

RECOPIACIÓN Y CONSOLIDACIÓN EN UNA BASE DE DATOS DE LOS TRABAJOS  
DE GRADO EN EL ÁREA TIC DE LAS UNIVERSIDADES DE VILLAVICENCIO CON  
PROGRAMAS DE INGENIERÍA O TECNOLOGÍA DE SISTEMAS.

ALAIN ENRIQUE CONTRERAS CALIFA

YEISSON HERNAN GUIO FONSECA

Trabajo de grado realizado en la modalidad de investigación para optar al título de Tecnólogo en  
Informática.

ING MARIO LONDOÑO  
ASESOR

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS

VILLAVICENCIO

2017

**NOTA DE ACEPTACIÓN:**

---

---

---

---

---

---

---

---

---

**Firma Presidente del Jurado**

---

**Firma Jurado**

---

**Firma Jurado**

**Villavicencio, 21 de Abril de 2017**

## **DEDICATORIA**

A Dios, sin su ayuda no hubiera sido posible, a mis padres por su amor y apoyo incondicional en especial a mi papá que guiándome desde el cielo está, a todos mis hermanos.

**Yeisson Hernán Guío Fonseca**

Primero a Dios por estar ahí, siempre cuidándome en cada momento, mis hijos y esposa, a mis padres y hermanos por su apoyo y su amor.

**Alain Enrique Contreras Califa**

## **AGRADECIMIENTOS**

Infinitas gracias, primeramente a Dios, quien todo lo puede, quien nos ilumina cada día en nuestras tareas y quien nos otorga el don de la ciencia para cumplir a cabalidad nuestro proyecto de vida, sin él nada sería posible.

A nuestros padres, esposa e hijos, a los docentes que nos guiaron y acompañaron durante estos semestres, a la universidad que nos brindó la oportunidad y las herramientas para crecer como profesionales.

## TABLA DE CONTENIDO

INTRODUCCIÓN .....	12
RESUMEN .....	13
PLANTEAMIENTO DEL PROBLEMA .....	14
OBJETIVO GENERAL.....	16
OBJETIVOS ESPECÍFICOS.....	16
JUSTIFICACIÓN .....	17
MARCO REFERENCIAL.....	18
Marco contextual.....	18
META.....	19
VILLAVICENCIO .....	24
ANTECEDENTES .....	26
LÍNEA DE INVESTIGACIÓN .....	27
MARCO TEÓRICO.....	28
POSTGRE SQL.....	28
HTML5 (HyperText Markup Languaje, versión 5) .....	28
BOOTSTRAP.....	29
CSS (Cascading Stylesheets) .....	29
PHP.....	29
JAVASCRIPT .....	30
AJAX (Asynchronous JavaScript And Xml) .....	30
XAMPP .....	30
DIAGRAMAS UML.....	31
Diagramas de clases. ....	34
ELEMENTOS .....	34
Métodos:.....	36

BASES DE DATOS.....	41
Sistemas Gestores de Bases de Datos (SGBD (database management system)). .....	41
Bases de datos dinámicas: .....	42
Bases de datos estáticas:.....	42
Bases de datos transaccionales:.....	42
Bases de datos relacionales: .....	43
MARCO LEGAL .....	46
Ley 23 de 1982.....	46
La Ley 201 del 2012 o Ley Lleras 2.0. ....	46
METODOLOGÍA.....	49
Investigación aplicada.....	49
Enfoque cualitativo .....	49
POBLACIÓN Y MUESTRA.....	50
POBLACIÓN .....	50
MUESTRA.....	50
FUENTES DE INSTRUMENTACIÓN Y EVALUACIÓN .....	50
Observación.....	50
Entrevista.....	51
PROCEDIMIENTO.....	52
ANÁLISIS DE DATOS.....	53
DESARROLLO DE SOFTWARE .....	57
METODOLOGÍA DE DESARROLLO DE SOFTWARE.....	57
MODELO ESPIRAL .....	57
NORMALIZACIÓN BASE DE DATOS.....	61
DIAGRAMA RELACIONAL .....	62
TABLAS .....	63
DICCIONARIO DE DATOS .....	66
CREACIÓN DE LA BASE DE DATOS.....	68
DISEÑO DE LA INTERFAZ.....	72

CONCLUSIONES .....	108
RECOMENDACIONES.....	109
BIBLIOGRAFÍA .....	110
ANEXOS .....	112

## LISTA DE FIGURAS

**Figura 1.** Imagen Mapa de la República de Colombia.

**Figura 2.** Imagen mapa del departamento del Meta.

**Figura 3.** Imagen Mapa Satelital de la ciudad de Villavicencio.

**Figura 4.** Estructura de una clase, presentando sus divisiones, nombre, atributos y operaciones

**Figura 5.** Ejemplo de la clase cuenta.

**Figura 6.** Ejemplo herencia entre las clases, vehículo, camioneta y auto, con sus atributos y métodos.

**Figura 7.** Ejemplo de agregación con las clases, almacén, cuentas y clientes.

**Figura 8.** Ejemplo de asociación con las clases, almacén, cuentas y clientes.

**Figura 9.** Ejemplo dependencia.

**Figura 10. Tabulación respuestas pregunta** ¿Qué tanto registran la información de Proyectos del área de las Tecnologías de la Información y las Comunicaciones TIC?.

**Figura 11.** Gráfica de barras de las respuestas a la pregunta, ¿Qué tanto registran la información de Proyectos del área de las Tecnologías de la Información y las Comunicaciones TIC?

**Figura 12. Tabulación de las respuestas a la pregunta** ¿Ha tenido inconvenientes al consultar la información referente a Proyectos TIC?

**Figura 13.** Gráfica de barras de la pregunta ¿Ha tenido inconvenientes al consultar la información referente a Proyectos TIC?

**Figura 14.** Gráfica de barras de la pregunta, Si respondió Si a la anterior pregunta, indique en donde se guarda esta información.

**Figura 15.** Gráfica de barras de la pregunta, Si respondió Si a la anterior pregunta, indique en donde se guarda esta información.

**Figura 16.** Tabulación de respuestas a la pregunta ¿En cuál de las siguientes líneas se clasifican los proyectos TIC que se han realizado en la Institución?

**Figura 17.** Gráfica de barras de las respuestas a la pregunta ¿En cuál de las siguientes líneas se clasifican los proyectos TIC que se han realizado en la Institución?

**Figura 18.** Modelo de desarrollo espiral.

**Figura 19.** Modelo entidad relación.

**Figura 20.** Modelo Relacional

**Figura 21.** Tabla Líneas

**Figura 22.** Tabla proyectos

**Figura 23.** Tabla universidades

**Figura 24.** Tabla usuarios

**Figura 25.** Tabla Rol

**Figura 26.** Interfaz inicio de sesión.

**Figura 27.** Interfaz rol Administrador.

**Figura 28.** Interfaz Registro e ingreso de proyectos TIC.

**Figura 29.** Interfaz consulta proyectos TIC.

**Figura 30.** Interfaz resultado consulta proyectos TIC

**Figura 31.** Interfaz ver un proyecto - proyectos TIC

**Figura 32.** Interfaz Registro de usuario proyectos TIC

**Figura 33.** Interfaz Consulta de usuario proyectos TIC

## **LISTA DE TABLAS**

**Tabla 1.** Formato recolección de datos proyectos TIC.

**Tabla 2.** Diccionario de datos proyectos

**Tabla 3.** Diccionario de datos tabla universidades.

**Tabla 4.** Diccionario de datos tabla usuarios.

**Tabla 5.** Tabla Informe con fines de publicación RAE.

## INTRODUCCIÓN

El presente trabajo se encuentra enmarcado en la recopilación de los diferentes trabajos de grado desarrollados en el marco de tecnología, desarrollo de software y afines en las instituciones de educación superior con programas de ingeniería de sistemas o programas afines a las TIC, en la ciudad de Villavicencio Meta.

Este trabajo se desarrolló con acompañamiento y es un producto del semillero de investigación Orinoco TIC de la Corporación Universitaria Minuto de Dios.

Es necesario conocer cómo va nuestra región en cuanto a desarrollo de proyectos TIC y su respectivo enfoque, por este motivo se diseñó un formato de consulta general (ver anexo 1) de los diferentes proyectos TIC de las universidades, a las cuales se consultarán en sus respectivos repositorios, (sistemas de información, medios digitales, libros de grado, documentos de informe final de proyectos). La información que se consultó de cada proyecto fue: Título del proyecto, programa, año de sustentación, autor, objetivo del proyecto, objetivos específicos del proyecto y resumen.

Teniendo en cuenta la información consultada en las diferentes universidades, con programas de ingeniería de sistemas o programas afines a las TIC, en la ciudad de Villavicencio, se consolidó en una base de datos, donde a través de una interfaz web, se pueden consultar algunas temáticas de desarrollo y la información de todos los proyectos TIC.

## RESUMEN

Actualmente no se cuenta con un registro de las diferentes líneas de investigación en el desarrollo de proyectos de grado de las universidades de Villavicencio en las facultades de ingeniería de sistemas.

En este trabajo se referencian algunos de los proyectos de TI elaborados y presentados como opción de grado, por los estudiantes de las diferentes universidades privadas y públicas de la ciudad de Villavicencio, en las cuales se desarrolló carrera en el área de ingeniería de sistemas, desarrollo de software, tecnología en redes y afines al área.

Se desarrolla esta tarea visitando cada una de las instituciones mencionadas (Ver tabla 1) en el trabajo de campo, donde se consultó en los diferentes repositorios y bibliotecas de las universidades que con ayuda del semillero de investigación de la Corporación Universitaria Minuto de Dios Orinoco TIC, que facilitó acceder a la información de los trabajos de grado presentados en las Universidades de Villavicencio en el Departamento del Meta, en las áreas de tecnología, desarrollo de software y afines.

Con el desarrollo de este trabajo se dará a conocer los tipos de proyectos y su respectivo enfoque de desarrollo que sobresalen en las universidades de la ciudad de Villavicencio, estos datos darán también una pequeña noción en cuanto a cómo va nuestra región en el desarrollo de proyectos de software en las universidades como opción de grado. De igual manera servirá a futuro como base para formular otros proyectos.

## PLANTEAMIENTO DEL PROBLEMA

En Villavicencio, día a día se establecen diferentes centros educativos de educación superior y acoge un buen número de estudiantes allegados del departamento del Meta, se vienen desarrollando importantes proyectos entre las universidades, el sector público y privado donde el sector de las Tecnologías de la información y las Comunicaciones es uno de los focos para el desarrollo de estos proyectos patrocinados por el gobierno nacional, como los son los proyectos desarrollados en CODALTEC, cofinanciada por el sector público, ministerio de defensa nacional, la gobernación del meta, y la alcaldía de Villavicencio.

(<http://www.codaltec.com/?q=es/nosotros>)

Actualmente no se cuenta con información alguna de cómo se encuentran las universidades en cuanto al desarrollo de proyectos y en especial aquellas que ofertan programas de Ingeniería de sistemas o programas afines a las TIC (Tecnologías de la información y las comunicación).

Teniendo en cuenta esta situación y que es de suma importancia conocer cómo se encuentran las diferentes universidades de Villavicencio en cuanto a desarrollo de proyectos TIC, el grupo de trabajo toma la opción de consultar en las diferentes universidades de la ciudad todo lo referente en cuanto a desarrollo de proyectos de grado orientados a las Tecnologías de la Información y la Comunicación.

## **FORMULACIÓN DEL PROBLEMA**

¿La implementación de la base de datos con la información de los proyectos TIC consultados en las universidades de la ciudad de Villavicencio, serviría como fuente de consulta a los estudiantes y docentes como referenciación de lineamientos de desarrollo de software y proyectos de grado?

## **OBJETIVO GENERAL**

Elaborar una base de datos con los trabajos de grado en el área TIC (Tecnologías de la información y las comunicaciones) en las universidades con programas de ingeniería o tecnología de sistemas de la ciudad de Villavicencio, Meta.

## **OBJETIVOS ESPECÍFICOS**

- Recolectar información de los proyectos de grado que han sido presentados en las instituciones de educación superior con programas de ingeniería de sistemas o afines a las TIC en la ciudad de Villavicencio.
- Normalizar en una base de datos la información obtenida, que genere consultas mediante una aplicación web.

## JUSTIFICACIÓN

La recopilación y consolidación de los datos de los diferentes proyectos TIC (Tecnologías de la información y la comunicación) en las instituciones de educación superior con programas de ingeniería de sistemas o programas afines a las TIC, en la ciudad de Villavicencio Meta, es importante para hacer un análisis de cómo se encuentran actualmente las universidades de la ciudad de Villavicencio en cuanto a desarrollo de software y propuestas innovadoras. Desarrollar este trabajo es muy útil, ya que esta información es de mucha relevancia para la UNIMITO porque le proporciona una idea de la orientación de los proyectos desarrollados por otras instituciones y esto le ayudaría aún más a orientar y motivar a sus estudiantes a desarrollar ideas nuevas de desarrollo de software y evitar la redundancia de los proyectos formulados por los estudiantes de tecnología en desarrollo de software.

De no implementarse el desarrollo de esta tarea, dejaría a la universidad sin la oportunidad de poder comparar los enfoques de desarrollo de software e ideas y proyectos de grado, y de esta manera poder mejorar las propias e ir un paso más adelante.

Mediante visitas programadas a los repositorios de las instituciones de educación superior con programas de ingeniería de sistemas o programas afines a las Tic, y un formato elaborado para recolectar los datos de los diferentes proyectos se consultará y recopilará toda la información sobre Proyectos Tic que se encuentren en estos repositorios, para luego organizarla y normalizarla en una base de datos donde se podrá consultar los diferentes proyectos recopilados así como sus lineamientos.

## MARCO REFERENCIAL

### Marco contextual

### Colombia

**Figura 1.** Imagen Mapa de la República de Colombia.



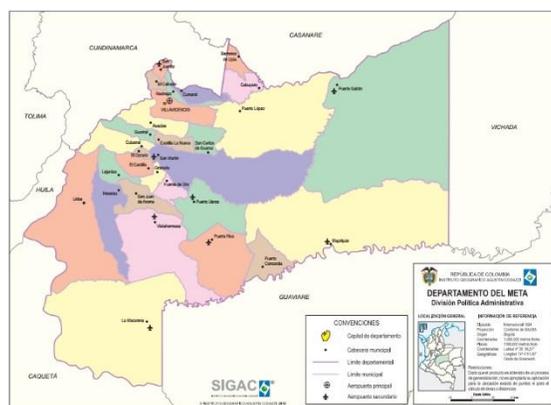
**Fuente:** Catalogo de logística documental Nacional

Oficialmente República de Colombia, es un país situado en la región noroccidental de América del Sur. Está constituido en un estado unitario, social y democrático de derecho cuya forma de gobierno es presidencialista. Esta república está organizada políticamente en 32 Departamentos descentralizados y el Distrito capital de Bogotá, sede del gobierno nacional.

Incluyendo Isla de Malpelo, Cayo Roncador, y Banco Serrana, el país abarca una superficie de 1 141 748 km, Reclama como mar territorial el área hasta las 12 millas náuticas de distancia, manteniendo un diferendo limítrofe al respecto con Venezuela y Nicaragua. Limita al este con Venezuela y Brasil, al sur con Perú y Ecuador y al noroeste con Panamá; en cuanto a límites marítimos, colinda con Panamá, Costa Rica, Nicaragua, Honduras, Jamaica, Haití, República Dominicana y Venezuela en el mar Caribe y con Panamá, Costa Rica y Ecuador en el océano Pacífico.

## META

**Figura 2.** Imagen mapa del departamento del Meta



**Fuente:** SIGAC

Localizado en la región central del país, el Meta, con 85 mil kilómetros cuadrados de extensión, es decir 8 millones 563.000 hectáreas, es uno de los departamentos con mayor crecimiento en los últimos tiempos, especialmente en lo que tiene que ver con productos para la generación de biocombustibles y la seguridad alimentaria.

Por tradición, el Meta ha sido reconocido como la despensa agrícola de Colombia y el mayor surtidor de alimentos de Bogotá, surtiendo el 48% de la demanda.

El volumen y la calidad han sido fundamentales para cumplir con las exigencias del mercado capitalino y, sobretodo, de los consumidores, que han mostrado gran preferencia por estos productos, especialmente plátano, arroz, cítricos y frutas, hortalizas y carne.

### **Altillanura**

El Meta cuenta con suelos fértiles en gran parte de su territorio, pero tiene una gran extensión de tierras en la sabana, conocida como Altillanura, ubicadas entre Puerto López y Puerto Gaitán, que, con mejora de suelos, posee un gran potencial para la producción avícola y porcícola, la industria de concentrados, forestales y biombustibles.

La Altillanura cuenta con una extensión superior a los tres millones de hectáreas para incorporar a la producción nacional, sin afectar el medio ambiente, ampliando de esta manera la frontera agrícola.

