

# UNIVERSIDAD PÚBLICA DE EL ALTO

## CARRERA INGENIERÍA DE SISTEMAS



### TESIS DE GRADO

**MINERÍA DE TEXTO DE LA WEB APLICADO EN REDES SOCIALES  
INFLUENCIADO EN TEMAS POLÍTICOS  
CASO: FACEBOOK**

**PARA OPTAR AL TÍTULO DE LICENCIATURA EN INGENIERÍA DE  
SISTEMAS  
MENCIÓN: INFORMÁTICA Y COMUNICACIONES**

**POSTULANTES** : GILDA AQUINO CASTRO  
: ADALID ALANOCA RAMIREZ

**TUTOR METODOLÓGICO** : M.SC. ENRIQUE FLORES BALTAZAR  
**TUTOR ESPECIALISTA** : LIC. FREDY ALANOCA COARETI  
**TUTOR REVISOR** : LIC. KATYA M. PEREZ MARTINEZ

**EL ALTO – BOLIVIA  
2020**

## DEDICATORIA

La presente tesis de grado va dedicado a Dios y a nuestros padres. A Dios porque ha estado con nosotros en cada paso que hemos dado, cuidándonos y dándome fortaleza para continuar, a nuestros padres, quienes a lo largo de nuestras vidas han velado por nuestro bienestar y educación siendo nuestro apoyo en todo momento. Depositando su entera confianza en cada reto que se nos presentaba sin dudar ni un solo momento en nuestra capacidad. Es por ello que ahora podemos concluir una etapa muy importante en esta vida.

*Gilda Aquino Castro*

*Adalid Alanoca Ramirez*

## **AGRADECIMIENTO**

Le agradecemos a Dios por habernos permitido vivir hasta este día, haberme guiado a lo largo de nuestras vidas, por ser nuestro apoyo. Por habernos dado la fortaleza para seguir adelante en aquellos momentos de debilidad. Le damos gracias a nuestros padres Cecilio Alanoca Tarqui, Gabino Aquino Aquino y Tereza Ramirez Alaro, Elena Castro Fernandez por todo el apoyo que nos han brindado a lo largo de nuestras vidas y por darnos la oportunidad de estudiar y conocer el mundo. A nuestros hermanos por apoyarnos en aquellos momentos de necesidad. A todos los docentes que tuve en la carrera que nos colaboraron a seguir adelante y sobre todo a nuestros Tutores M.Sc Ing. Enrique Flores Baltazar, Lic. Fredy Alanoca Cuareti y Lic. Katya M. Perez Martinez por su visión crítica y sus conocimientos en muchos aspectos cotidianos de la vida, por su rectitud en su profesión como docente, su amistad y por sus consejos que nos ayudaron a formarnos como personas, realmente estamos muy agradecidos, por esfuerzo y dedicación, su experiencia, su paciencia y su motivación ha logrado que pudiéramos concluir con nuestros estudios con éxito. A nuestros amigos Marlien, Juan, Alvaro e Ivan por ser parte muy importante de nuestras vidas, por el apoyo recibido desde el día que nos conocimos llegan a ser como unos hermanos para nosotros, les quisiera agradecer por todos los momentos que pasamos juntos, a los administrativos y docentes que fueron de la mejor manera durante nuestro estudio en la carrera Ingeniería de Sistemas. Quisiéramos agradecer a las diferentes organizaciones que nos ayudaron a crecer especialmente a Posgrado donde aprendimos mucho del ámbito laboral.

## **RESUMEN**

El presente trabajo pretende analizar los comentarios de las publicaciones de su página de Facebook de la dirección de Posgrado Universidad Pública de El Alto implementando la minería de texto para encontrar patrones o relaciones.

Para realizar este trabajo se utilizará el proceso Knowledge Discovery in Databases para analizar datos de texto no estructurados; para la etapa de recolección de datos se utilizará dos herramientas la primera es una extensión para navegadores Data Scraper, que extrae código HTML de forma automática de cualquier página web, la segunda es la API de Facebook que nos da acceso para extraer datos de la página de Posgrado. Ambas herramientas nos ayudarán a obtener los comentarios para después pasar a la etapa de preparación de datos en la que se utilizara algoritmo de RapidMiner para el procesamiento de los datos, en donde se extraerán los datos necesarios para el estudio objetivo.

**Palabras clave:** kdd, api, clustering, rapidminer, Facebook, k-means.

## **ABSTRACT**

The present work tries to analyze the comments of the publications of its Facebook page of the management of the Postgraduate Public University of El Alto, implementing text mining to find patterns or relationships.

