automática, evaluación entre pares); unos ejercicios de tipo test para evaluar el aprendizaje; y un foro para discutir con el profesor o con otros estudiantes.

Sin embargo, una mirada más profunda nos revela que el papel, o más bien, el comportamiento que presenta un alumno en un curso MOOC es distinto al comportamiento que presenta en un curso en línea tradicional; además, el carácter de masivo que implica que en un curso pueden coexistir decenas de miles de alumnos al mismo tiempo, crea una clara diferencia cualitativa (y claro está, cuantitativa), respecto a los cursos en línea tradicionales

## 1.4 Justificación

* Social

En el mundo hay un aproxima de 12000 universidades y 125000 colegios los cuales todavía tienen formas de enseñanza bastante obsoletos. El software está buscando ayudar a los profesores y docentes a tener una nueva forma de enseñanza más didáctica y fácil de usar sin ningún costo para los alumnos.

* Económica

El software tendrá dos partes. La parte del generador de nodos el cual tendrá un costo y el visor que sea gratis. Así solo el usuario que compre el generador pagara por el uso y cualquiera con el visor podrá ver y usar lo generado.

* Científica

Para el software se usara Android Studios; lenguaje Java. Como también la estructura de datos conocida como árboles para crear las redes y plataformas que cada usuario desee.

## 1.5 Descripción del problema

## 1.6 Formulación del problema

Debido a que, hasta el momento la forma de enseñanza en los colegios y en algunas universidades sigue siendo de manera analógica estando en una era completamente digital, se vio la necesidad de desarrollar un sistema de información con una nueva forma de enseñanza más eficaz, más fácil de utilizar y mucho más entretenida con una estructura diseñada para autoevaluar al usuario por cada una de las unidades y cursos.

## 1.7 Alcance

El proyecto tiene como finalidad facilitar las tareas Críticas, priorizando las más necesarias que presenta actualmente el sistema de nodos cognitivos, además de simplificar y automatizar otras tareas. Por lo tanto, el sistema contará con los siguientes módulos:

### Módulo Informacion

Este módulo se centra en la información básica del usuario y los perfiles de los clientes.

**1.7.1.1 Gestionar clientes. –** Es necesario guardar toda la información básica del cliente que vaya a ser uso del sistema. Así tener una mejor seguridad de quien está haciendo uso del sistema. Se va a guardar los siguientes datos:

* **Persona**

**Id**

Nombre

Apellido

Email

Telefono

**1.7.1.2 Administrar Perfil. –** Cada uno de los usuarios van a tener un perfil en el cual van a poder acceder utilizando un login y podrán modificar su información a su gusto.

### Módulo Parámetros

Este módulo se centra en todos los parámetros que utiliza el administrador del sistema. Capaz de visualizar las actividades de los usuarios y dar soporte al sistema.

**1.7.2.1 Detallar Bitacora. –** El sistema tiene que ser capaz de guardar cada actividad hecha por cada usuario en específico, tomando en cuenta su hora y día.

* **Bitacora**

**Id**

Inicio

Fin

* **Detalle\_Bitacora**

**Id**

Accion

Hora

Tabla

**1.7.2.2 Realizar Backup. –** Al insertar, modificar o eliminar cualquier nodo, curso, pregunta, etc todo se almacena en un backup manual del administrador

**1.7.2.3 Administrar Privilegios. –** El administrador, los clientes y los usuarios al tener cada uno diferentes niveles de acceso a la aplicación, se necesita registrar los casos de uso a los cuales va a tener acceso cada persona.

### Módulo Visor

**1.7.3.1 Gestionar Curso. –** Un curso viene a ser el primer nivel de aprendizaje, hecho por varios nodos en el cual solo llega a ser un titulo general de que se podrá aprender al hacer uso del curso

* **Nodo**

**Id**

Nombre

Descripcion

Concluido

Puntaje

**1.7.3.2 Gestionar Nodo. –** Un nodo viene a ser un nivel mas especifico de un nodo, conocido como una unidad. En dicha unidad se va a poder apreciar un grupo de subunidades para mejor el aprendizaje de la manera de la micro enseñanza.

* **SubNodo**

**Id**

Titulo

Contenido

Aprobado

Habilitado

Puntaje

**1.7.3.3 Gestionar SubNodo. –** Es necesario guardar toda la información del contenido en una base de datos para así poder ingresar a la misma en visor. También para poder confirmar si es que ya cumplió con la retroalimentación y aprobó el subnodo.