Después de Brasil, Colombia es el único país de Suramérica con sabanas, o llanuras, para intervenir sin que se requiera la destrucción forestal, y la mayor parte están en el Meta, hecho que pone al Departamento en una situación de privilegio, porque se está generando desarrollo agrícola, industrial y social, acompañado de generación de empleo.

## **Generalidades**

### **Geografía**

#### Localización

Se encuentra localizado en la región centro-oriental de Colombia entre  $1^{\circ}32'30''$  y los  $4^{\circ}57'30''$  lat. Norte y entre los  $71^{\circ}2'30''$  y  $74^{\circ}57'00''$  long. Oeste de Greenwich.

### **Límites**

Norte: Cundinamarca y Casanare

Sur: Caquetá y Guaviare

Oriente: Vichada

Occidente: Huila y Bogotá, Distrito Capital.

### **Aspectos fisiográficos**

Presenta gran diversidad topográfica con terrenos que van desde llanuras -un 80% de su territorio- hasta importantes elevaciones como el cerro El Nevado con 4.560 m sobre el nivel del mar y el páramo de Sumapaz con 4.000 m-. Se encuentra La Sierra de la Macarena, uno de los macizos más antiguos del mundo y con una gran biodiversidad en flora y fauna.

## **Economía**

La principal fuente de riqueza del Meta es la explotación de hidrocarburos, seguida de la ganadería de extensión y la agricultura con cultivos tecnificados de arroz, cacao, forestales, caña y palma africana, además de cultivos tradicionales de yuca, plátano, cítricos, frutas, ente otros.

Cuenta también con minas de sal en Cumaral y Upia.

El territorio está formado por tres grandes regiones fisiográficas:

La primera, ubicada en el occidente del Departamento, la constituye la parte montañosa representada por el flanco oeste de la cordillera Oriental, con alturas que alcanzan los 4.000 m sobre el nivel del mar, en límites con los departamentos de Caquetá, Huila y Cundinamarca. El piedemonte o área de transición entre la cordillera, la llanura, y la Serranía de La Macarena, ubicada esta última en forma casi perpendicular a la cordillera Oriental.

La segunda unidad fisiográfica es la planicie, con alturas que no sobrepasan los 200 m sobre el nivel del mar en el centro y oriente del Departamento.

La tercera corresponde a la zona sur del Departamento y cubre alrededor del 60% de su territorio. Su fisiografía varía desde las tierras de páramo y laderas de la Cordillera Oriental hasta las vegas y planicies cálidas de los ríos Ariari y Guaviare. Allí confluyen grandes áreas del Sistema de Parques Nacionales Naturales: PNN Páramo de Sumapaz, PNN Cordillera de los Picachos, PNN Tinigua y PNN Serranía de la Macarena; comprende además, los paisajes de altillanura con vegetación de sabana y bosque primario.

## **Altura**

La altura sobre el nivel del mar oscila entre 125 y 4000 metros.

## Temperatura

La temperatura promedio es de 30°C.

## Hidrografía

Entre los ríos más importantes se destacan: Meta, Ariari, Guayabero, Guayariba, Manacacías, Metica, Guamal, Humadea, Yucao, Upía y Humea.

Las lagunas más importantes son: Lomalinda, El Porvenir, La Conquista, San Vicente Santaya y Remanzón.

Regiones del Meta (nuevo estudio). Producto de un estudio realizado en el año, aprobado por los diputados de la Asamblea departamental, el Meta se divide en seis subregiones, con el fin de lograr una mejor planeación y hacer más eficiente la inversión de recursos.

- Río Meta. Conformada por Puerto Gaitán, Puerto López, Cabuyaro y Barranca de Upía.
- Capital Cordillera. Villavicencio, San Juanito, El Calvario, Restrepo y Cumaral.
- Bajo Ariari. Mapiripán, Puerto Concordia y Puerto Rico
- Alto Ariari Centro. Acacías, Guamal, San Carlos de Guaroa, Castilla La Nueva, Cubarral y

El Dorado

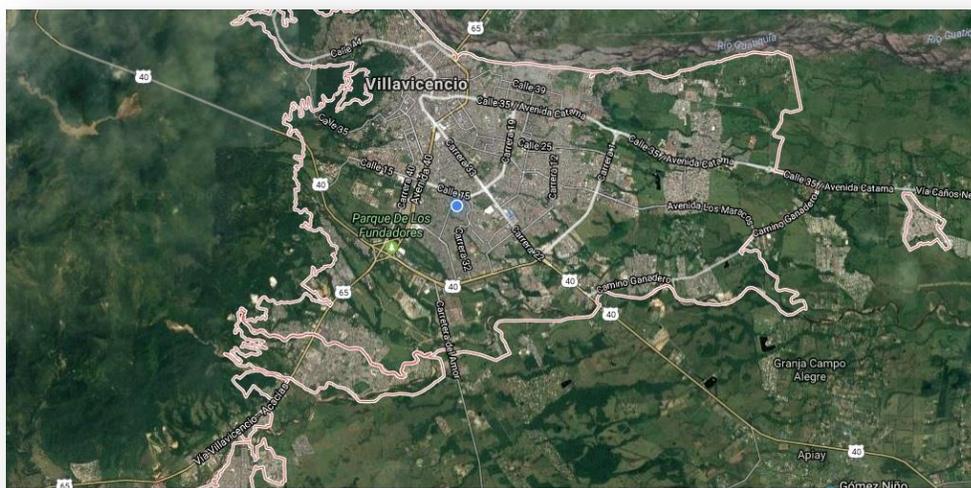
- Sub Región Ariari. Granada, Fuentedeoro, El Castillo, Puerto Lleras, San Juan de Arama, San Martín y Lejanías
- Macarena. Uribe, Mesetas, Vista Hermosa y La Macarena

## Demografía

La población del departamento del Meta para el año 2016, de acuerdo a las proyecciones del censo hecho por el Departamento Administrativo Nacional de Estadística, DANE, en 1993, es de 979.710 habitantes, distribuidos en 29 municipios, en su mayoría situados a lo largo del piedemonte llanero. (Gobernación del Meta (Web Site), Nuestra Tierra, 2017)

## VILLAVICENCIO

**Figura 3.** Imagen Mapa Satelital de la ciudad de Villavicencio.



**Fuente:** Google Maps.

Villavicencio es una ciudad y corredor principal de comercio para el Departamento y la Orinoquia. Está ubicada en el piedemonte de la cordillera oriental al Noroccidente del Departamento del Meta, en la margen izquierda del rio Guatiquia. Fundada el 6 de abril de 1840, cuenta con una población urbana aproximada de 486.363 habitantes en 2016. Presenta un clima

cálido y muy húmedo, con temperaturas medias de 28° C. Por pertenecer a la región de la Orinoquía de mayor perspectiva para el desarrollo agroindustrial y minero del país, Villavicencio se perfila como una ciudad que será un futuro indispensable para el desarrollo económico de la Orinoquia y del país.

Villavicencio, como la futura metrópoli del departamento del Meta tiene la tarea de ir a la vanguardia de otras ciudades capitales del país en cuanto a innovación y desarrollo de ideas y propuestas de nuevas economías como las que se dan en el desarrollo de las tecnologías de la información y comunicación.

En la ciudad actualmente existen ya varias universidades que forman la gran academia y malla de educación superior en el departamento, cuentan con un gran número de estudiantes de diferentes facultades que le apuestan al desarrollo y competitividad regional.

Actualmente en el marco del desarrollo urbano de la ciudad, se siguen construyendo sedes y sucursales de universidades ya presentes en la región y otras que llegan a la ciudad, con el fin de brindar un mejor servicio a la población llanera.

En el auge en el país de las Tecnologías de la Información y la Comunicación (TIC) y con el apoyo del gobierno nacional, se vienen desarrollando actividades y promociones en cuanto a desarrollo tecnológico en las principales regiones del país, y por supuesto en la ciudad de Villavicencio en donde hay gran oferta en cuanto a programas referentes de las TIC en las principales instituciones de educación superior.

## ANTECEDENTES

Recientemente en la Corporación Universitaria Minuto de Dios se formuló y desarrolló un prototipo de software para la administración del proceso del anteproyecto, informe final y socialización para los proyectos de grado en general de todas las facultades de la universidad.

De igual manera el semillero de Investigación de Orinoco Cluster TIC, en cabeza de la Ingeniera y especialista en gerencia de proyectos Nydia Maritza Gachancipa, ha formulado un proyecto para el Sistema de información del Banco de proyectos Tic para el departamento del Meta, el cual contó con la aprobación de la administración gubernamental del departamento, siendo este uno los proyectos formulados y aprobados por la dirección de investigación de la gobernación del departamento del Meta.

El proyecto recopilará la información y documentación de los diversos proyectos que tienen las Instituciones de educación superior de la ciudad de Villavicencio, en un primer momento para su almacenamiento en una base de datos y posterior consulta por parte de estudiantes, docentes y público en general que requiera el uso de esta información.

## **LÍNEA DE INVESTIGACIÓN**

Este trabajo se enmarca, y sigue la línea de investigación: INNOVACIONES SOCIALES Y PRODUCTIVAS. En el desarrollo de este trabajo se sigue este lineamiento porque se considera que el proyecto planteado es algo nuevo y necesario para la comunidad académica de la Universidad Minuto de Dios.

La sublínea de Investigación que en que se basa este trabajo es: DESARROLLO DE SOFTWARE Y TELEINFORMÁTICA.

## MARCO TEÓRICO

Los siguientes son los conceptos teóricos en los que se fundamenta este trabajo:

### POSTGRE SQL

Es un sistema de gestión de base de datos relacional orientado a objetos y libre, publicado bajo licencia PostgreSQL, como muchos otros proyectos de código abierto, el desarrollo de postgres no es manejado por una empresa o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, libre o apoyados por organizaciones comerciales ( PostgreSQL Global Development Group, 2017)

Se utilizó PostgreSql, ya que es un potente motor de base de datos, robusto, libre con el cual se normalizó y se creó la base de datos en donde se alojó la información recolectada.

### HTML5 (HyperText Markup Lenguaje, versión 5)

HTML es el lenguaje de marcado estándar para la creación de páginas Web.

HTML significa Hyper Text Markup Language

HTML describe la estructura de las páginas Web usando marcado elementos HTML son los bloques de construcción de páginas HTML.

Los elementos HTML están representados por etiquetas

Los navegadores no muestran las etiquetas HTML, pero las usan para representar el contenido de la página.(W3C, 2017).

## **BOOTSTRAP**

Es un framework o conjunto de herramientas de código abierto para diseños de sitios y aplicaciones web. Contiene Plantillas de diseño con tipografía, formularios, cuadros, menús de navegación y otros elementos de diseño basados en HTML y CSS. (GitHub 2017)

Se implementa esta tecnología para el maquetado y el diseño visual de la aplicación.

## **CSS (Cascading Stylesheets)**

Hojas de estilo en cascada, describe cómo los elementos HTML se van a mostrar en la pantalla, papel, o en otros medios. CSS ahorra mucho trabajo, con CSS se puede controlar el diseño de varias páginas web a la vez. (W3C, 2017).

Se usó para establecer el diseño visual de la aplicación web donde se generan las consultas de trabajos de grado.

## **PHP**

PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página web resultante.(W3C, 2017).

## JAVASCRIPT

A veces abreviado como JS, es un lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, más conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador, tales como nod.js o apache couchDB. Es un lenguaje script multi-paradigma, basado en prototipos, dinámico, soporta estilos de programación funcional, orientada a objetos e imperativa. (Mozilla Foundation, 2017).

## AJAX

AJAX = **A** síncrono **J** avascript **A** nd **X**ML.

AJAX no es un lenguaje de programación, este permite que las páginas web se actualicen de forma asincrónica mediante el intercambio de datos con un servidor Web detrás de las escenas. Esto significa que es posible actualizar partes de una página web, sin necesidad de recargar la página entera. (W3C, 2017).

## XAMPP

Es un servidor de plataforma libre, es un software que integra en una sola aplicación, un servidor web Apache, intérpretes de lenguaje de scripts PHP, un servidor de base de datos MySQL, un servidor de FTP FileZilla, el popular administrador de base de datos escrito en PHP, MySQL, entre otros módulos.

XAMPP es un paquete formado por un servidor web Apache, una base de datos MySQL y los intérpretes para los lenguajes PHP y Perl. De hecho su nombre viene de ahí, X (para cualquier sistema operativo), A (Apache), M (MySQL), P (PHP) y P (Perl). -XAMPP solamente requiere descargar y ejecutar un archivo .zip, .tar, o .exe, con unas pequeñas configuraciones en alguno de sus componentes que el servidor Web necesitará. XAMPP.

Una de las características sobresalientes de este sistema es que es multiplataforma, es decir, existen versiones para diferentes sistemas operativos, tales como: Microsoft Windows, GNU/Linux, Solaris, y MacOS X. Existen versiones para Linux (testado para SuSE, RedHat, Mandrake y Debian), Windows (Windows 98, NT, 2000, XP y Vista), MacOS X y Solaris (desarrollada y probada con Solaris 8, probada con Solaris 9).

Por esto y por muchas más razones, este software es el que se ha elegido para implementar las bases de datos, ya que ofrece seguridad y un fácil manejo del mismo. (Medina, 2014).

## **DIAGRAMAS UML**

El lenguaje de modelado unificado (UML - Unified Modeling Language) facilita varios tipos de diagramas, lo que permite describir los requisitos, funcionalidad, y otros conceptos relativos a un proyecto de desarrollo de software.

Estos diagramas se pueden organizar en dos grupos:

Los que describen el comportamiento del negocio, del sistema, de un aspecto en particular.

- Diagrama de Actividad (Activity Diagram): Representa los procesos de negocio o la lógica de un sistema complejo. Incluye, opcionalmente, el flujo de datos. el nivel de abstracción suele ser bastante alto, pero pueden realizarse diagramas de actividad exploratorios cuando la lógica que se trata es compleja. (Kendall y Kenneth, 2005).
- Diagrama de Estados (State Machine Diagram): Describe los estados de un objeto así como la transición entre estados. Muy útil para los desarrolladores.
- Diagrama de Casos de Uso (Use Case Diagram): Muestra casos de uso individuales, actores y las relaciones entre ellos. El Proceso Unificado dice está dirigido por los

casos de uso, esto significa que este diagrama (en el nivel de abstracción que sea) es la base del lenguaje de modelado y representación.

- Diagrama de Comunicación (Communication Diagram): Muestra las relaciones entre instancias de las clases y el flujo de mensajes entre ellas, antes (UML 1.0) se llamaba Diagrama de Colaboración. La cuestión tiene que ser realmente complicada para tener que utilizar estos diagramas.
- Diagrama de Interacción (Interaction Overview Diagram): Es una variante del Diagrama de Actividad, muestra un panorama general del flujo de control dentro del sistema o proceso de negocio.
- Diagrama de Secuencia (Sequence Diagram): Muestra la secuencia de la lógica, el orden en que se suceden los mensajes. Importante, especialmente cuando se trabaja en ambientes altamente compartidos.
- Diagrama de Tiempo (Timing Diagram): Muestra el cambio de estado de un objeto a través del tiempo en respuesta a eventos externos.
- Diagrama de Clases (Class Diagram): Muestra una colección de clases, sus tipos, sus contenidos y sus relaciones. Importantísimo representa el modelo de datos, y en consecuencia su persistencia en alguna forma de almacenamiento. (Kendall y Kenneth, 2005, Pag,664).
- Diagrama de Estructura (Composite Structure Diagram): Muestra la estructura interna de una clase, componente o caso de uso. Especialmente debe indicar los puntos de interacción con otras partes del sistema.

- Diagrama de Componentes (Component Diagram): Describe los elementos que componen un sistema. Debe detallar los elementos o componentes, las interacciones y relaciones así como las interfaces públicas.
- Diagrama de Despliegue (Deployment Diagram): Muestra la arquitectura de ejecución de un sistema. Incluye nodos, entornos de hardware y software.
- Diagrama de Objetos (Object Diagram): Describe los objetos y sus relaciones en algún momento. Generalmente se usa en casos especiales para diagramas de clase o de comunicaciones.
- Diagrama de Paquetes (Package Diagram): Describe como los elementos del modelo se organizan en "paquetes", debe indicar la dependencia entre paquetes. (Kendall y Kenneth, 2005, Pag 665).

UML se ha establecido como el estándar en la industria de desarrollo de software. Es cierto que puede utilizarse otro tipo de lenguaje, pero eso reduce la cantidad de personas que pueden leer (entender) el desarrollo.

“No debe confundirse el lenguaje de modelado con la metodología del proceso de desarrollo. El lenguaje de modelado, es el conjunto de símbolos y reglas que utilizamos para comunicarnos; la metodología del proceso de desarrollo es una guía de actividades que pueden seguirse en el ciclo de vida de un proyecto de desarrollo de software. De hecho la metodología incluye un lenguaje.” (Booch, Rumbaugh y Jacobso, 2006).