To perform this work, the Knowledge Discovery in Databases process will be used to analyze unstructured text data; For the data collection stage two tools will be used the first is an extension for Data Scraper browsers, which automatically extracts HTML code from any web page, the second is the Facebook API that gives us access to extract data from the page Graduate. Both tools will help us to obtain the comments and then proceed to the data preparation stage in which the RapidMiner algorithm will be used for data processing, where the necessary data will be extracted for the objective study.

**Keywords:** kdd, api, clustering, rapidminer, Facebook, k-means.

## INDICÉ GENERAL

<b>DEDICATORIA .....</b>	<b>I</b>
<b>AGRADECIMIENTOS .....</b>	<b>II</b>
<b>RESUMEN .....</b>	<b>III</b>
<b>ABSTRACT .....</b>	<b>IV</b>
<b>CAPITULO I: MARCO INTRODUCTORIO .....</b>	<b>1</b>
<b>CAPITULO II: MARCO TEÓRICO .....</b>	<b>13</b>
<b>CAPITULO III: MARCO APLICATIVO .....</b>	<b>87</b>
<b>CAPITULO IV: CALIDAD Y COSTOS DEL SOFTWARE .....</b>	<b>134</b>
<b>CAPITULO V: PRUEBAS Y RESULTADOS .....</b>	<b>152</b>
<b>CAPITULO VI: CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>163</b>
<b>BIBLIOGRAFÍA</b>	
<b>ANEXOS</b>	
<b>MANUAL DEL ADMINISTRADOR</b>	
<b>MANUAL DEL VISOR</b>	

## ÍNDICE ESPECIFICO

<b>1.</b>	<b>MARCO PRELIMINAR</b> .....	<b>1</b>
<b>1.1.</b>	<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>1.2.</b>	<b>ANTECEDENTES</b> .....	<b>1</b>
<b>1.3.</b>	<b>PLANTEAMIENTO DEL PROBLEMA</b> .....	<b>4</b>
<b>1.4.</b>	<b>Descripción del problema</b> .....	<b>4</b>
<b>1.4.1.</b>	<b>Problema principal</b> .....	<b>5</b>
<b>1.4.2.</b>	<b>Problemas secundarios</b> .....	<b>5</b>
<b>1.4.3.</b>	<b>Formulación de pregunta</b> .....	<b>5</b>
<b>1.5.</b>	<b>OBJETIVOS</b> .....	<b>5</b>
<b>1.5.1.</b>	<b>Objetivo General</b> .....	<b>5</b>
<b>1.5.2.</b>	<b>Objetivos Específicos</b> .....	<b>6</b>
<b>1.6.</b>	<b>HIPÓTESIS</b> .....	<b>6</b>
<b>1.7.</b>	<b>JUSTIFICACIÓN</b> .....	<b>7</b>
<b>1.7.1.</b>	<b>Justificación Económica</b> .....	<b>7</b>
<b>1.7.2.</b>	<b>Justificación Social</b> .....	<b>7</b>
<b>1.7.3.</b>	<b>Justificación Científica</b> .....	<b>7</b>
<b>1.8.</b>	<b>METODOLOGÍA</b> .....	<b>8</b>
<b>1.8.1.</b>	<b>Método Científico</b> .....	<b>8</b>
<b>1.8.2.</b>	<b>Método de Desarrollo</b> .....	<b>8</b>
<b>1.8.2.1.</b>	<b>Metodología UWE</b> .....	<b>8</b>
<b>1.8.3.</b>	<b>Método de Ingeniería</b> .....	<b>8</b>
<b>1.8.3.1.</b>	<b>Metodología CRISP-DM</b> .....	<b>8</b>
<b>1.8.4.</b>	<b>Metodología de Evaluación de Software</b> .....	<b>8</b>
<b>1.8.4.1.</b>	<b>WebQEM</b> .....	<b>8</b>