* **Nodo**

**Id**

Titulo

Contenido

Aprobado

**1.7.3.4 Gestionar Pregunta. –** Al insertar una pregunta o retroalimentación, es necesario saber de qué nodo es la pregunta. Como su descripción y su el tipo de pregunta que es. También cada uno de las respuestas de dicha pregunta y su enlace con los nodos correspondientes.

* **Pregunta**

**Id**

Descripción

* **Respuesta**

**Id**

Descripción

Tipo

Comentario

# Diseño de la base de datos

## 2.1 Diseño conceptual

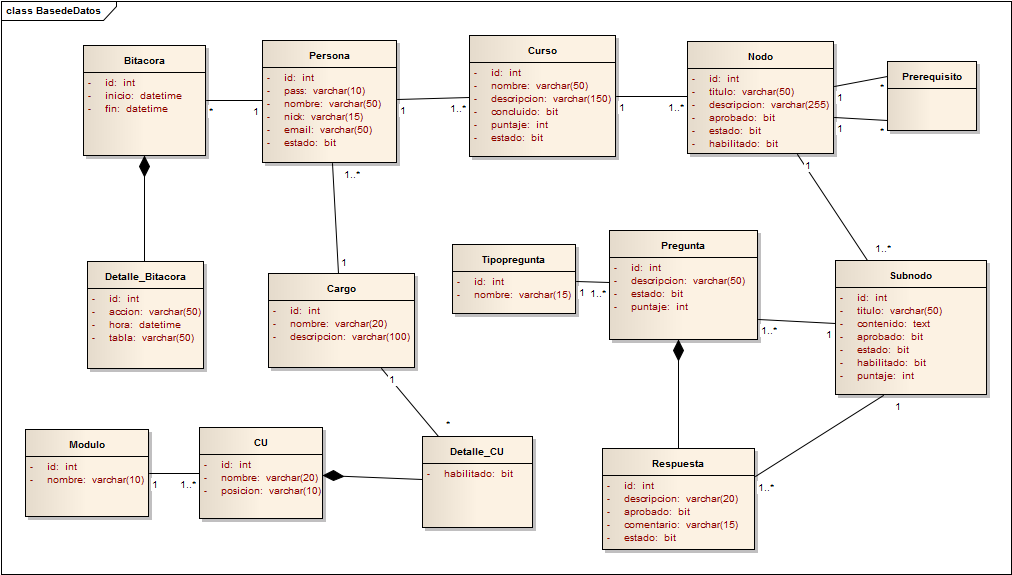
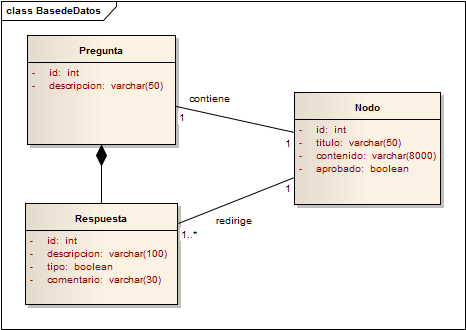
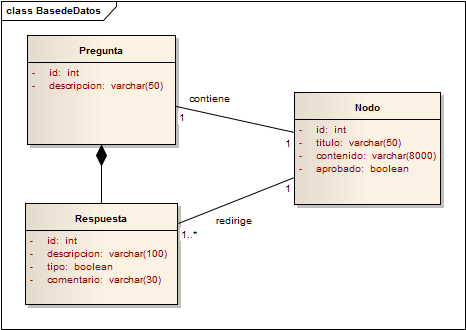


Diagrama de clase estático (lógico)

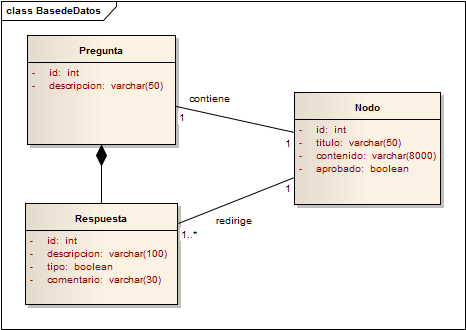
### 2.1.1 Lista De Clases y Descripción

1. 

La tabla Pregunta, es la retroalimentación de cada nodo y es compuesta de varias respuestas. La tabla tiene relación de asociación con la tabla nodo y una de composición con la tabla respuestas.

1. 

La tabla Respuesta, es la composición de la tabla pregunta y redirige a un nodo en específico. La tabla tiene relación de asociación con la tabla nodo y una de composición con la tabla respuestas.