- **Diagramas de clases.** Sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de contenido. (Kendall y Kenneth, 2005)

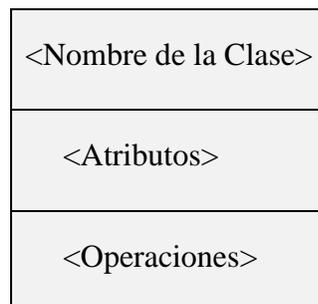
Un diagrama de clases está compuesto por los siguientes elementos:

- Clase: atributos, métodos y visibilidad.
- Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

## ELEMENTOS

**Clase:** es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.). (Kendall y Kenneth, 2005, Pag.658).

**Figura 4.** Estructura de una clase, presentando sus divisiones, nombre, atributos y operaciones.



**Fuente:** Modelo de clases, Universidad de Chile.

División superior: Contiene el nombre de la clase.

División media: Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).

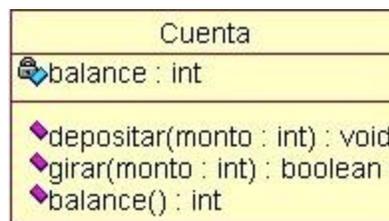
División inferior: Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public). (Modelo UML UdChile, 2017).

Ejemplo: Una Cuenta Corriente que posee como característica:

Balance: Puede realizar las operaciones de: Depositar Girar y Balance

El diseño asociado es: Atributos y Métodos:

**Figura 5.** Ejemplo de clase cuenta.



**Fuente.** Modelo de clases, Universidad de Chile.

### Atributos:

Los atributos o características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno, estos son:

**public:** Indica que el atributo será visible tanto dentro como fuera de la clase, es decir, es accesible desde todos lados.

**private:** Indica que el atributo sólo será accesible desde dentro de la clase (sólo sus métodos lo pueden acceder).

**protected:** Indica que el atributo no será accesible desde fuera de la clase, pero si podrá ser accesado por métodos de la clase además de las subclases que se deriven (ver herencia).

### Métodos:

Los métodos u operaciones de una clase son la forma en como ésta interactúa con su entorno, éstos pueden tener las características:

**public:** Indica que el método podrá ser invocado tanto dentro como fuera de la clase, es decir, es accesible desde todos lados.

**private:** Indica que el método sólo será accesible desde dentro de la clase (sólo sus métodos lo pueden acceder).

**protected:** Indica que el método no será accesible desde fuera de la clase, pero si podrá ser accesado por métodos de la clase además de las subclases que se deriven

**Relaciones entre Clases** ahora ya definido el concepto de Clase, es necesario explicar cómo se pueden interrelacionar dos o más clases (cada uno con características y objetivos diferentes).

Antes es necesario explicar el concepto de cardinalidad de relaciones: En UML, la cardinalidad de las relaciones indica el grado y nivel de dependencia, se anotan en cada extremo de la relación y éstas pueden ser:

**uno o muchos:** 1..\* (1..n)

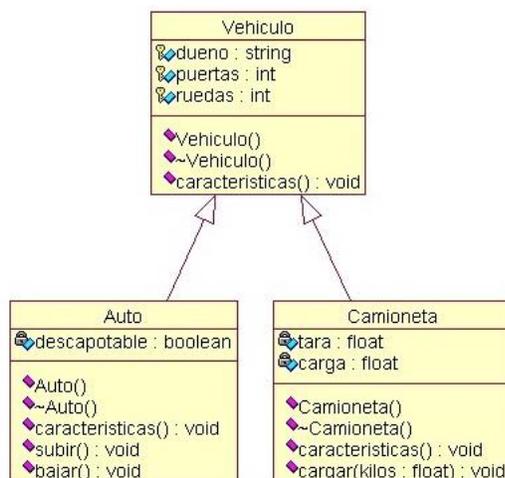
**o muchos:** 0..\* (0..n)

**número fijo:** m (m denota el número). (Modelo UML UdChile, 2017).

### Herencia (Especialización/Generalización):

Indica que una subclase hereda los métodos y atributos especificados por una Super Clase, por ende la Subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Super Clase (**public y protected**), ejemplo:

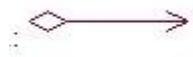
**Figura 6.** Ejemplo herencia entre las clases, vehículo, camioneta y auto, con sus atributos y métodos.



**Fuente:** Modelo de clases, Universidad de Chile.

En la figura se especifica que Auto y Camioneta heredan de Vehículo, es decir, Auto posee las Características de Vehículo (Dueño, Puertas, etc) además posee algo particular que es descapotable, en cambio Camioneta también hereda las características de Vehículo (Puertas, Ruedas, etc) pero posee como particularidad propia Acoplado, Tara y Carga.

Cabe destacar que fuera de este entorno, lo único "visible" es el método Características aplicable a instancias de Vehículo, Auto y Camión, pues tiene definición pública, en cambio atributos como descapotable no son visibles por ser privados.



Agregación:

Para modelar objetos complejos n, bastan los tipos de datos básicos que proveen los lenguajes: enteros, reales y secuencias de caracteres. Cuando se requiere componer objetos que son instancias de clases definidas por el desarrollador de la aplicación, tenemos dos posibilidades:

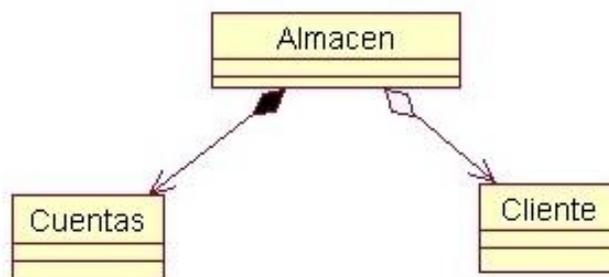
Por Valor: Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido está condicionado por el tiempo de vida del que lo incluye. Este tipo de relación es comúnmente llamada Composición (el Objeto base se construye a partir del objeto incluido, es decir, es "parte/todo").(Modelo UML UdChile, 2017).

**Por Referencia:** Es un tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye. Este tipo de relación es comúnmente llamada

**Agregación** (el objeto base utiliza al incluido para su funcionamiento).

Un Ejemplo es el siguiente:

**Figura 7.** Ejemplo de agregación con las clases, almacén, cuentas y clientes.



**Fuente:** Modelo de clases, Universidad de Chile.

En donde se destaca que:

Un Almacén posee Clientes y Cuentas (los rombos van en el objeto que posee las referencias).

Cuando se destruye el Objeto Almacén también son destruidos los objetos Cuenta asociados, en cambio no son afectados los objetos Cliente asociados.

La composición (por Valor) se destaca por un rombo relleno.

La agregación (por Referencia) se destaca por un rombo transparente.

La flecha en este tipo de relación indica la navegabilidad del objeto referenciado. Cuando no existe este tipo de particularidad la flecha se elimina.

 Asociación:

La relación entre clases conocida como Asociación, permite asociar objetos que colaboran entre sí. Cabe destacar que no es una relación fuerte, es decir, el tiempo de vida de un objeto no depende del otro.

Ejemplo:

**Figura 8.** Ejemplo de asociación con las clases, almacén, cuentas y clientes.



**Fuente:** Modelo de clases, Universidad de Chile.

Un cliente puede tener asociadas muchas Órdenes de Compra, en cambio una orden de compra solo puede tener asociado un cliente.

Dependencia o instanciación  (Uso):

Representa un tipo de relación muy particular, en la que una clase es instanciada (su instanciación es dependiente de otro objeto/clase). Se denota por una flecha punteada.

El uso más particular de este tipo de relación es para denotar la dependencia que tiene una clase de otra, como por ejemplo una aplicación grafica que instancia una ventana (la creación del Objeto Ventana está condicionado a la instanciación proveniente desde el objeto Aplicación):

**Figura 9.** Ejemplo dependencia.



**Fuente:** Modelo de clases, Universidad de Chile

Cabe destacar que el objeto creado (en este caso la Ventana gráfica) no se almacena dentro del objeto que lo crea (en este caso la Aplicación).

## **BASES DE DATOS**

Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. (Bases de datos, 2017)

### **Sistemas Gestores de Bases de Datos (SGBD (database management system)).**

Permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos DBMS, así como su utilización y administración, se estudian dentro del ámbito de la informática.

Las bases de datos pueden clasificarse de varias maneras, de acuerdo al contexto que se esté manejando, la utilidad de las mismas o las necesidades que satisfagan. (Bases de datos, 2017).

**Bases de datos dinámicas:** Son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización, borrado y edición de datos, además de las operaciones fundamentales de consulta. Un ejemplo, puede ser la base de datos utilizada en un sistema de información de un supermercado. (Bases de datos, 2017).

**Bases de datos estáticas:** Son bases de datos únicamente de lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones, tomar decisiones y realizar análisis de datos.(C:J Date, 2017)

**Bases de datos transaccionales:** Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible, por lo tanto la redundancia y duplicación de información no es un problema como con las demás bases de datos, por lo general para poderlas aprovechar al máximo permiten algún tipo de conectividad a bases de datos relacionales.(C.J. Date, 2017).

Un ejemplo habitual de transacción es el traspaso de una cantidad de dinero entre cuentas bancarias. Normalmente se realiza mediante dos operaciones distintas, una en la que se debita el

saldo de la cuenta origen y otra en la que se acreditó el saldo de la cuenta destino. Para garantizar la atomicidad del sistema (es decir, para que no aparezca o desaparezca dinero), las dos operaciones deben ser atómicas, es decir, el sistema debe garantizar que, bajo cualquier circunstancia (incluso una caída del sistema), el resultado final es que, o bien se han realizado las dos operaciones, o bien no se ha realizado ninguna. (C.J Date, 2017).

**Bases de datos relacionales:** Este es el modelo **utilizado en la actualidad** para representar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que esta es la teoría de las bases de datos relacionales creadas por Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por *registros* (las filas de una tabla), que representarían las tuplas, y *campos* (las columnas de una tabla). (C.J Date, 2017).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, Structured Query Language o Lenguaje Estructurado de Consultas, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos. (C.J Date, 2017).

**Bases de datos Orientadas a Objetos:** Este modelo, bastante reciente, y propio de los modelos informáticos orientados a objetos, trata de almacenar en la base de datos los *objetos* completos (estado y comportamiento).

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

Encapsulación - Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.

Herencia - Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.

Polimorfismo - Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o signatura) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o parámetros). La implementación (o método) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones.

SQL:2003, es el estándar de SQL92 ampliado, soporta los conceptos orientados a objetos y mantiene la compatibilidad con SQL92. (C.J Date, 2017).

## **MARCO LEGAL**

En Colombia rigen ya, varias leyes sobre el derecho de autor.

### **Ley 23 de 1982**

“Artículo 1º—Los autores de obras literarias, científicas y artísticas gozarán de protección para sus obras e la forma prescrita por la presente ley y, en cuanto fuere compatible con ella, por el derecho común. También protege esta ley a los intérpretes o ejecutantes, a los productores de fonogramas y a los organismos de radiodifusión, en sus derechos conexos a los del autor”

### **La Ley 201 del 2012 o Ley Lleras 2.0.**

“La ley nace y es aprobada como un compromiso de Colombia con Estados Unidos al aprobar el Tratado de libre Comercio Su artículo 13, en el que se prohíbe la retransmisión de señales por Internet sin autorización del titular, es bastante polémico y evidencia que los que la aprobaron no tienen el conocimiento necesario de lo que es la convergencia digital” (Ley Lleras, 2016)

### **Acuerdos que se reglamentan en las universidades.**

**Artículo 71 de la Constitución política de Colombia.** “La búsqueda del conocimiento y la expresión artística son libres. Los planes de desarrollo económico y social incluirán el formato a las ciencias y, en general, a la cultural. El estado creará incentivos para personas e instituciones que desarrollen y fomenten la ciencia y la tecnología y las demás manifestaciones culturales y ofrecerá estímulos especiales a personas e instituciones que ejerzan esta actividades.” (Consulta de la Norma, 2016).

**La ley 29 de febrero de 1990** “le otorga al estado colombiano la responsabilidad de promover y orientar el adelanto científico y tecnológico, lo obliga a incorporar la Ciencia y Tecnología a los planes y programas de desarrollo económico y social del país además de formular planes a formular planes de Ciencia y Tecnología tanto para el mediano como para el largo plazo. (Consulta de la Norma, 2016).

**Decreto 393 del 26 de febrero de 1991** “Por el cual se dictan normas sobre Asociación para actividades científicas y tecnológicas, proyectos de investigación y creación de tecnologías. (Consulta de la Norma, 2016).

**Decreto 585 del 26 de febrero de 1991** “Por el cual se crea el consejo nacional de ciencia y tecnología, se organiza el instituto colombiano para el desarrollo del a ciencia y tecnología Colciencias y se dictan otras disposiciones. (Consulta de la Norma, 2016).

**Decreto 591 del 26 de febrero de 1991** “Por el cual se regulan las modalidades específicas de contratos de fomento de actividades científicas y tecnológicas. El cual autoriza a las entidades descentralizadas del orden nacional para celebrar contratos de financiamiento destinados para actividades científicas y tecnológicas. (Consulta de la Norma, 2016).

El reglamento estudiantil de la Corporación Universitaria Minuto de Dios, centro regional Villavicencio en el capítulo XV enuncia el derecho a titulación del alumnado mediante los siguientes artículos.

**Capítulo XV titulación.**

**Artículo 83 Egresado.** “Se considera egresado el estudiante que ha usado y aprobado en su totalidad el plan de estudios de un programa académico y solo le falta cumplir con los demás requisitos obtenidos para optar el título. (Consulta de la norma, 2016).

**Artículo 84 Derecho al título.** “El alumno regular que ha terminado y aprobado todas las asignaturas de su programa, de acuerdo con el plan de estudio correspondiente, tiene derecho a recibir, previos los requisitos de grado de acuerdo con los artículos 85 y 86, el título que ofrece la corporación universitaria. (Consulta de la norma, 2016).

## METODOLOGÍA

### **Investigación aplicada.**

Se tomó como parte inicial la problemática que necesita ser intervenida, se inició con la descripción detallada y delimitada del problema, luego se evalúa el problema y se escoge una metodología aceptada dando como resultado la solución a dicha necesidad. La solución debe incluir los conocimientos de los desarrolladores y los lineamientos de los procesos y procedimientos.

### **Enfoque cualitativo.**

De esta forma la Corporación Universitaria Minuto de Dios permite y da a los estudiantes los medios, formas y herramientas, para realizar investigación y de esta forma generar servicios o productos que impacten a la sociedad, generando conocimiento y aportes valiosos.

De igual manera es un enfoque que se encuentra enmarcado en las líneas y sublíneas de investigación, establecidos por la Corporación Universitaria Minuto de Dios, sirviendo como soporte para que los estudiantes tenga enmarcados los pasos a seguir y la forma en cómo se debe estructurar cada proyecto.

## **POBLACIÓN Y MUESTRA**

### **POBLACIÓN**

Se establece como población las universidades de la ciudad de Villavicencio públicas y privadas que oferten programas de ingeniería de sistemas o programas afines a las tecnologías de la información y la comunicación siendo cinco (5) instituciones las que cumplen estos requerimientos.

### **MUESTRA**

Son 5 unidades de investigación correspondientes a los 5 bibliotecarios de las universidades: Corporación Universitaria Minuto de Dios, Universidad Cooperativa de Colombia, Corporación Universitaria del Meta, Corporación Universitaria Autónoma de Nariño y Universidad de los Llanos.

## **FUENTES DE INSTRUMENTACIÓN Y EVALUACIÓN**

### **Observación.**

El instrumento de la observación se utilizó para obtener la información pertinente de todos los proyectos de TI (Tecnologías de la información), en las diferentes universidades consultadas, identificando los lineamientos de desarrollo, el autor (es), título del proyecto, objetivo general, objetivos específicos, programa, resumen. Para esto se realizó el siguiente formato:.

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS	
RECOPIACIÓN Y CONSOLIDACIÓN DE TRABAJOS DE GRADO EN EL ÁREA TIC EN LAS UNIVERSIDADES CON PROGRAMAS DE INGENIERÍA O TECNOLOGÍA DE SISTEMAS DE LA CIUDAD DE VILLAVICENCIO – META.	
UNIVERSIDAD	
TÍTULO PROYECTO	
FACULTAD	
AÑO	
AUTOR (ES)	
OBJETIVO GENERAL	
OBJETIVOS ESPECÍFICOS	
RESUMEN	

**Tabla 1.** Formato recolección de datos proyectos TIC.

## Entrevista

Se realizó entrevista a los bibliotecarios encargados de cada institución, donde se solicitó que describiera el proceso que lleva cada proyecto de grado ya realizado en la biblioteca, (almacenamiento, clasificación) y también se les solicitó que con qué frecuencia y qué cantidad de proyectos recibía para archivar y si tenían un sistema de información en donde se pudieran consultar.

## **Encuesta**

Aplicada a bibliotecarios o personal a cargo del recepcionamiento y almacenamiento de los proyectos de grado en las universidades.