1.8.5.	Estimación de Costos.....	9
1.8.5.1.	Cocoma II .....	9
1.9.	HERRAMIENTAS.....	9
1.9.1.	Lenguaje de Programación .....	9
1.9.1.1.	Framework .....	10
1.9.2.	Base de Datos.....	10
1.9.3.	Servidor .....	10
1.9.4.	Minería de Datos .....	10
1.10.	LÍMITES Y ALCANCES .....	10
1.10.1.	Limites .....	10
1.10.2.	Alcances .....	11
1.11.	APORTES .....	11
2.	MARCO TEÓRICO.....	13
2.1.	INTRODUCCIÓN.....	13
2.2.	CONCEPTOS BÁSICOS.....	14
2.3.	DATO.....	15
2.4.	INFORMACIÓN.....	15
2.5.	CONOCIMIENTO.....	16
2.6.	PATRONES DE DISEÑO.....	17
2.6.1.	¿Qué son los patrones de diseño de software?.....	17
2.6.2.	¿Por qué son útiles los patrones de diseño?.....	18
2.6.3.	¿Cómo identificar qué patrón encaja con tu problema? .....	19
2.6.4.	Anti patrones .....	19
2.7.	ALGORITMO.....	20
2.7.1.	Definición.....	20

2.7.2.	Características de los algoritmos.....	21
2.8.	<b>MINERÍA DE DATOS.....</b>	21
2.8.1.	Definición de minería de datos.....	22
2.8.2.	Métodos de minería de datos .....	23
2.8.2.1.1.	Clustering .....	24
2.8.3.	Procesos de la Minería de Datos .....	25
2.8.3.1.	Selección de Datos .....	26
2.8.3.2.	Preprocesamiento de Datos.....	26
2.8.3.3.	Extracción del Conocimiento.....	27
2.8.3.4.	Evaluación e Interpretación de Patrones .....	29
2.8.3.5.	Descubrimiento de Conocimiento en Bases de Datos.....	29
2.8.4.	Algoritmos de minería de datos .....	30
2.9.	<b>METODOLOGÍA CRISP-DM.....</b>	31
2.9.1.	Ciclo de vida de la metodología CRISP-DM .....	31
2.9.2.	Fases de la metodología.....	32
2.10.	<b>INGENIERÍA DE SOFTWARE.....</b>	41
2.10.1.	Ingeniería de Software Orientado a Objetos.....	42
2.10.2.	Proceso del Software .....	43
2.10.2.1.	El ciclo de desarrollo de la arquitectura.....	46
2.10.3.	Arquitectura Cliente – Servidor .....	47
2.10.3.1.	El Esquema de funcionamiento de un Sistema Cliente/Servidor: .....	47
2.10.3.2.	Componentes de la Arquitectura Cliente – Servidor.....	48
2.10.3.3.	Elementos principales .....	48
2.10.3.3.1.	Cliente.....	48
2.10.3.3.2.	Servidor .....	49

2.10.3.4.	<b>Tipos de Arquitectura Cliente – Servidor</b> .....	49
2.11.	<b>HERRAMIENTAS DE LA APLICACIÓN</b> .....	51
2.11.1.	<b>PHP</b> .....	51
2.11.2.	<b>JavaScript</b> .....	52
2.11.3.	<b>Bootstrap</b> .....	53
2.11.4.	<b>Ajax</b> .....	53
2.11.5.	<b>jQuery</b> .....	55
2.11.6.	<b>HTML</b> .....	56
2.11.7.	<b>CodeIgniter</b> .....	57
2.11.7.1.	<b>Aplicación de CodeIgniter</b> .....	58
2.11.7.2.	<b>Definición de MVC según CodeIgniter</b> .....	59
2.11.8.	<b>PostgreSQL</b> .....	60
2.11.8.1.	<b>Ventajas de PostgreSQL</b> .....	62
2.11.9.	<b>Servidor</b> .....	63
2.11.10.	<b>Minería de Datos</b> .....	63
2.11.10.1.	<b>Formato de Archivos</b> .....	64
2.11.10.2.	<b>Operadores</b> .....	64
2.11.10.3.	<b>Interfaz Grafica</b> .....	65
2.12.	<b>MÉTODO CIENTÍFICO</b> .....	66
2.12.1.	<b>Fases del Método Científico</b> .....	67
2.13.	<b>METODOLOGÍA UWE</b> .....	67
2.13.1.	<b>Características</b> .....	68
2.13.2.	<b>Meta Modelo de UWE</b> .....	68
2.13.3.	<b>Actividades de modelado de UWE</b> .....	69
2.13.3.1.	<b>Escenarios Web</b> .....	70