1. 

La tabla Nodo, es el “conocimiento” con un atributo que verifica su aprobación. La tabla tiene relación de asociación con la tabla pregunta y la tabla respuestas.

## 2.**2 Diseño lógico**

*Pregunta*

|  |  |
| --- | --- |
| PK |  |
| id | descripcion |

*Nodo*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PK |  |  |  | FK |
| id | Titulo | Contenido | Aprobado | idPregunta |

*Respuesta*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | FK |
| id | idPregunta | Descripcion | Tipo | Comentario | idNodo |

*Modulo*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PK |  |  |  | FK |
| id | Titulo | Contenido | Aprobado | idPregunta |

*Caso de Uso*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | FK |
| id | idPregunta | Descripcion | Tipo | Comentario | idNodo |

*Cargo*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PK |  |  |  | FK |
| id | Titulo | Contenido | Aprobado | idPregunta |

*Bitacora*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | FK |
| id | idPregunta | Descripcion | Tipo | Comentario | idNodo |

*Usuario*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PK |  |  |  | FK |
| id | Titulo | Contenido | Aprobado | idPregunta |

*Detalle\_Bitacora*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | FK |
| id | idPregunta | Descripcion | Tipo | Comentario | idNodo |

## 2.3 Diseño físico

### 2.3.1. Tablas de Volumen

*Nodo*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| Codigo\_Usuario | Int |  | NO | PK | Llave primaria de la tabla Usuario |
| Nombre | varchar | 30 | NO |  | Nombre del usuario |
| email | varchar | 10 | NO |  | Email del Usuario |
| Tipo | varchar | 1 | NO |  | Tipo de Usuario: Persona o Empresa |
| estado | bit |  | NO |  | Estado del Usuario(Activo-Pasivo) |

*Pregunta*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| ID | Int |  | NO | PK | Llave primaria de la tabla Telefono |
| Numero | Int |  | NO |  | Numero de telefono |
| Codigo\_Usuario | Int |  | NO | FK | Llave foránea a usuario |

*Respuesta*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| Nro | Int | ░░░░ | NO | PK | Llave primaria de la tabla Reserva. |
| Descripcion | Varchar | 50 | NO | ░░░░ |  |
| Fecha\_Solicitud | Date | ░░░░ | NO | ░░░░ | Fecha en que se realiza la reserva. |
| Fecha\_Reserva | Date | ░░░░ | NO | ░░░░ | Fecha en que se realizará el evento. |
| Ubicación | Varchar | 50 | NO | ░░░░ | Dirección del evento |
| Descuento | Float | ░░░░ | ░░░░ | ░░░░ | Cantidad de dinero descontado del total. |
| Monto\_Adicional | Float | ░░░░ | ░░░░ | ░░░░ | Cantidad de dinero añadido a |
| Monto\_Cancelado | Float | ░░░░ | NO | ░░░░ | Cantidad de dinero cancelado por adelantado. |
| Codigo\_Usuario | Int | ░░░░ | NO | FK | Llave foránea de la tabla Persona. |
| Nro\_Privado | Int | ░░░░ | NO | FK | Llave foránea de la tabla Normal. |
| tipo | varchar | 1 | NO | ░░░░ | Permite especificar si un evento es protocolar o normal. |
| estado | bit | ░░░░ | NO | ░░░░ | Especifica el estado de un evento. |

*Usuario*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| Codigo\_Usuario | Int |  | NO | PK | Llave primaria de la tabla Usuario |
| Nombre | varchar | 30 | NO |  | Nombre del usuario |
| email | varchar | 10 | NO |  | Email del Usuario |
| Tipo | varchar | 1 | NO |  | Tipo de Usuario: Persona o Empresa |
| estado | bit |  | NO |  | Estado del Usuario(Activo-Pasivo) |

*Bitacora*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| ID | Int |  | NO | PK | Llave primaria de la tabla Telefono |
| Numero | Int |  | NO |  | Numero de telefono |
| Codigo\_Usuario | Int |  | NO | FK | Llave foránea a usuario |

*Detalle\_Bitacora*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| Codigo\_Usuario | Int |  | NO | PK | Llave primaria de la tabla Usuario |
| Nombre | varchar | 30 | NO |  | Nombre del usuario |
| email | varchar | 10 | NO |  | Email del Usuario |
| Tipo | varchar | 1 | NO |  | Tipo de Usuario: Persona o Empresa |
| estado | bit |  | NO |  | Estado del Usuario(Activo-Pasivo) |