El formato de la encuesta se evidencia en los anexos del documento.

## **PROCEDIMIENTO**

Se utilizó una muestra de 5 universidades en la ciudad en donde se consultaron los repositorios de las bibliotecas los diferentes proyectos TIC formulados por estudiantes como opción de grado, lo cual servirá como insumo importante para este proyecto.

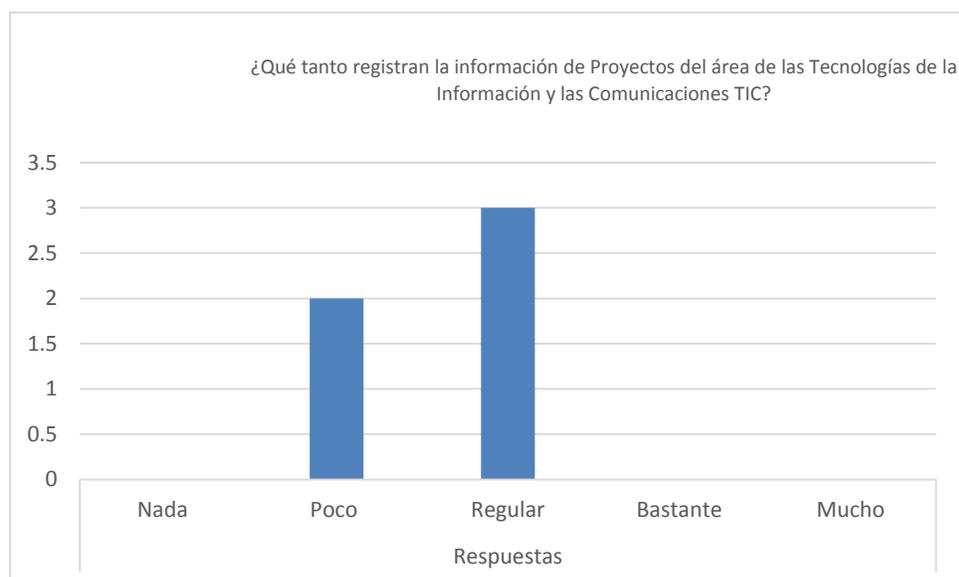
Para la consulta y búsqueda de la información requerida se solicitó con el apoyo del semillero de investigación Orinoco TIC, a cada institución su aprobación y disposición para el desarrollo de esta tarea, la cual se logró llevar a cabo en 4 meses (noviembre de 2016, enero, febrero y marzo de 2017).

## ANÁLISIS DE DATOS

### Pregunta 1.

Respuestas				
Nada	Poco	Regular	Bastante	Mucho
0	2	3	0	0

**Figura 10. Tabulación respuestas pregunta** ¿Qué tanto registran la información de Proyectos del área de las Tecnologías de la Información y las Comunicaciones TIC?



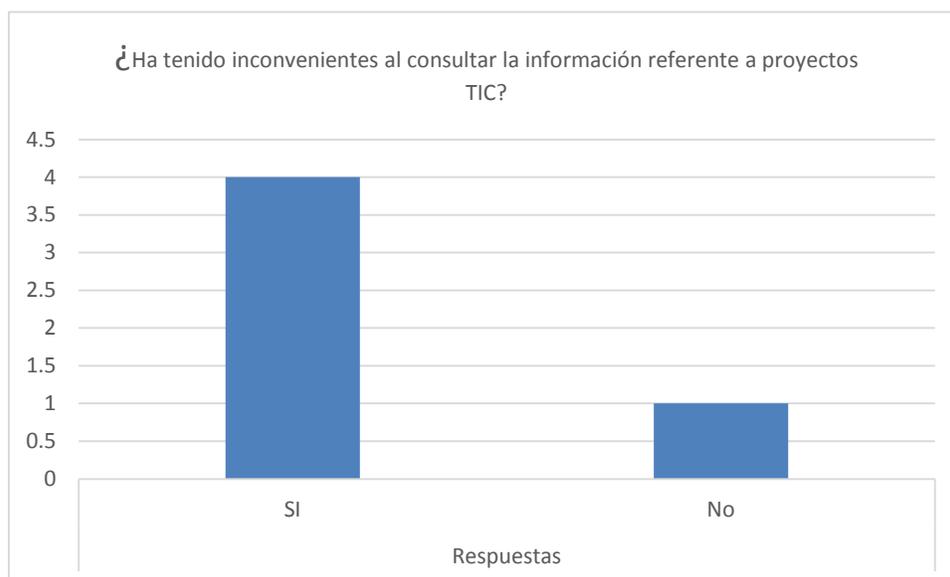
**Figura 11.** Gráfica de barras de las respuestas a la pregunta, ¿Qué tanto registran la información de Proyectos del área de las Tecnologías de la Información y las Comunicaciones TIC?

**Interpretación:** La gráfica representa que en los repositorios encuestados el registro de proyectos orientados a TI, se registran regularmente y en los otros el registro de este tipo de proyectos es muy poco. Al interpretar el gráfico se evidencia que es bajo el registro de proyectos de grado en los repositorios de las diferentes universidades.

Pregunta 2.

Respuestas	
SI	No
4	1

**Figura 12.** Tabulación de las respuestas a la pregunta ¿Ha tenido inconvenientes al consultar la información referente a Proyectos TIC?



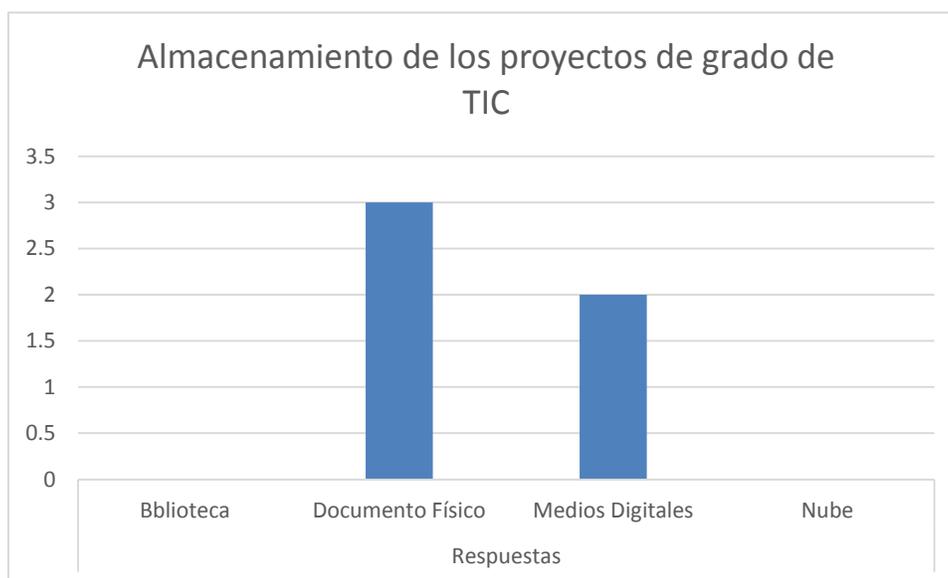
**Figura 13.** Gráfica de barras de la pregunta ¿Ha tenido inconvenientes al consultar la información referente a Proyectos TIC?

**Interpretación:** La gráfica de barras representa que actualmente hay dificultad para acceder a la información de los proyectos presentados en las diferentes universidades.

## Pregunta 3.

Respuestas			
Biblioteca	Documento Físico	Medios Digitales	Nube
0	3	2	0

**Figura 14.** Gráfica de barras de la pregunta, Si respondió Si a la anterior pregunta, indique en donde se guarda esta información.



**Figura 15.** Gráfica de barras de la pregunta, Si respondió Si a la anterior pregunta, indique en donde se guarda esta información.

**Interpretación:** Se puede evidenciar en la gráfica que aun 3 de las 5 bibliotecas consultadas guardan sus trabajos de grado en medios físicos como libros y documentos.

## Pregunta 5.

Respuestas	Formulación y gestión de proyectos TI	3
	Desarrollo de software	4
	Multimedia y contenidos web	5
	Ciberseguridad y seguridad de datos	0
	Antivirus	1
	Geo posicionamiento y geo localización	0
	Aplicaciones móviles	4
	Modelado y simulación	2
	Consultoría Gobierno en Línea	0
	Infraestructura telecomunicaciones	4
	Soporte tecnológico	3
	Electrónica	2
	Aplicaciones drones	0
	Capacitación	4
	Equipamiento de oficina y mobiliario	4
Teletrabajo	0	

**Figura 16.** Tabulación de respuestas a la pregunta ¿En cuál de las siguientes líneas se clasifican los proyectos TIC que se han realizado en la Institución?.



**Figura 17.** Gráfica de barras de las respuestas a la pregunta ¿En cuál de las siguientes líneas se clasifican los proyectos TIC que se han realizado en la Institución?.

**Interpretación:** la gráfica representa las líneas de clasificación de proyectos en TI donde los desarrollos, la mayoría son orientados a la web.

## DESARROLLO DE SOFTWARE

### METODOLOGÍA DE DESARROLLO DE SOFTWARE

**MODELO ESPIRAL** en el desarrollo del software es un modelo meta del ciclo de vida del software donde el esfuerzo del desarrollo es iterativo, tan pronto culmina un esfuerzo del desarrollo por ahí mismo comienza otro; además en cada ejecución del desarrollo se sigue cuatro pasos principales:

1. Determinar o fijar los objetivos. En este paso se definen los objetivos específicos para posteriormente identifica las limitaciones del proceso y del sistema de software, además se diseña una planificación detallada de gestión y se identifican los riesgos.

2. Análisis del riesgo. En este paso se efectúa un análisis detallado para cada uno de los riesgos identificados del proyecto, se definen los pasos a seguir para reducirlos y luego del análisis de estos, se planean estrategias alternativas.

3. Desarrollar, verificar y validar. En este tercer paso, después del análisis de riesgo, se elige un paradigma para el desarrollo del sistema de software y se lo desarrolla.

4. Planificar. En este último paso es donde el proyecto se revisa y se toma la decisión si se debe continuar con un ciclo posterior al de la espiral. Si se decide continuar, se desarrollan los planes para la siguiente fase del proyecto. ( Barry Boehm, 2017).

Con cada iteración alrededor de la espiral, se crean sucesivas versiones del software, cada vez más completas y, al final, el sistema de software ya queda totalmente funcional.

La diferencia principal entre el modelo espiral y los demás modelos (ej.: cascada, evolutivo, incremental, etc.) es la evaluación del riesgo. El riesgo es todo lo que pueda salir mal en un proyecto de desarrollo de software. Por ejemplo, si se quiere utilizar un lenguaje de programación para desarrollar un sistema operativo, un riesgo posible es que los compiladores utilizables no produzcan un código objeto eficiente. Los riesgos originan problemas en el proyecto, como el exceso de los costos. Es así que, la disminución de los riesgos es una actividad muy importante. Un modelo espiral comienza con la determinación de los objetivos tanto funcionales como de rendimiento. Después se enumeran algunas formas posibles de alcanzar estos objetivos identificando las fuentes de riesgos posibles. Luego se continuó con el siguiente paso que es resolver estos riesgos y llevar a cabo las actividades de desarrollo, para finalizar con la planificación del siguiente ciclo de la espiral. . ( Barry Boehm, 2017).

### **Características Del Modelo En Espiral Para El Desarrollo De Software**

Es considerado como un modelo evolutivo ya que combina el modelo clásico con el diseño de prototipos.

- Contiene una nueva etapa que es el análisis de riesgos, no incluida anteriormente.
- Este modelo es el indicado para desarrollar software con diferentes versiones actualizadas como se hace con los programas modernos de PC's.
- La ingeniería puede desarrollarse a través del ciclo de vida clásico o el de construcción de prototipos.

- Este es el enfoque más realista actualmente.
- El modelo en espiral esta compartida en varias actividades estructurales, también llamadas regiones de tareas. Existen seis regiones de tareas que son:
- **Comunicación con el cliente:** esta es una tarea requerida para establecer comunicación entre el desarrollador y el cliente.
- **Planificación:** esta tarea es necesaria aplicarla para poder definir los recursos, el tiempo y otras informaciones relacionadas con el proyecto, es decir, son todos los requerimientos.
- **Análisis de riesgos:** esta es una de las tareas principales por lo que se aplica el modelo en espiral, es requerida para evaluar los riesgos técnicos y otras informaciones relacionadas con el proyecto.
- **Ingeniería:** esta es una tarea necesaria ya que se requiere construir una o más representaciones de la aplicación.
- **Construcción y adaptación:** esta tarea es requerida en el modelo espiral porque se necesita construir, probar, instalar y proporcionar soporte al usuario.
- **Evaluación el cliente:** esta también es una tarea principal, necesaria para adquirir la reacción del cliente según la evaluación de las representaciones del software creadas durante la etapa de ingeniería y la de implementación creada durante la etapa de instalación. . ( Barry Boehm, 2017).

### Ventajas Del Modelo Espiral

- No requiere una definición completa de los requerimientos del software a desarrollar para comenzar su funcionalidad.
- En la terminación de un producto desde el final de la primera iteración es muy factible aprobar los requisitos.
- Sufrir retrasos corre un riesgo menor, porque se comprueban los conflictos presentados tempranamente y existe la forma de poder corregirlos a tiempo.

### Desventajas Del Modelo Espiral

- Existe complicación cuando se evalúa los riesgos.
- Se requiere la participación continua por parte del cliente.
- Se pierde tiempo al volver producir inicialmente una especificación completa de los requerimientos cuando se modifica o mejora el software. . ( Barry Boehm, 2017).



**Figura 18.** Modelo de desarrollo espiral. **Fuente:** (Google/Images/2017)

## NORMALIZACIÓN BASE DE DATOS

### Diagrama Entidad Relación

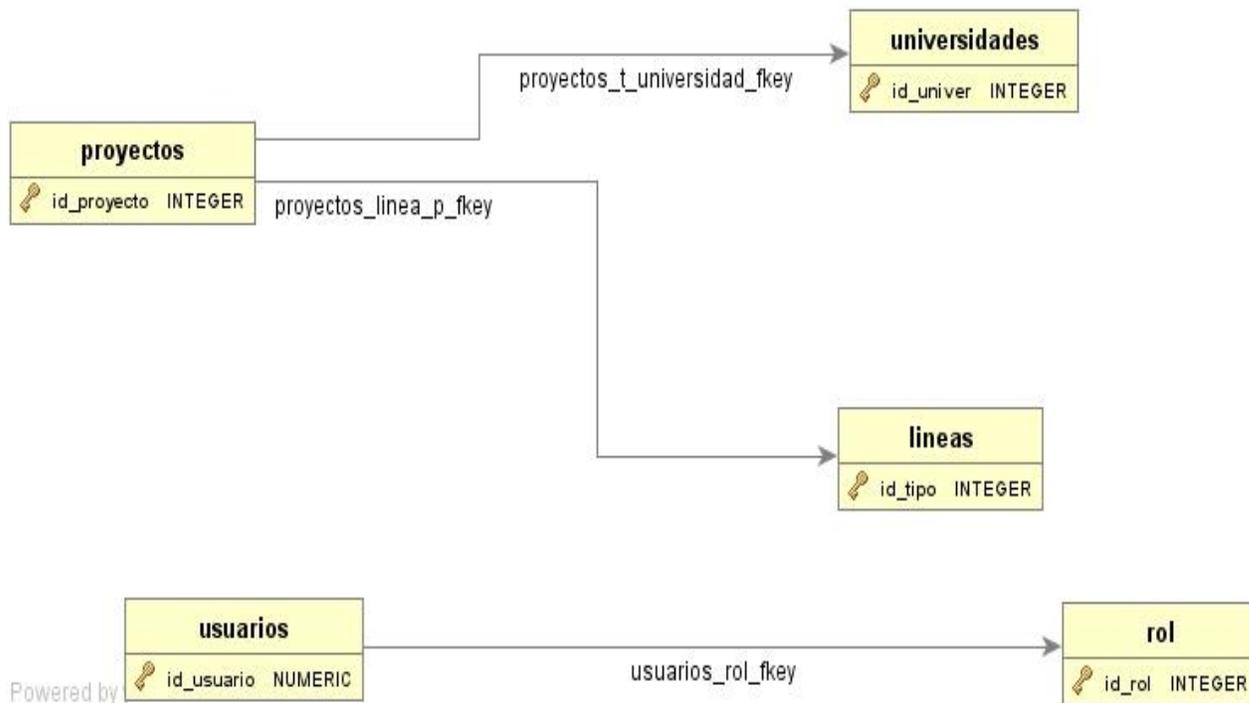
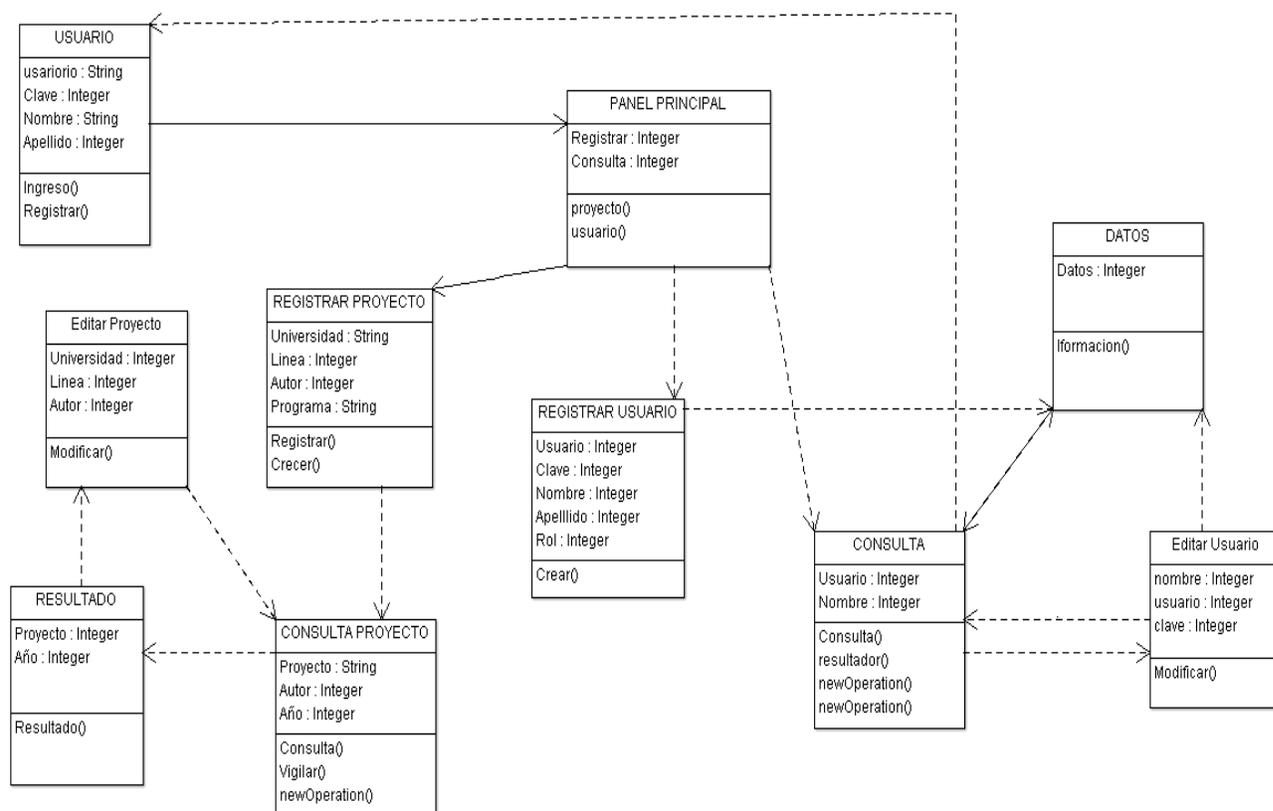