*Cargo*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| ID | Int |  | NO | PK | Llave primaria de la tabla Telefono |
| Numero | Int |  | NO |  | Numero de telefono |
| Codigo\_Usuario | Int |  | NO | FK | Llave foránea a usuario |

*Caso de Uso*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| Codigo\_Usuario | Int |  | NO | PK | Llave primaria de la tabla Usuario |
| Nombre | varchar | 30 | NO |  | Nombre del usuario |
| email | varchar | 10 | NO |  | Email del Usuario |
| Tipo | varchar | 1 | NO |  | Tipo de Usuario: Persona o Empresa |
| estado | bit |  | NO |  | Estado del Usuario(Activo-Pasivo) |

*Modulo*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ATRIBUTOS | TIPO DE DATOS | AMPLITUD | NULO | LLAVE | DESCRIPCION |
| ID | Int |  | NO | PK | Llave primaria de la tabla Telefono |
| Numero | Int |  | NO |  | Numero de telefono |
| Codigo\_Usuario | Int |  | NO | FK | Llave foránea a usuario |

### 2.3.2 Script

CREATE DATABASE NodosCognitivos;

USE NodosCognitivos;

--//////////////////////////////////////////////////////////

--MODULO / PAQUETES DE LOS CASOS DE USO

--//////////////////////////////////////////////////////////

CREATE TABLE MODULO

(

ID INT PRIMARY KEY,

NOMBRE VARCHAR(20) NOT NULL

);

--//////////////////////////////////////////////////////////

-- CASOS DE USO

--//////////////////////////////////////////////////////////

CREATE TABLE CU

(

ID INT PRIMARY KEY,

NOMBRE VARCHAR(20) NOT NULL,

POSICION VARCHAR(10),

ID\_MODULO INT NOT NULL,

FOREIGN KEY(ID\_MODULO) REFERENCES MODULO(ID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

--//////////////////////////////////////////////////////////

--CARGOS / PRIVILEGIOS

--//////////////////////////////////////////////////////////

CREATE TABLE CARGO

(

ID INT PRIMARY KEY,

NOMBRE VARCHAR(20) NOT NULL,

DESCRIPCION VARCHAR(100) NOT NULL

);

--//////////////////////////////////////////////////////////

-- DETALLE CASO DE USO CON CARGOS

--//////////////////////////////////////////////////////////

CREATE TABLE DETALLE\_CU

(

ID INT NOT NULL,

ID\_CU INT NOT NULL,

PRIMARY KEY(ID\_CU,ID),

IDCARGO INT NOT NULL,

HABILITADO BIT DEFAULT 1,

FOREIGN KEY(ID\_CU) REFERENCES CU(ID),

FOREIGN KEY(IDCARGO) REFERENCES CARGO(ID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

--//////////////////////////////////////////////////////////

--USUARIO PRINCIPAL

--//////////////////////////////////////////////////////////

CREATE TABLE PERSONA

(

ID INT PRIMARY KEY,

PASS VARCHAR(10) NOT NULL,

NOMBRE VARCHAR(50) NOT NULL,

NICK VARCHAR(15) NOT NULL,

EMAIL VARCHAR(50) NOT NULL,

ESTADO BIT DEFAULT 1,

IDCARGO INT NOT NULL,

FOREIGN KEY(IDCARGO) REFERENCES CARGO(ID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

--//////////////////////////////////////////////////////////

--BITACORA DEL ADMINISTRATIVO

--//////////////////////////////////////////////////////////

CREATE TABLE BITACORA

(

ID INT PRIMARY KEY,

INICIO DATETIME NOT NULL,

FIN DATETIME NOT NULL,

ID\_PERSONA INT NOT NULL,

FOREIGN KEY(ID\_PERSONA) REFERENCES PERSONA(ID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

--//////////////////////////////////////////////////////////

-- DETALLE BITACORA

--//////////////////////////////////////////////////////////

CREATE TABLE DETALLE\_BITACORA

(

ID INT NOT NULL,

ID\_BITACORA INT NOT NULL,

ACCION VARCHAR(50) NOT NULL,

HORA DATETIME NOT NULL,

TABLA VARCHAR(50) NOT NULL,

PRIMARY KEY(ID\_BITACORA,ID),

FOREIGN KEY(ID\_BITACORA) REFERENCES BITACORA(ID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

--//////////////////////////////////////////////////////////

--CURSOS CREADOS

--//////////////////////////////////////////////////////////

CREATE TABLE CURSO

(

ID INT PRIMARY KEY,

NOMBRE VARCHAR(50) NOT NULL,

DESCRIPCION VARCHAR(150) NOT NULL,

CONCLUIDO BIT DEFAULT 0,

PUNTAJE INT NOT NULL,

ID\_PERSONA INT NOT NULL,

ESTADO BIT DEFAULT 1,

FOREIGN KEY(ID\_PERSONA) REFERENCES PERSONA(ID)