Figura 19 Modelo entidad relación.

**Fuente:** Diagrama elaborado por los desarrolladores del proyecto.

**Descripción:** En la figura se contempla el modelo entidad relación de la Base de Datos, en donde se ven las relaciones de la tabla proyectos que contiene la información de los datos recolectados, con la tabla líneas que contiene la información de los lineamientos de desarrollo y la relación también con la tabla universidades la cual almacena las instituciones consultadas para este trabajo. El modelo ER, también muestra la relación de la tabla de los posibles usuarios y sus roles en el sistema.

## DIAGRAMA RELACIONAL



**Figura 20.** Modelo Relacional

**Fuente:** Diagrama elaborado por los desarrolladores del proyecto.

**Descripción:** La figura muestra la relación que existe entre las tablas de la base de datos y la forma en que funciona el sistema de información partiendo desde el usuario y su rol, para así registrar, consultar, ingresar, borrar y modificar un la información de los proyectos.

## TABLAS

**Figura 21.** Tabla Líneas

<b>Id_tipo: INTEGER</b>
<b>Tipo: Character Varying</b>

**Fuente:** Tabla elaborada por los desarrolladores del proyecto.

**Descripción:** Esta tabla permite almacenar las líneas de desarrollo, en los que fueron enfocados los diferentes trabajos de grado.

**Figura 22.** Tabla proyectos.

<b>Id_proyecto: INTEGER</b>
<b>Título: VARCHAR</b>
<b>Programa: VARCHAR</b>
<b>Fecha: DATE</b>
<b>Autores: VARCHAR</b>
<b>Objetivo: general: VARCHAR</b>
<b>Objetivo específico: VARCHAR</b>
<b>Resumen: VARCHAR</b>
<b>Observación: VARCHAR</b>
<b>Universidad; INTEGER</b>
<b>Línea_p: INTEGER</b>

**Fuente:** Tabla elaborada por los desarrolladores del proyecto.

**Descripción:** En esta tabla se almacena toda la información tomada de los proyectos TIC, como el título del proyecto, el programa de la universidad donde se dio el proyecto, los autores que lo formularon y desarrollaron, los objetivos general y específicos, resumen, también

contiene un campo para observaciones del proyecto y una llave foránea a la tabla línea de desarrollo.

**Figura 23.** Tabla universidades

<b>Id_UNIVER:</b> <b>INTEGER</b>
<b>n_universidad:</b> <b>VARCHAR</b>

**Fuente:** Tabla elaborada por los desarrolladores del proyecto.

**Descripción:** Esta tabla permite almacenar la información de la universidad a la que pertenece el proyecto.

**Figura 24.** Tabla usuarios.

<b>Id_usuario: INTEGER</b>
<b>usuario: TEXT</b>
<b>Clave: VARCHAR</b>
<b>Nombre: VARCHAR</b>
<b>Apellido: VARCHAR</b>
<b>Fecha Creación: DATE</b>
<b>Rol: INTEGER</b>

**Fuente:** Tabla elaborada por los desarrolladores del proyecto.

**Descripción:** Esta tabla almacena los usuarios que pueden acceder al sistema de información. Contiene también la clave de cada usuario, fecha de creación y una llave foránea a la tabla rol, la cual proporciona los permisos.

**Figura 25.** Tabla Rol.

<b>Id_rol: Serial</b>
<b>Rol: VARCHAR</b>

**Fuente:** Tabla elaborada por los desarrolladores del proyecto.

**Descripción:** Esta tabla almacena el rol y sus correspondientes permisos para cada usuario en el sistema de información.

## DICCIONARIO DE DATOS

PROYECTOS				
NOMBRE	TIPO DE DATO	TAMAÑO	RANGO	DESCRIPCIÓN
Título	Varchar	100	0,9A-Z,a-z	almacena título del proyecto
id_proyecto	Integer	2	0,9A-Z,a-z	Almacena número de identificación del proyecto
fecha	Date	1	0,9-/	Almacena la fecha de creación del proyecto
Objetivo General	Varchar	300	0,9A-Z,a-z	Almacena el nombre del objetivo general del proyecto
Objetivo Especifico	Varchar	cada	0,9A-Z,a-z	Almacena los objetivos específicos de cada proyecto
Resumen	Varchar	900	0,9A-Z,a-z	Almacena el resumen de cada proyecto
Observación	Varchar	400	0,9A-Z,a-z	Almacena observaciones o notas sobre cada proyecto
Universidad	Integer	1	0,9A-Z,a-z	Almacena número de universidad
línea	Integer	1	0,9A-Z,a-z	Almacena el número de la línea de desarrollo

**Tabla 2.** Diccionario de datos proyectos

**Fuente:** Tabla elaborada por los desarrolladores del proyecto.

Universidades				
NOMBRE	TIPO DE DATO	TAMAÑO	RANGO	DESCRIPCIÓN
n_universidad	Varchar	100	0,9A-Z,a-z	almacena nombre de la universidad
id_univer	Integer	1	0,9A-Z,a-z	Almacena número de identificación de la universidad

**Tabla 3.** Diccionario de datos tabla universidades

**Fuente:** Tabla elaborada por los desarrolladores del proyecto.

Usuarios				
NOMBRE	TIPO DE DATO	TAMAÑO	RANGO	DESCRIPCIÓN
id_usuario	Integer	1	0,9	Almacena número de identificación del usuario
usuario	text	20	0,9A-Z,a-z	Almacena el alias del usuario
Clave	Varchar	15	0,9A-Z,a-z	Almacena la clave de usuario
nombre	Varchar	50	0,9A-Z,a-z	Almacena el nombre del usuario
apellido	Varchar	50	0,9A-Z,a-z	Almacena el apellido del usuario
fecha_creacion	date	8	0,9-/	Almacena la fecha de creación del usuario
Rol	Varchar	20	0,9A-Z,a-z	Almacena el rol, de si es admin o invitado

**Tabla 4.** Diccionario de datos tabla usuarios.

**Fuente:** Tabla elaborada por los desarrolladores del proyecto.

## CREACIÓN DE LA BASE DE DATOS

Estructura de la tabla Líneas:

```
-- Table: líneas
```

```
- DROP TABLE líneas;
```

```
CREATE TABLE líneas
```

```
(
```

```
  id_tipo integer NOT NULL DEFAULT nextval("equipo_tipo_Id_tipo_seq"::regclass),
```

```
  tipo character varying NOT NULL,
```

```
  CONSTRAINT equipo_tipo_pkey PRIMARY KEY (id_tipo)
```

```
)
```

```
WITH (
```

```
  OIDS=FALSE
```

```
);
```

```
ALTER TABLE líneas
```

```
  OWNER TO postgres;
```

**Fuente:** Código PL/pgSQL- Tablas elaborado por los desarrolladores del proyecto.

Estructura de la tabla proyectos:

```
-- Table: proyectos
```

```
-- DROP TABLE proyectos;
```

```
CREATE TABLE proyectos
```

```
(
```

```
id_proyecto integer NOT NULL DEFAULT nextval("Equipos_id_pc_seq"::regclass),
```

```
titulo character varying(1000) NOT NULL,
```

```
programa character varying(100),
```

```
fecha character varying(20),
```

```
autores character varying(100),
```

```
objetivo_general character varying(1000) NOT NULL,
```

```
objetivo_especificos character varying(1000) NOT NULL,
```

```
resumen character varying(800) NOT NULL,
```

```
observacion character varying(600),
```

```
t_universidad integer,
```

```
linea_p integer,
```

```
CONSTRAINT "Equipos_pkey" PRIMARY KEY (id_proyecto),
```

```
CONSTRAINT proyectos_linea_p_fkey FOREIGN KEY (linea_p)
```

```
REFERENCES lineas (id_tipo) MATCH SIMPLE
```

```
ON UPDATE NO ACTION ON DELETE NO ACTION,
```

```
CONSTRAINT proyectos_t_universidad_fkey FOREIGN KEY (t_universidad)
```

```
REFERENCES universidades (id_univer) MATCH SIMPLE
```

```
ON UPDATE NO ACTION ON DELETE NO ACTION
```

```

)
WITH (
  OIDS=FALSE
);
ALTER TABLE proyectos
  OWNER TO postgres;

```

**Fuente:** Código PL/pgSQL- Tablas elaborado por los desarrolladores del proyecto

Estructura de la tabla rol:

```

-- Table: rol
-- DROP TABLE rol;
CREATE TABLE rol
(
  id_rol serial NOT NULL,
  rol character varying(15),
  CONSTRAINT rol_pkey PRIMARY KEY (id_rol)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE rol
  OWNER TO postgres;

```

**Fuente:** Código PL/pgSQL- Tablas elaborado por los desarrolladores del proyecto.

Estructura de la tabla universidades:

```
-- Table: universidades
```

```
-- DROP TABLE universidades;
```

```
CREATE TABLE universidades
```

```
(
```

```
id_univer integer NOT NULL DEFAULT nextval('equipo_marca_id_marca_seq'::regclass),
```

```
n_universidad character varying,
```

```
CONSTRAINT equipo_marca_pkey PRIMARY KEY (id_univer)
```

```
)
```

```
WITH (
```

```
  OIDS=FALSE
```

```
);
```

```
ALTER TABLE universidades
```

```
  OWNER TO postgres;
```

**Fuente:** Código PL/pgSQL- Tablas elaborado por los desarrolladores del proyecto.

## DISEÑO DE LA INTERFAZ

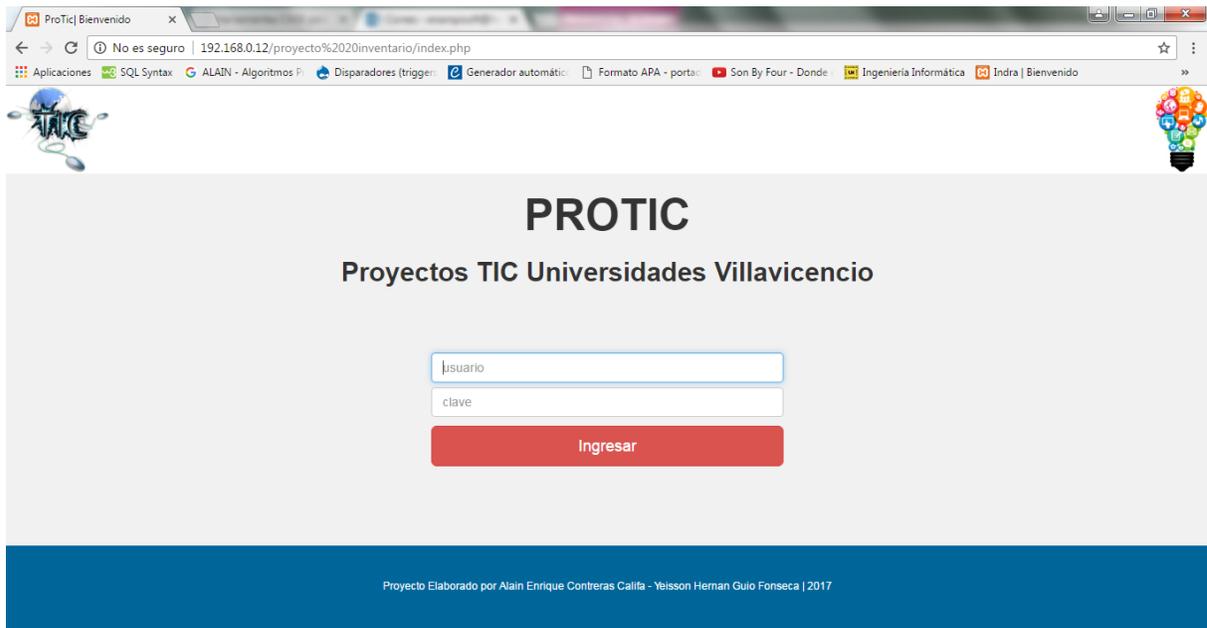


Figura 26. Interfaz inicio de sesión.

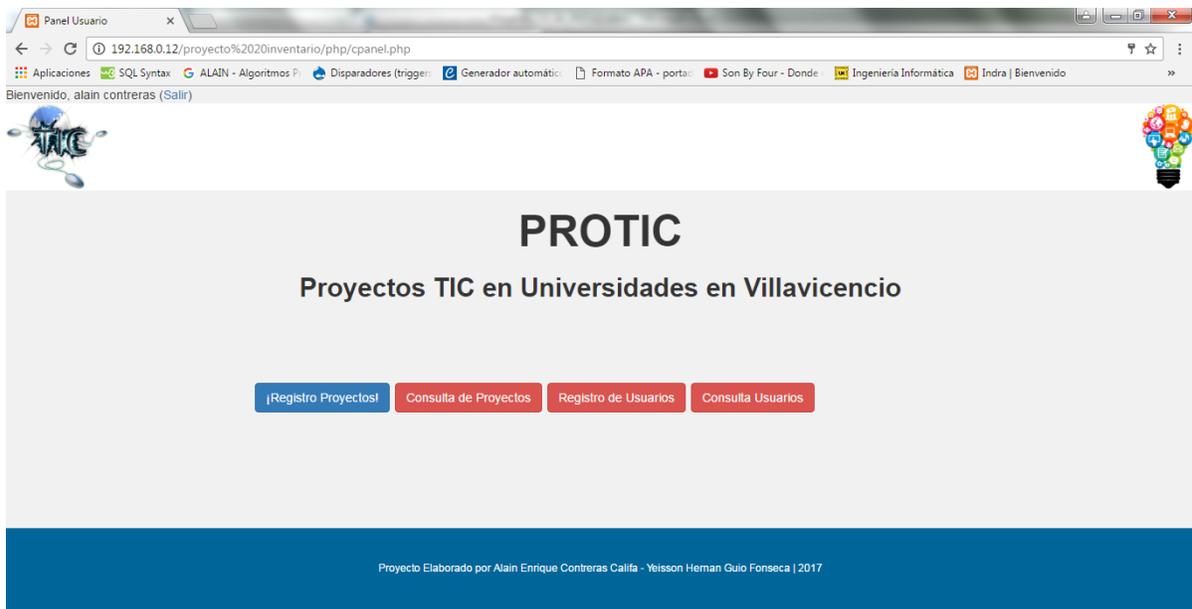


Figura 27. Interfaz rol Administrador.