);

--//////////////////////////////////////////////////////////

--NODO PRINCIPAL

--//////////////////////////////////////////////////////////

CREATE TABLE NODO

(

ID INT PRIMARY KEY,

TITULO VARCHAR(50) NOT NULL,

DESCRIPCION VARCHAR(255) NOT NULL,

APROBADO BIT DEFAULT 0,

HABILITADO BIT DEFAULT 0,

ID\_CURSO INT NOT NULL,

ESTADO BIT DEFAULT 1,

FOREIGN KEY(ID\_CURSO) REFERENCES CURSO(ID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

--//////////////////////////////////////////////////////////

---PREREQUISITO DE NODO

--//////////////////////////////////////////////////////////

CREATE TABLE PRE

(

ID\_NODO INT NOT NULL,

ID\_PRE INT NOT NULL,

PRIMARY KEY(ID\_PRE,ID\_NODO)

);

ALTER TABLE PRE

ADD FOREIGN KEY(ID\_NODO) REFERENCES NODO(ID)

ON UPDATE NO ACTION ON DELETE NO ACTION;

ALTER TABLE PRE

ADD FOREIGN KEY (ID\_PRE) REFERENCES NODO(ID)

ON UPDATE NO ACTION ON DELETE NO ACTION;

--//////////////////////////////////////////////////////////

--SUBNODOS DENTRO DEL NODO PRINCIPAL

--//////////////////////////////////////////////////////////

CREATE TABLE SUBNODO

(

ID INT PRIMARY KEY,

ID\_NODO INT NOT NULL,

TITULO VARCHAR(50) NOT NULL,

CONTENIDO VARCHAR(MAX) NOT NULL,

APROBADO BIT DEFAULT 0,

HABILITADO BIT DEFAULT 0,

PUNTAJE INT,

ESTADO BIT DEFAULT 1,

FOREIGN KEY(ID\_NODO) REFERENCES NODO(ID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

--//////////////////////////////////////////////////////////

---TIPO DE PREGUNTA (SELECCION MULTIPLE, ESCRIBIR, ETC)

--//////////////////////////////////////////////////////////

CREATE TABLE TIPO\_PREGUNTA

(

ID INT PRIMARY KEY,

NOMBRE VARCHAR(15) NOT NULL

);

--//////////////////////////////////////////////////////////

--RETROALIMENTACION DEL NODO

--//////////////////////////////////////////////////////////

CREATE TABLE PREGUNTA

(

ID INT PRIMARY KEY,

DESCRIPCION VARCHAR(20) NOT NULL,

PUNTAJE INT NOT NULL,

ID\_TIPO INT NOT NULL,

ID\_SUBNODO INT NOT NULL,

ESTADO BIT DEFAULT 1,

FOREIGN KEY(ID\_TIPO) REFERENCES TIPO\_PREGUNTA(ID)

ON UPDATE CASCADE

ON DELETE CASCADE,

FOREIGN KEY(ID\_SUBNODO) REFERENCES SUBNODO(ID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

--//////////////////////////////////////////////////////////

-- RESPUESTA DE LA PREGUNTA

--//////////////////////////////////////////////////////////

CREATE TABLE RESPUESTA

(

ID INT NOT NULL,

ID\_PREGUNTA INT NOT NULL,

PRIMARY KEY(ID\_PREGUNTA, ID),

DESCRIPCION VARCHAR(20) NOT NULL,

APROBADO BIT DEFAULT 0,

COMENTARIO VARCHAR(15) NOT NULL,

ID\_SUBNODO INT NOT NULL,

ESTADO BIT DEFAULT 1,

FOREIGN KEY(ID\_PREGUNTA) REFERENCES PREGUNTA(ID)

ON UPDATE CASCADE

ON DELETE CASCADE,

FOREIGN KEY(ID\_SUBNODO) REFERENCES SUBNODO(ID)