Bienvenido, alain contreras (Salir)



[Inicio](#) [Consulta de Proyectos](#)

# PROTIC

## Proyectos TIC Universidades Villavicencio

### Registro de Proyectos

**Figura 28.** Interfaz Registro e ingreso de proyectos TIC.

Bienvenido, alain contreras (Salir)



[Inicio](#) [¡Registro Proyectos!](#) [Todos los Proyectos](#)

# PROTIC

## Proyectos TIC en Universidades en Villavicencio

[Buscar](#) [Limpiar](#)

Proyecto Elaborado por Alain Enrique Contreras Califa - Yeisson Heman Guilo Fonseca | 2017

**Figura 29.** Interfaz consulta proyectos TIC.

Bienvenido, alain contreras (Salir)




Inicio

# PROTIC

## Proyectos TIC en Universidades en Villavicencio

id_proyecto	Titulo	Linea	Universidad	Programa	Fecha	Autores	Ver	Editar	Eliminar
28	ANALISIS Y GESTION DE RIESGOS INFORMATICOS PARA LA RED LAN DE LA GOBERNACION DEL META	Informatica	U Cooperativa	Ing de Sistemas	2012	CAMILO ANDRES MARIN MARTINEZ y JUAN DIEGO LAITON RAMOS			
29	SISTEMA DE INFORMACION WEB GESTION ADMINISTRATIVA DE LA ESTACION DE SERVICIO LA LLANERITA	Comercial	U Cooperativa	Ing de Sistemas	2012	MIGUEL ANGEL GUTIERREZ y DIEGO MAURICIO SUAREZ			
30	SISTEMA DE INFORMACION MANTENIMIENTO DE EQUIPOS SINFOMECA	Informatica	U Cooperativa	Ing de Sistemas	2012	WILSON GEONANY LOZANO - JHOM EDISON CHAVARRIAGA BAHAMON- HARDY DAVID POLO CASTRO			
31	O DE UN SISTEMA DE INFORMACION WEB, PARA LA SISTEMATIZACION GESTION Y ADMINISTRACION DE LA PELUQUERIA KYP EVOLUTION	Comercial	Unimeta	Ing de sistemas	2013	Pablo Alberto Martinez			
32	ANALISIS Y DISEÑO DEL SOPORTE TECNICO PARA EL SOFTWARE MEMPHYS MEDIANTE LA MODALIDAD DE TELETRABAJO	Informatica	Unimeta	Ing de Sistemas	2003	YOVANA CAROLINA JARA MORALES - ENDERSON MORA AROCA			
33	DISEÑO Y DESARROLLO ENTRENADOR ECAES APLICADO A LAS	Informatica	Unimeta	Ing. de	2006	JOSE OCTAVIO LOPEZ LEIVA			

**Figura 30.** Interfaz resultado consulta proyectos TIC.

### Código PHP que ejecuta la consulta en la base de datos:

```
<?php
require_once("config.php");

$html = "<tr>
    <th>id_proyecto</th>
    <th>Titulo</th>
    <th>Linea</th>
        <th>Universidad</th>
    <th>Programa</th>
        <th>Fecha</th>
    <th>Autores</th>
        <th>Ver</th>
        <th>Editar</th>
        <th>Eliminar</th>
    </tr>";

$titulo = $_POST['inputtitulo'];
$universidad = $_POST['inputinstitucion'];
```

```

$linea = $_POST['inputlinea'];
$autor = $_POST['autor'];

if($titulo == "" || $universidad == "" || $linea == "" || $autor == ""){

}

if($universidad != "" && $titulo != "" && $linea != "" && $autor != ""){
    $consulta1 = "SELECT id_univer
                FROM universidades
                WHERE UPPER(n_universidad) LIKE
'%" . $universidad . "%";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                FROM lineas
                WHERE UPPER(tipo) LIKE '%" . $linea . "%";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
                e.id_proyecto,
                t.tipo,
                m.n_universidad,
                e.programa,
                e.fecha,
                e.autores,
                e.objetivo_general,
                e.titulo,
                e.resumen
                FROM proyectos e
                INNER JOIN universidades m
                ON e.t_universidad = m.id_univer
                INNER JOIN lineas t
                ON e.linea_p = t.id_tipo
                WHERE
                UPPER(e.autores)
                LIKE '%" . $autor . "%' AND e.t_universidad = '" . $universidad1 . "'
                AND UPPER(e.titulo) LIKE '%" . $titulo . "%' AND e.linea_p = '" . $linea1 . "'";
    $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

```

```

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_proyecto']."</td>
            <td>".$fila['titulo']."</td>
            <td>".$fila['tipo']."</td>
            <td>".$fila['n_universidad']."</td>
            <td>".$fila['programa']."</td>
            <td>".$fila['fecha']."</td>
            <td>".$fila['autores']."</td>
            <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
            <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar(".$fila['id_proyecto'].") id=".$fila['id_proyecto']."'></span></td>
            </tr>";
        }
        print $html;
    }
}

}

}

if($universidad == "" && $titulo == "" && $linea != "" && $autor != ""){
    $consulta1 = "SELECT id_univer
                        FROM universidades
                        WHERE UPPER(n_universidad) LIKE
'%" . $universidad . "%";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                FROM lineas
                WHERE UPPER(tipo) LIKE '%" . $linea . "%";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
                e.id_proyecto,

```

```

        t.tipo,
        m.n_universidad,
        e.programa,
        e.fecha,
        e.autores,
        e.objetivo_general,
        e.titulo,
        e.resumen
    FROM proyectos e
    INNER JOIN universidades m
    ON e.t_universidad = m.id_univer
    INNER JOIN lineas t
    ON e.linea_p = t.id_tipo
        WHERE
        UPPER(e.autores)
        LIKE '%"$.autor.%' AND e.linea_p = "$.linea1."";
$resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
        <td width=5%>".$fila['id_proyecto']."</td>
        <td>".$fila['titulo']."</td>
        <td>".$fila['tipo']."</td>
        <td>".$fila['n_universidad']."</td>
        <td>".$fila['programa']."</td>
        <td>".$fila['fecha']."</td>
        <td>".$fila['autores']."</td>
        <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id="$.fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
        <td><a href='/proyecto 20inventario/php/editar.php?id="$.fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar("$.fila['id_proyecto'].")' id="$.fila['id_proyecto']."'></span></td>
        </tr>";
    }
    print $html;

}else{
    print $html;
}

```

```

}

if($universidad == "" && $titulo != "" && $linea == "" && $autor != ""){
    $consulta1 = "SELECT id_univer
                  FROM universidades
                  WHERE UPPER(n_universidad) LIKE
%'".$universidad."%";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                  FROM lineas
                  WHERE UPPER(tipo) LIKE '>".$linea."%";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
                  e.id_proyecto,
                  t.tipo,
                  m.n_universidad,
                  e.programa,
                  e.fecha,
                  e.autores,
                  e.objetivo_general,
                  e.titulo,
                  e.resumen
                  FROM proyectos e
                  INNER JOIN universidades m
                  ON e.t_universidad = m.id_univer
                  INNER JOIN lineas t
                  ON e.linea_p = t.id_tipo
                  WHERE
                  UPPER(e.autores)
                  LIKE '>".$autor."%' AND UPPER(e.titulo) LIKE
%'".$titulo."%";
    $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

    $numReg = pg_num_rows($resultado);

    if($numReg>0){
        while ($fila = pg_fetch_array($resultado)) {
            $html .= "<tr>
            <td width='5%'>".$fila['id_proyecto']."</td>

```

```

        <td>".$fila['titulo']."</td>
        <td>".$fila['tipo']."</td>
        <td>".$fila['n_universidad']."</td>
        <td>".$fila['programa']."</td>
        <td>".$fila['fecha']."</td>
        <td>".$fila['autores']."</td>
        <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
        <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar($(".$fila['id_proyecto'].") id=".$fila['id_proyecto'].")></span></td>
    </tr>";
    }
    print $html;

}else{
    print $html;
}

}

if($Suniversidad == "" && $titulo != "" && $linea != "" && $autor == ""){
    $consulta1 = "SELECT id_univer
                    FROM universidades
                    WHERE UPPER(n_universidad) LIKE
'%" . $Suniversidad . "%'";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                    FROM lineas
                    WHERE UPPER(tipo) LIKE '%" . $linea . "%'";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
                    e.id_proyecto,
                    t.tipo,
                    m.n_universidad,
                    e.programa,
                    e.fecha,
                    e.autores,

```

```

        e.objetivo_general,
        e.titulo,
        e.resumen
    FROM proyectos e
    INNER JOIN universidades m
    ON e.t_universidad = m.id_univer
    INNER JOIN lineas t
    ON e.linea_p = t.id_tipo
        WHERE
        UPPER(e.titulo) LIKE '%" . $titulo . "%' AND e.linea_p =
'" . $linea1 . "'";
        $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
        <td width='5%'>". $fila['id_proyecto']. "</td>
        <td>". $fila['titulo']. "</td>
        <td>". $fila['tipo']. "</td>
        <td>". $fila['n_universidad']. "</td>
        <td>". $fila['programa']. "</td>
        <td>". $fila['fecha']. "</td>
        <td>". $fila['autores']. "</td>
        <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=" . $fila['id_proyecto']. "'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
        <td><a href='/proyecto 20inventario/php/editar.php?id=" . $fila['id_proyecto']. "'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar(" . $fila['id_proyecto']. ")' id=" . $fila['id_proyecto']. "'></span></td>
        </tr>";
    }
    print $html;
}
}
}

if($universidad == "" && $titulo != "" && $linea != "" && $autor != ""){
    $consulta1 = "SELECT id_univer
        FROM universidades

```

```

WHERE UPPER(n_universidad) LIKE
'%"$.Suniversidad."%";
$resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
while ($fila = pg_fetch_array($resultado1)) {
    $Suniversidad1 = $fila['id_univer'];
}
$consulta2 = "SELECT id_tipo
FROM lineas
WHERE UPPER(tipo) LIKE '%"$.linea."%";
$resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
while ($fila = pg_fetch_array($resultado2)) {
    $linea1 = $fila['id_tipo'];
}
$consulta = "SELECT
    e.id_proyecto,
    t.tipo,
    m.n_universidad,
    e.programa,
    e.fecha,
    e.autores,
    e.objetivo_general,
    e.titulo,
    e.resumen
FROM proyectos e
INNER JOIN universidades m
ON e.t_universidad = m.id_univer
INNER JOIN lineas t
ON e.linea_p = t.id_tipo
WHERE
UPPER(e.autores)
LIKE '%"$.$autor."%' AND UPPER(e.titulo) LIKE '%"$. $titulo."%'
AND e.linea_p = '"$.linea1."";
$resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
        <td width='5%'>".$fila['id_proyecto']."</td>
        <td>".$fila['titulo']."</td>
        <td>".$fila['tipo']."</td>
        <td>".$fila['n_universidad']."</td>
        <td>".$fila['programa']."</td>
        <td>".$fila['fecha']."</td>

```

```

        <td>".$fila['autores']. "</td>
        <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
        <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar($(".$fila['id_proyecto'].") id=".$fila['id_proyecto'].")"></span></td>
    </tr>";
    }
    print $html;

}else{
    print $html;
}

}

if($universidad != "" && $titulo == "" && $linea == "" && $autor != ""){
    $consulta1 = "SELECT id_univer
                    FROM universidades
                    WHERE UPPER(n_universidad) LIKE
'%" . $universidad . "%";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                FROM lineas
                WHERE UPPER(tipo) LIKE '%" . $linea . "%";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
                e.id_proyecto,
                t.tipo,
                m.n_universidad,
                e.programa,
                e.fecha,
                e.autores,
                e.objetivo_general,
                e.titulo,
                e.resumen
                FROM proyectos e
                INNER JOIN universidades m

```

```

        ON e.t_universidad = m.id_univer
        INNER JOIN lineas t
        ON e.linea_p = t.id_tipo
           WHERE
           UPPER(e.autores)
           LIKE '%"$.sautor.%' AND e.t_universidad = '".$universidad1.'";
$resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
        <td width=5%>".$fila['id_proyecto']."</td>
        <td>".$fila['titulo']."</td>
        <td>".$fila['tipo']."</td>
        <td>".$fila['n_universidad']."</td>
        <td>".$fila['programa']."</td>
        <td>".$fila['fecha']."</td>
        <td>".$fila['autores']."</td>
        <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
        <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar(".$fila['id_proyecto'].")' id=".$fila['id_proyecto']."'></span></td>
        </tr>";
    }
    print $html;

}

}

}

if($universidad != "" && $titulo == "" && $linea != "" && $sautor == ""){
    $consulta1 = "SELECT id_univer
                FROM universidades
                WHERE UPPER(n_universidad) LIKE
'%"$.$universidad.%";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];

```

```

    }
    $consulta2 = "SELECT id_tipo
    FROM lineas
    WHERE UPPER(tipo) LIKE '%".$linea."%";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
    e.id_proyecto,
    t.tipo,
    m.n_universidad,
    e.programa,
    e.fecha,
    e.autores,
    e.objetivo_general,
    e.titulo,
    e.resumen
    FROM proyectos e
    INNER JOIN universidades m
    ON e.t_universidad = m.id_univer
    INNER JOIN lineas t
    ON e.linea_p = t.id_tipo
    WHERE
    e.t_universidad = ".$universidad1." AND e.linea_p =
    ".$linea1."";
    $resultado = pg_query($conexion, $consulta) or die("Error en la
    Consulta SQL");

    $numReg = pg_num_rows($resultado);

    if($numReg>0){
        while ($fila = pg_fetch_array($resultado)) {
            $html .= "<tr>
            <td width='5%'>".$fila['id_proyecto']."</td>
            <td>".$fila['titulo']."</td>
            <td>".$fila['tipo']."</td>
            <td>".$fila['n_universidad']."</td>
            <td>".$fila['programa']."</td>
            <td>".$fila['fecha']."</td>
            <td>".$fila['autores']."</td>
            <td><a href='/proyecto
    20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
    glyphicon-eye-open click'</span></a></td>
            <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
    class='glyphicon glyphicon-pencil click'</span></a></td>

```

```

        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar("$fila['id_proyecto'].") id="$fila['id_proyecto']."'></span></td>
    </tr>";
    }
    print $html;

}else{
    print $html;
}

}

if($universidad != "" && $titulo == "" && $linea != "" && $autor != ""){
    $consulta1 = "SELECT id_univer
                FROM universidades
                WHERE UPPER(n_universidad) LIKE
'%"$universidad."%";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                FROM lineas
                WHERE UPPER(tipo) LIKE '%"$linea."%";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
                e.id_proyecto,
                t.tipo,
                m.n_universidad,
                e.programa,
                e.fecha,
                e.autores,
                e.objetivo_general,
                e.titulo,
                e.resumen
                FROM proyectos e
                INNER JOIN universidades m
                ON e.t_universidad = m.id_univer
                INNER JOIN lineas t
                ON e.linea_p = t.id_tipo
                WHERE
                UPPER(e.autores)

```

```

                LIKE '%" . $autor . "%' AND e.t_universidad = '" . $universidad1 . "'
AND e.linea_p = '" . $linea1 . "'";
                $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>" . $fila['id_proyecto'] . "</td>
            <td>" . $fila['titulo'] . "</td>
            <td>" . $fila['tipo'] . "</td>
            <td>" . $fila['n_universidad'] . "</td>
            <td>" . $fila['programa'] . "</td>
            <td>" . $fila['fecha'] . "</td>
            <td>" . $fila['autores'] . "</td>
            <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=" . $fila['id_proyecto'] . "'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
            <td><a href='/proyecto 20inventario/php/editar.php?id=" . $fila['id_proyecto'] . "'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar(" . $fila['id_proyecto'] . ")' id=" . $fila['id_proyecto'] . "'></span></td>
            </tr>";
        }
        print $html;
    }else{
        print $html;
    }
}

if($universidad != "" && $titulo != "" && $linea == "" && $autor == ""){
    $consulta1 = "SELECT id_univer
                FROM universidades
                WHERE UPPER(n_universidad) LIKE
%' . $universidad . "%'";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                FROM lineas
                WHERE UPPER(tipo) LIKE '%" . $linea . "%'";

```

```

$resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
while ($fila = pg_fetch_array($resultado2)) {
    $linea1 = $fila['id_tipo'];
}
    $consulta = "SELECT
        e.id_proyecto,
        t.tipo,
        m.n_universidad,
        e.programa,
        e.fecha,
        e.autores,
        e.objetivo_general,
        e.titulo,
        e.resumen
    FROM proyectos e
    INNER JOIN universidades m
    ON e.t_universidad = m.id_univer
    INNER JOIN lineas t
    ON e.linea_p = t.id_tipo
    WHERE
        e.t_universidad = ".$universidad1." AND UPPER(e.titulo) LIKE
'%" . $titulo . "%";
        $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_proyecto']. "</td>
            <td>".$fila['titulo']. "</td>
            <td>".$fila['tipo']. "</td>
            <td>".$fila['n_universidad']. "</td>
            <td>".$fila['programa']. "</td>
            <td>".$fila['fecha']. "</td>
            <td>".$fila['autores']. "</td>
            <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
            <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar(".$fila['id_proyecto'].")' id='".$fila['id_proyecto']."'></span></td>
            </tr>";
        }
}

```

```

        print $html;
    }else{
        print $html;
    }
}

if($universidad != "" && $titulo != "" && $linea != "" && $autor == ""){
    $consulta1 = "SELECT id_univer
                FROM universidades
                WHERE UPPER(n_universidad) LIKE
%'".$universidad."%";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                FROM lineas
                WHERE UPPER(tipo) LIKE '%" . $linea . "%";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
                e.id_proyecto,
                t.tipo,
                m.n_universidad,
                e.programa,
                e.fecha,
                e.autores,
                e.objetivo_general,
                e.titulo,
                e.resumen
                FROM proyectos e
                INNER JOIN universidades m
                ON e.t_universidad = m.id_univer
                INNER JOIN lineas t
                ON e.linea_p = t.id_tipo
                WHERE
                e.t_universidad = '".$universidad1.'" AND UPPER(e.titulo) LIKE
%'".$titulo."%' AND e.linea_p = '".$linea1.'";
    $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

```

```

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_proyecto']."</td>
            <td>".$fila['titulo']."</td>
            <td>".$fila['tipo']."</td>
            <td>".$fila['n_universidad']."</td>
            <td>".$fila['programa']."</td>
            <td>".$fila['fecha']."</td>
            <td>".$fila['autores']."</td>
            <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
            <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar(".$fila['id_proyecto'].") id=".$fila['id_proyecto']."'></span></td>
            </tr>";
        }
        print $html;
    }
}

}

}

if($universidad != "" && $titulo != "" && $linea == "" && $autor != ""){
    $consulta1 = "SELECT id_univer
                        FROM universidades
                        WHERE UPPER(n_universidad) LIKE
'%" . $universidad . "%";
    $resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado1)) {
        $universidad1 = $fila['id_univer'];
    }
    $consulta2 = "SELECT id_tipo
                FROM lineas
                WHERE UPPER(tipo) LIKE '%" . $linea . "%";
    $resultado2 = pg_query($conexion, $consulta2) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado2)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
                e.id_proyecto,

```

```

t.tipo,
m.n_universidad,
e.programa,
e.fecha,
e.autores,
e.objetivo_general,
e.titulo,
e.resumen
FROM proyectos e
INNER JOIN universidades m
ON e.t_universidad = m.id_univer
INNER JOIN lineas t
ON e.linea_p = t.id_tipo
WHERE
UPPER(e.autores)
LIKE '%".$autor.'" AND e.t_universidad = '".$universidad1.'"
AND UPPER(e.titulo) LIKE '%".$titulo.'"";
$resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
while ($fila = pg_fetch_array($resultado)) {
    $html .= "<tr>
    <td width='5%'>".$fila['id_proyecto']."</td>
    <td>".$fila['titulo']."</td>
    <td>".$fila['tipo']."</td>
    <td>".$fila['n_universidad']."</td>
    <td>".$fila['programa']."</td>
    <td>".$fila['fecha']."</td>
    <td>".$fila['autores']."</td>
    <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
    <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
    <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar("'".$fila['id_proyecto'].")' id='".$fila['id_proyecto']."'></span></td>
    </tr>";
    }
print $html;

}else{
print $html;
}

```

```

}

if($universidad != "" && $titulo == "" && $linea == "" && $autor == ""){
    $consulta3 = "SELECT id_univer
                FROM universidades
                WHERE UPPER(n_universidad) LIKE
'%".$universidad."%";

    $universidad1 = 0;
    $resultado3 = pg_query($conexion, $consulta3) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado3)){

        $universidad1 = $fila['id_univer'];
    }
    $consulta = "SELECT
                e.id_proyecto,
                t.tipo,
                m.n_universidad,
                e.programa,
                e.fecha,
                e.autores,
                e.objetivo_general,
                e.titulo,
                e.resumen
                FROM proyectos e
                INNER JOIN universidades m
                ON e.t_universidad = m.id_univer
                INNER JOIN lineas t
                ON e.linea_p = t.id_tipo
                WHERE
                e.t_universidad = ".$universidad1."";

    $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
                <td width='5%'>".$fila['id_proyecto']."</td>
                <td>".$fila['titulo']."</td>
                <td>".$fila['tipo']."</td>
                <td>".$fila['n_universidad']."</td>
                <td>".$fila['programa']."</td>
                <td>".$fila['fecha']."</td>

```

```

        <td>".$fila['autores']. "</td>
        <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
        <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar($(".$fila['id_proyecto'].")' id=".$fila['id_proyecto'].")"></span></td>
    </tr>";
    }
    print $html;

}else{
    print $html;
}

}

```

```

if($titulo != "" && $universidad == "" && $linea == "" && $autor == ""){

```

```

    $consulta = "SELECT
        e.id_proyecto,
        t.tipo,
        m.n_universidad,
        e.programa,
        e.fecha,
        e.autores,
        e.objetivo_general,
        e.titulo,
        e.resumen
    FROM proyectos e
    INNER JOIN universidades m
    ON e.t_universidad = m.id_univer
    INNER JOIN lineas t
    ON e.linea_p = t.id_tipo
        WHERE UPPER(e.titulo) LIKE '%".$titulo."%";
    $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

```

```

$numReg = pg_num_rows($resultado);

```

```

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
        <td width='5%'>".$fila['id_proyecto']. "</td>
        <td>".$fila['titulo']. "</td>

```

```

        <td>".$fila['tipo']."</td>
        <td>".$fila['n_universidad']."</td>
        <td>".$fila['programa']."</td>
        <td>".$fila['fecha']."</td>
        <td>".$fila['autores']."</td>
        <td><a href="/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
        <td><a href="/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar($(".$fila['id_proyecto'].") id=".$fila['id_proyecto'].")></span></td>
    </tr>";
    }
    print $html;

}else{
    print $html;
}

}

if($linea != "" && $universidad == "" && $titulo == "" && $autor == ""){
    $consulta4 = "SELECT id_tipo
    FROM lineas
    WHERE UPPER(tipo) LIKE '%" . $linea . "%'";
    $linea1 = 0;
    $resultado4 = pg_query($conexion, $consulta4) or die("Error en la Consulta SQL");
    while ($fila = pg_fetch_array($resultado4)) {
        $linea1 = $fila['id_tipo'];
    }
    $consulta = "SELECT
        e.id_proyecto,
        t.tipo,
        m.n_universidad,
        e.programa,
        e.fecha,
        e.autores,
        e.objetivo_general,
        e.titulo,
        e.resumen
    FROM proyectos e
    INNER JOIN universidades m
    ON e.t_universidad = m.id_univer
    INNER JOIN lineas t
    ON e.linea_p = t.id_tipo

```

```

WHERE e.linea_p = ".$linea1."";
$resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_proyecto']."</td>
            <td>".$fila['titulo']."</td>
            <td>".$fila['tipo']."</td>
            <td>".$fila['n_universidad']."</td>
            <td>".$fila['programa']."</td>
            <td>".$fila['fecha']."</td>
            <td>".$fila['autores']."</td>
            <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
            <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar(".$fila['id_proyecto'].")' id=".$fila['id_proyecto']."'></span></td>
            </tr>";
        }
        print $html;
    }else{
        print $html;
    }
}

if($autor != "" && $universidad == "" && $titulo == "" && $linea == ""){

    $consulta = "SELECT
        e.id_proyecto,
        t.tipo,
        m.n_universidad,
        e.programa,
        e.fecha,
        e.autores,
        e.objetivo_general,
        e.titulo,
        e.resumen
    FROM proyectos e

```

```

INNER JOIN universidades m
ON e.t_universidad = m.id_univer
INNER JOIN lineas t
ON e.linea_p = t.id_tipo
WHERE UPPER(e.autores) LIKE '%".$autor.%";
$resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_proyecto']."</td>
            <td>".$fila['titulo']."</td>
            <td>".$fila['tipo']."</td>
            <td>".$fila['n_universidad']."</td>
            <td>".$fila['programa']."</td>
            <td>".$fila['fecha']."</td>
            <td>".$fila['autores']."</td>
            <td><a href='/proyecto
20inventario/php/ver_proyecto.php?id=".$fila['id_proyecto']."'><span class='glyphicon
glyphicon-eye-open click'</span></a></td>
            <td><a href='/proyecto 20inventario/php/editar.php?id=".$fila['id_proyecto']."'><span
class='glyphicon glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar(".$fila['id_proyecto'].")' id="".$fila['id_proyecto']."'></span></td>
            </tr>";
        }
        print $html;
    }else{
        print $html;
    }
}

?>

```

The screenshot shows a web interface titled "PROTIC" with the subtitle "Proyectos TIC en Universidades en Villavicencio". A red "Inicio" button is in the top left. The main content area displays the following information:

- Universidad:** U Cooperativa
- Línea:** Informática
- Título del proyecto:** ANALISIS Y GESTION DE RIESGOS INFORMATICOS
- Autor:** CAMILO ANDRES MARIN MARTINEZ y JUAN DIEGO
- Programa:** Ing de Sistemas
- Objetivo General:** Analizar la red lan de la gobernacion del meta y evider
- Objetivos Especificos:** Analizar la red lan de la gobernacion del meta y evider
- Resumen:** Analizar la red lan de la gobernacion del meta y evidenciar posibles riesgos.

**Figura 31.** Interfaz ver un proyecto - proyectos TIC.

### Código PHP que ejecuta Ver proyectos TIC:

```
<?php
session_start();
require_once("bloqueo.php");
require_once("config.php");
$id = $_GET['id'];

$query = "SELECT
    e.id_proyecto,
    m.n_universidad,
    t.tipo,
    e.titulo ,
    e.autores,
    e.programa,
    e.objetivo_general,
    e.objetivo_especificos,
    e.resumen,
    e.fecha,
    e.observacion
```

```

        FROM proyectos e
        INNER JOIN universidades m
        ON e.t_universidad = m.id_univer
        INNER JOIN lineas t
        ON e.linea_p = t.id_tipo
        WHERE e.id_proyecto = $id";
$resultado = pg_query($conexion, $query) or die("Error en la consulta SQL1");
while($row = pg_fetch_array($resultado)){
    $id1 = $row["id_proyecto"];
    $autor = $row["autores"];
    $titulo = $row["titulo"];
    $programa = $row["programa"];
    $objetivo = $row["objetivo_general"];
    $objetivos = $row["objetivo_especificos"];
    $resume = $row["resumen"];
    $fecha = $row["fecha"];
    $observacion = $row["observacion"];
    $universidad = $row["n_universidad"];
    $tipo = $row["tipo"];

}

$query1 = "SELECT
    t_universidad,
    linea_p
    FROM
    proyectos
    WHERE id_proyecto = $id";
$resultado1 = pg_query($conexion, $query1) or die("Error en la consulta SQL2");
while($row = pg_fetch_array($resultado1)){
    $universidad1 = $row["t_universidad"];
    $linea = $row["linea_p"];

}
?>

```

Bienvenido, alain contreras (Salir)




[Inicio](#) [Consulta Usuarios](#)

# PROTIC

## Proyectos TIC Universidades Villavicencio

### Registro de Usuarios

[Registrar](#) [Limpiar](#)

Proyecto Elaborado por Alain Enrique Contreras Califa - Ybisson Hernan Guio Fonseca | 2017

**Figura 32.** Interfaz Registro de usuario proyectos TIC.

### **Código PHP que registra los usuarios en el sistema de información:**

```

<?php
require_once("config.php");

$nombre = $_POST['nombre'];
$apellido= $_POST['apellido'];
$usuario = $_POST['usuario'];
$clave = $_POST['clave'];

$con consulta = "INSERT INTO usuarios(
    usuario,
    clave,
    nombre,
    apellido,
    rol
    )"

```

```
VALUES
    (".$usuario.", ".$slave.", ".$nombre.", ".$apellido.", 2)
";
$consulta1 = "select usuario from usuarios where usuario = ".$usuario."";
$resultado = pg_query($conexion, $consulta) or die("Error en la Consulta SQL");
$resultado1 = pg_query($conexion, $consulta1) or die("Error en la Consulta SQL");
$numReg = pg_num_rows($resultado1);

if($numReg>0){

    print "1";

}else{
    print "0";
}
?>
```



**Figura 33.** Interfaz Consulta de usuario proyectos TIC.

### **Código PHP que ejecuta consulta usuarios:**

```
<?php
require_once("config.php");

$html = "<tr>
    <th>id_usuario</th>
    <th>Nombre</th>
        <th>Apellido</th>
    <th>Usuario</th>
        <th>Contraseña</th>
    <th>Rol</th>
        <th>Editar</th>
        <th>Eliminar</th>
    </tr>";

$usuario = $_POST['inputusuario'];
$apellido = $_POST['inputapellido'];
$nombre = $_POST['inputnombre'];

if($usuario == "" || $apellido == "" || $nombre == ""){
```

```

}

if($usuario != "" && $apellido != "" && $nombre != ""){
    $consulta = "SELECT
        u.id_usuario,
        u.nombre,
        u.apellido,
        u.usuario,
        u.clave,
        r.rol
    FROM usuarios u
    INNER JOIN rol r
    ON u.rol = r.id_rol
    WHERE
    UPPER(u.apellido) LIKE '%".$apellido."' AND
    UPPER(u.usuario) LIKE '%".$usuario."' AND
    UPPER(u.nombre) LIKE '%".$nombre."%";
    $resultado = pg_query($conexion, $consulta) or die("Error en la
Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_usuario']."</td>
            <td>".$fila['nombre']."</td>
            <td>".$fila['apellido']."</td>
            <td>".$fila['usuario']."</td>
            <td>".$fila['clave']."</td>
            <td>".$fila['rol']."</td>
            <td><a href='/proyecto
20inventario/php/editar_usuario.php?id=".$fila['id_usuario']."'><span class='glyphicon
glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar1(\".$fila['id_usuario'].")' id=".$fila['id_usuario']."'></span></td>
            </tr>";
        }
        print $html;
    }else{
        print $html;
    }
}
if($usuario != "" && $apellido != "" && $nombre == ""){
    $consulta = "SELECT

```

```

        u.id_usuario,
        u.nombre,
        u.apellido,
        u.usuario,
        u.clave,
        r.rol
    FROM usuarios u
    INNER JOIN rol r
    ON u.rol = r.id_rol
WHERE
UPPER(u.apellido) LIKE '%"$.apellido.'" AND
UPPER(u.usuario) LIKE '%"$.usuario.'"";
$resultado = pg_query($conexion, $consulta) or die("Error en la Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_usuario']."</td>
            <td>".$fila['nombre']."</td>
            <td>".$fila['apellido']."</td>
            <td>".$fila['usuario']."</td>
            <td>".$fila['clave']."</td>
            <td>".$fila['rol']."</td>
            <td><a href=/proyecto
20inventario/php/editar_usuario.php?id=".$fila['id_usuario']."'><span class='glyphicon
glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar1("$.fila['id_usuario'].")' id=".$fila['id_usuario']."'></span></td>
            </tr>";
        }
    print $html;

} else {
    print $html;
}
}
if($usuario != "" && $apellido == "" && $nombre != ""){
    $consulta = "SELECT
        u.id_usuario,
        u.nombre,
        u.apellido,
        u.usuario,
        u.clave,
        r.rol

```

```

        FROM usuarios u
        INNER JOIN rol r
        ON u.rol = r.id_rol
    WHERE
    UPPER(u.usuario) LIKE '%".$usuario."' AND
    UPPER(u.nombre) LIKE '%".$nombre."%";
    $resultado = pg_query($conexion, $consulta) or die("Error en la Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_usuario']."</td>
            <td>".$fila['nombre']."</td>
            <td>".$fila['apellido']."</td>
            <td>".$fila['usuario']."</td>
            <td>".$fila['clave']."</td>
            <td>".$fila['rol']."</td>
            <td><a href='/proyecto
20inventario/php/editar_usuario.php?id=".$fila['id_usuario']."'><span class='glyphicon
glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar1(\".$fila['id_usuario'].\")' id=\".$fila['id_usuario'].\"></span></td>
            </tr>";
        }
    print $html;

}else{
    print $html;
}
}
if($usuario != "" && $apellido == "" && $nombre == ""){
    $consulta = "SELECT
        u.id_usuario,
        u.nombre,
        u.apellido,
        u.usuario,
        u.clave,
        r.rol
    FROM usuarios u
    INNER JOIN rol r
    ON u.rol = r.id_rol
    WHERE
    UPPER(u.usuario) LIKE '%".$usuario."%";
    $resultado = pg_query($conexion, $consulta) or die("Error en la Consulta SQL");
}