ON UPDATE NO ACTION

ON DELETE NO ACTION

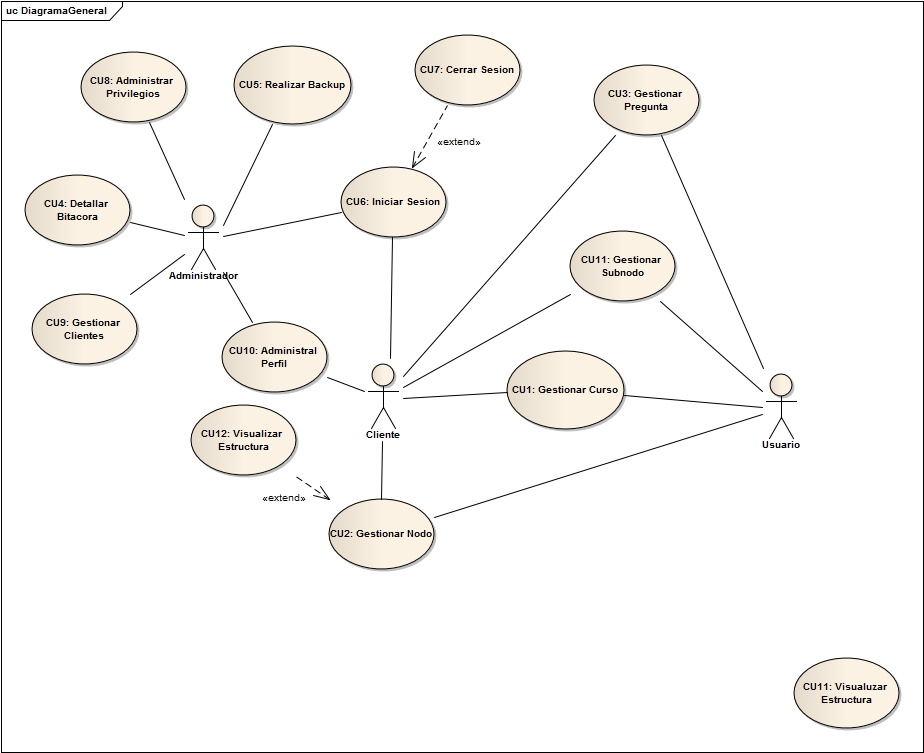
);

--//////////////////////////////////////////////////////////

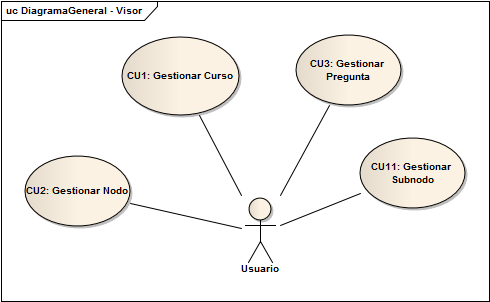
--------------INSERCIONES -----------------

# Estructurar Modelo de Casos de Uso

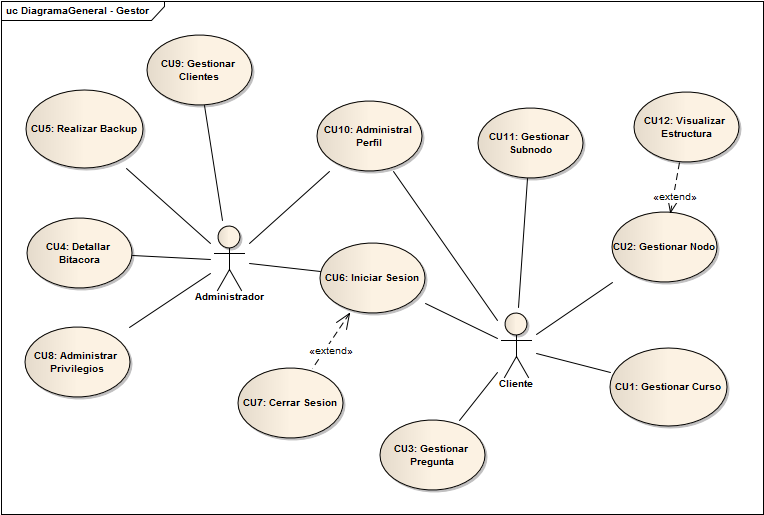
## Diagrama General



## Diagrama General Visor



## Diagrama General Gestor



# Bibliografia

<https://es.duolingo.com/>

<https://www.eldia.com.bo/mobile.php?c=Especiales&articulo=Microaprendizaje,-la-nueva-tecnica-&cat=389&pla=7&id_articulo=230537>