```

```

$numReg = pg_num_rows($resultado);

if($numReg>0){
  while ($fila = pg_fetch_array($resultado)) {
    $html .= "<tr>
      <td width='5%'>".$fila['id_usuario']."</td>
      <td>".$fila['nombre']."</td>
      <td>".$fila['apellido']."</td>
      <td>".$fila['usuario']."</td>
      <td>".$fila['clave']."</td>
      <td>".$fila['rol']."</td>
      <td><a href='/proyecto
20inventario/php/editar_usuario.php?id=".$fila['id_usuario']."'><span class='glyphicon
glyphicon-pencil click'</span></a></td>
      <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar1(\".$fila['id_usuario'].")' id=\".$fila['id_usuario']."'></span></td>
    </tr>";
  }
  print $html;

}else{
  print $html;
}
}

if($usuario == "" && $apellido != "" && $nombre != ""){
  $consulta = "SELECT
    u.id_usuario,
    u.nombre,
    u.apellido,
    u.usuario,
    u.clave,
    r.rol
  FROM usuarios u
  INNER JOIN rol r
  ON u.rol = r.id_rol
  WHERE
  UPPER(u.apellido) LIKE '%" . $apellido . "%' AND
  UPPER(u.nombre) LIKE '%" . $nombre . "%'";
  $resultado = pg_query($conexion, $consulta) or die("Error en la Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
  while ($fila = pg_fetch_array($resultado)) {
    $html .= "<tr>

```

```

        <td width='5%'>".$fila['id_usuario']."</td>
        <td>".$fila['nombre']."</td>
        <td>".$fila['apellido']."</td>
        <td>".$fila['usuario']."</td>
        <td>".$fila['clave']."</td>
        <td>".$fila['rol']."</td>
        <td><a href='/proyecto
20inventario/php/editar_usuario.php?id=".$fila['id_usuario']."'><span class='glyphicon
glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar1($(".$fila['id_usuario'].")' id=".$fila['id_usuario'].")></span></td>
    </tr>";
}
print $html;

}else{
    print $html;
}
}
if($usuario == "" && $apellido != "" && $nombre == ""){
    $consulta = "SELECT
        u.id_usuario,
        u.nombre,
        u.apellido,
        u.usuario,
        u.clave,
        r.rol
    FROM usuarios u
    INNER JOIN rol r
    ON u.rol = r.id_rol
    WHERE
    UPPER(u.apellido) LIKE '%" . $apellido . "%'";
    $resultado = pg_query($conexion, $consulta) or die("Error en la Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_usuario']."</td>
            <td>".$fila['nombre']."</td>
            <td>".$fila['apellido']."</td>
            <td>".$fila['usuario']."</td>
            <td>".$fila['clave']."</td>
            <td>".$fila['rol']."</td>

```

```

        <td><a href='/proyecto
20inventario/php/editar_usuario.php?id=".$fila['id_usuario']."'><span class='glyphicon
glyphicon-pencil click'</span></a></td>
        <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar1($(".fila['id_usuario'].")' id=".$fila['id_usuario']."'></span></td>
        </tr>";
    }
    print $html;

}else{
    print $html;
}
}
if($usuario == "" && $apellido == "" && $nombre != ""){
    $consulta = "SELECT
        u.id_usuario,
        u.nombre,
        u.apellido,
        u.usuario,
        u.clave,
        r.rol
    FROM usuarios u
    INNER JOIN rol r
    ON u.rol = r.id_rol
    WHERE
    UPPER(u.nombre) LIKE '%" . $nombre . "%'";
    $resultado = pg_query($conexion, $consulta) or die("Error en la Consulta SQL");

$numReg = pg_num_rows($resultado);

if($numReg>0){
    while ($fila = pg_fetch_array($resultado)) {
        $html .= "<tr>
            <td width='5%'>".$fila['id_usuario']. "</td>
            <td>".$fila['nombre']. "</td>
            <td>".$fila['apellido']. "</td>
            <td>".$fila['usuario']. "</td>
            <td>".$fila['clave']. "</td>
            <td>".$fila['rol']. "</td>
            <td><a href='/proyecto
20inventario/php/editar_usuario.php?id=".$fila['id_usuario']."'><span class='glyphicon
glyphicon-pencil click'</span></a></td>
            <td><span class='glyphicon glyphicon-trash click'
onClick='eliminar1($(".fila['id_usuario'].")' id=".$fila['id_usuario']."'></span></td>
            </tr>";
    }
}

```

```
    print $html;  
  }else{  
    print $html;  
  }  
  
}  
?>
```

## CONCLUSIONES

Realizado el trabajo de campo y diseñada la aplicación que contiene los proyectos TIC consultados en las universidades mencionadas (ver anexo 01), se logró materializar lo propuesto en cuanto a evidenciar cuales son las tendencias de desarrollo de software en las universidades de la ciudad de Villavicencio.

Se sistematizó la información recolectada para consulta e ingreso de nuevas temáticas de desarrollo de proyectos TIC, con el fin de que a futuro sea implementado para consulta en la universidad.

Se trabajó y se desarrolló con el acompañamiento del semillero de investigación Orinoco TIC de la Corporación Universitaria Minuto de Dios.

## RECOMENDACIONES

Actualizar el software entregado con los nuevos proyectos de la Corporación Universitaria Minuto de Dios, al igual se sugiere colocar el software en el repositorio de la biblioteca de la universidad donde los estudiantes o quien lo necesite lo puedan consultar.

Incentivar al desarrollo de proyectos como opción de grado, ya que de ahí parte la innovación y la creación de nuevas ideas.

La información recolectada puede servir a futuro para formular proyectos nuevos, que no se repitan, que esto incentive a los estudiantes a innovar. De igual manera se deduce que por el número tan bajo de proyectos en las universidades consultadas (Ver Anexo1), los estudiantes dan preferencia a los diplomados o seminarios como opción de grado; este proyecto también puede llevar a incentivar a los estudiantes a que opten por el desarrollo de un proyecto como opción de grado.

## BIBLIOGRAFÍA

Uniminuto. (2016). Instructivo proceso de grados 2016-2. Recuperado el 20 de Julio de 2016, de <http://www.uniminuto.edu/documents/991974/2211242/INSTRUCTIVO+DE+GRADOS+UNIMINUTO+2015-21+final.pdf/b981b571-b408-4a40-8049-f15a5f0621e0>

Kendall, E y Kenneth. (2005). Análisis y diseño de Sistemas. México.

C.J DATE. (2015). Introducción a los sistemas de bases de datos, séptima edición.

Booch, Rumbaugh y Jacobson (2006): El Lenguaje Unificado de Modelado.

Medina, N. (2014). XAMPP, ventajas y desventajas. Recuperado el 1 de Marzo de 2017, de <http://pnfiservidores.blogspot.com/2013/01/ventajas-y-desventajas-de-xampp.html>

Desarrollo en espiral. (2017, 22 de marzo). Boehm B, "[A Spiral Model of Software Development and Enhancement](#)", IEEE Computer, IEEE, Marzo de 1988 desde <http://csse.usc.edu/TECHRPTS/1988/usccse88-500/usccse88-500.pdf>.

Rodriguez, M. (2016). *Sede Amazonia - Universidad Nacional de Colombia. Unal.edu.co.*

Retrieved 23 November 2016, from <http://www.unal.edu.co/siamac/sig/metadatos1.html>

*Ley Lleras - Derechos de autor Propiedad Intelectual - Tecnología Colombia.com.* (2016). *Colombia.com.* Retrieved 23 November 2016, from <http://www.colombia.com/tecnologia/derechos-de-autor-propiedad-intelectual/nacional/ley-lleras.aspx>

*Consulta de la Norma:.* (2016). *Alcaldiabogota.gov.co.* Retrieved 23 November 2016, from <http://www.alcaldiabogota.gov.co/sisjur/normas/Normal.jsp?i=4125>

(2016). Retrieved 23 November 2016, from <http://www.uniminuto.edu/documents>

Camacho, H. and Cámara, L. (2001). *El Enfoque del marco lógico: 10 casos prácticos*. 1st ed. Madrid: Fundación CIDEAL.

Arenas Martínez, R. (2016). *Manual Para la Elaboración de proyectos*. 1st ed. [ebook] Bolivia. Available at: <http://www.dalcame.com/wdescarga/manualdeProyectos.pdf>/Accessed 8 Oct. 2016.

Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2006). *Metodología de la investigación* (1st ed.). México: McGraw Hill.

Metodologías de desarrollo de software. Obtenido de Metodologías de desarrollo de software: Accessed 08 Oct de 2016. <https://sites.google.com/site/metdlgsddesarrollodesoftware/modelo-contractual>.

Gobernación del Meta (2017 11 de Abril) El Meta. Obtenido de Nuestra tierra: <http://www.meta.gov.co/web/content/nuestro-departamento>.

PostgreSQL. (2017, 23 de marzo ( PostgreSQL Global Development Group, 2017). Fecha de consulta: 23:41, abril 11, 2017 desde <http://www.postgresql.org.es/>.

HTML5. (2017, 7 de abril). World Wide Web Consortium (W3C) - España . Fecha de consulta: 23:49, abril 11, 2017 desde <https://www.w3schools.com/>.

Bootstrap (framework). (2017, 27 de marzo). *GetBootstrap*. Fecha de consulta: 00:14, abril 12, 2017 desde <http://getbootstrap.com/>

Hoja de estilos en cascada. (2017, 23 de febrero). World Wide Web Consortium (W3C). Fecha de consulta: 00:16, abril 12, 2017 desde <https://www.w3schools.com/css/default.asp>

PHP. (2017, 4 de abril). World Wide Web Consortium (W3C).. Fecha de consulta: 00:21, abril 12, 2017 desde <https://www.w3schools.com/php/default.asp>

JavaScript. (2017, 7 de abril). *Mozilla Foundation, 2017*. Fecha de consulta: 00:22, abril 12, 2017 desde <https://developer.mozilla.org/es/docs/Web/JavaScript>

AJAX. (2016, 2 de septiembre). World Wide Web Consortium (W3C), 2107 Fecha de consulta: 00:24, abril 12, 2017 desde [https://www.w3schools.com/php/php\\_ajax\\_intro.asp](https://www.w3schools.com/php/php_ajax_intro.asp).

## ANEXOS

**Anexo 001.** Lista universidades consultadas para la investigación.

- UNIVERSIDAD COOPERATIVA DE COLOMBIA. (UCC).
- UNIVERSIDAD DEL META (UNIMETA).
- UNIVERSIDAD AUTONOMA DEL NARIÑO (Uniautonoma).
- UNIVERSIDAD DE LOS LLANOS (UNILLANOS).
- CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS. (Uniminuto).

**Anexo 002.** Encuesta.**ENCUESTA DE RECOPIACION DE DATOS TÉCNICOS DE LOS PROYECTOS TIC DE LA UNIVERSIDADES EN VILLAVICENCIO****Objetivo**

Recopilar información de los proyectos TIC de las instituciones de educación superior en la ciudad de Villavicencio

- I. ¿Qué tanto registran la información de Proyectos del área de las Tecnologías de la Información y las Comunicaciones TIC?
  1. Nada
  2. Poco
  3. Regular
  4. Bastante
  5. Mucho
  
- II. Ha tenido inconvenientes al consultar la información referente a Proyectos TIC
  1. Si
  2. No
  
- III. Si respondió Si a la anterior pregunta, indique en dónde se guarda esta información.
  1. Biblioteca de la Institución.
  2. Documentos en archivo físico.
  3. Medio digital (CD, DVD, Memoria USB, Discos duros de respaldo)
  4. Nube o Servidor externo
  
- IV. Califique la velocidad de consulta de la información:
  1. Baja
  2. Media
  3. Alta
  
- V. ¿En cuál de las siguientes líneas se clasifican los proyectos TIC que se han realizado en la Institución?
 

1. Formulación y gestión de proyectos TI	10. Infraestructura telecomunicaciones
2. Desarrollo de software	11. Soporte tecnológico
3. Multimedia y contenidos web	12. Electrónica
4. Ciberseguridad y seguridad de datos	13. Aplicaciones drones
5. Antivirus	14. Capacitación
6. Geo posicionamiento y geo localización	15. Equipamiento de oficina y mobiliario

7. Aplicaciones móviles
8. Modelado y simulación
9. Consultoría Gobierno en Línea

16. Teletrabajo
17. Gestión sostenible de residuos

VI. La tipología del sector al que se aplica el proyecto:

1. Desarrollo a medida
2. Portales, redes sociales Web 2.0
3. Integración de Redes
4. Infraestructura Tecnológica
5. Desarrollo de aplicaciones Web
6. E-commerce
7. Integración de sistemas
8. Animación digital
9. Computación en la nube
10. E-marketing

11. Gestión documental
12. Capacitación y entrenamiento
13. Seguridad informática
14. E-learning
15. Soporte y mantenimiento de software
16. Aplicaciones móviles
17. Tecnologías transversales (crm, erp)
18. Servicios de consultoría
19. Soporte y mantenimiento de infraestructura
20. Sistemas personales (biometría)

### Anexo 003: RESUMEN ANALÍTICO ESPECIALIZADO CON FINES DE PUBLICACIÓN RAE

Tema	“Recopilación y consolidación en una base de datos de los trabajos de grado en el área TIC de las universidades de Villavicencio con programas de ingeniería o tecnología de sistemas”
Autor (es).	Alain Enrique Contreras Califa, Yeisson Hernán Guío Fonseca
Fuente	
Año	2017
Resumen	<p>En este proyecto se referencian algunos de los proyectos de TI elaborados y presentados como trabajos de grado, por los estudiantes de las diferentes universidades privadas y públicas de la ciudad de Villavicencio, en las cuales se desarrolló carrera en el área de ingeniería de sistemas, desarrollo de software, tecnología en redes y afines al área.</p> <p>Se desarrolla esta tarea visitando cada una de las instituciones mencionadas (anexo1) en el trabajo de campo, donde se consultó en los diferentes repositorios y bibliotecas de las universidades que con ayuda del semillero de investigación de la Corporación Universitaria Minuto de Dios Orinoco TIC, nos permitieron acceder a la información de los trabajos de grado presentado en las Universidades de Villavicencio en el Departamento del Meta, en las áreas de tecnología, desarrollo de software y afines.</p> <p>Con el desarrollo de este trabajo se pretende dar conocer los tipos de proyectos y su respectivo enfoque de desarrollo que se dan en las universidades de la ciudad de Villavicencio, estos datos nos darán también una pequeña noción en cuanto a cómo va nuestra región en el desarrollo de proyectos de software en las universidades como opción de grado. De igual manera nos servirá en un futuro como base para formular otros proyectos que su enfoque final sea el de promocionar e invitar a los estudiantes de las diferentes carreras de las universidades con programas afines a las TIC.</p>
Problema de la investigación.	Actualmente no se cuenta con información alguna de cómo van nuestras universidades en cuanto al desarrollo de proyectos y en especial las universidades que tienen programas de Ingeniería de sistemas o programas afines a las TIC y que competen a nuestra línea de aprendizaje.
Objetivos	<p>Recopilar y consolidar trabajos de grado en el área TIC en las universidades con programas de ingeniería o tecnología de sistemas de la ciudad de Villavicencio, Meta.</p> <ul style="list-style-type: none"> <li>• Recopilar información de los proyectos de grado que han sido presentados en las instituciones de educación superior con programas de ingeniería de sistemas o afines a las TIC en la ciudad de Villavicencio.</li> <li>• Consolidar y normalizar en una base de datos la información obtenida, que genere consultas mediante una aplicación web.</li> </ul>
Principales conceptos	Recopilación de proyectos TIC. Líneas de desarrollo de software. Universidades con enfoque TI.
Metodología.	Desarrollar encuestas sobre los lineamientos de desarrollo de software en los repositorios donde se alojan los documentos de estos proyectos. Consultar los documentos físicos o digitales, los objetivos de cada proyecto, sus autores y resumen.
Duración de la investigación.	Nueve (9) meses.
Tipo de investigación.	Investigación aplicada
Población y muestra	La población universidades de Villavicencio con programas afines a las TIC. Muestra Repositorios universidades Con programas afines a las TIC.
Argumentos	En el marco y auge en el país en el desarrollo de las tecnologías de la información y la comunicación, es necesario conocer los proyectos planteados y desarrollados en las instituciones de educación superior. Es muy importante tener una noción de cómo va nuestra región en cuanto al desarrollo de software y el desarrollo de investigaciones sobre el mismo.
Conclusiones.	Son muy pocos los estudiantes que toman el desarrollo de una investigación o proyecto como opción de grado. En las universidades se debe invitar a retomar la investigación y el desarrollo de proyectos.

**Tabla 5.** Tabla de informe publicación RAE.

**Fuente:** Tabla diseñada por los desarrolladores del proyecto.