2.13.3.2.	Diagramas.....	70
2.13.4.	Fases de Uwe.....	70
2.13.4.1.	Fase Modelo de Análisis de Requerimiento.....	71
2.13.4.2.	Fase Modelo de Conceptual .....	72
2.13.4.3.	Fase Modelo de Navegación .....	73
2.13.4.3.1.	Modelo del espacio de navegación .....	74
2.13.4.3.2.	Modelo de la estructura de navegación .....	74
2.13.4.4.	Fase Modelo de Diseño de Presentación .....	75
2.14.	MÉTRICAS DE CALIDAD DE SOFTWARE .....	76
2.14.1.	WebQem .....	76
2.14.1.1.	Características de calidad .....	77
2.14.1.1.1.	Usabilidad .....	78
2.14.1.1.2.	Funcionalidad .....	78
2.14.1.1.3.	Confiabilidad .....	80
2.14.1.1.4.	Mantenimiento .....	81
2.14.1.1.5.	Portabilidad .....	82
2.15.	COSTOS DE SOFTWARE.....	83
2.15.1.	Cocomo II .....	83
2.15.1.1.	Nivel Post-arquitectura .....	84
3.	MARCO APLICATIVO .....	87
3.1.	DESCRIPCIÓN DEL MODELO .....	87
3.2.	FASE DE INICIO .....	88
3.2.1.	Descripción de Actores del Caso de Uso General.....	89
3.2.2.	Modelo De Requerimientos.....	90
3.2.2.1.	Descripción de Requerimientos a Nivel Técnico .....	91

3.2.2.2.	Descripción de Requerimientos a Nivel de Sistema .....	91
3.2.2.3.	Descripción de Requerimientos a Nivel de Usuario .....	92
3.3.	FASE DE ELABORACIÓN .....	92
3.3.1.	Modelado de Análisis .....	93
3.3.1.1.	Modelo De Casos De Uso.....	93
3.3.1.3.	Descripción de Casos de Uso.....	93
3.3.2.	Modelo Conceptual .....	102
3.3.3.	Modelo De Navegación.....	103
3.3.4.	Modelo De Presentación.....	104
3.3.4.1.	Modelo De Presentación De Iniciar Sesión.....	104
3.3.4.2.	Recuperar Contraseña .....	105
3.3.4.3.	Registro de Usuario .....	106
3.3.4.4.	Página Principal .....	107
3.4.	MODELO CRISP-DM.....	108
3.4.1.	COMPRESIÓN DEL NEGOCIO .....	108
3.4.1.1.	Contexto.....	108
3.4.1.2.	Objetivos de Negocio .....	108
3.4.1.3.	Inventario de Recursos.....	108
3.4.2.	COMPRESIÓN DE LOS DATOS .....	109
3.4.2.1.	Recolección de datos iniciales .....	109
3.4.2.2.	Descripción de los datos .....	109
3.4.3.	PREPARACIÓN DE LOS DATOS .....	109
3.5.	DESARROLLO E IMPLEMENTACIÓN .....	115
3.5.1.	Diseño de vistas del sistema.....	115
3.5.2.	Implementación de las vistas del sistema .....	120

<b>3.5.3.</b>	<b>Implementación de la Base de Datos .....</b>	<b>127</b>
<b>3.5.3.1.</b>	<b>Diccionario de datos .....</b>	<b>128</b>
<b>4.</b>	<b>CALIDAD Y COSTOS DEL SOFTWARE.....</b>	<b>134</b>
<b>4.1.</b>	<b>MÉTRICAS DE CALIDAD DE SOFTWARE .....</b>	<b>134</b>
<b>4.1.1.</b>	<b>Usabilidad.....</b>	<b>134</b>
<b>4.1.2.</b>	<b>Funcionalidad.....</b>	<b>136</b>
<b>4.1.3.</b>	<b>Portabilidad.....</b>	<b>140</b>
<b>4.1.4.</b>	<b>Confiabilidad.....</b>	<b>141</b>
<b>4.1.5.</b>	<b>Mantenibilidad .....</b>	<b>142</b>
<b>4.1.6.</b>	<b>Resultados.....</b>	<b>143</b>
<b>4.2.</b>	<b>Cocomo II .....</b>	<b>144</b>
<b>4.2.1.</b>	<b>Punto de función.....</b>	<b>144</b>
<b>4.2.1.1.</b>	<b>Número de entradas de usuario.....</b>	<b>144</b>
<b>4.2.1.2.</b>	<b>Número de salidas de usuario .....</b>	<b>145</b>
<b>4.2.1.3.</b>	<b>Número de peticiones de usuario.....</b>	<b>145</b>
<b>4.2.1.4.</b>	<b>Numero de Archivos.....</b>	<b>146</b>
<b>4.2.1.5.</b>	<b>Cálculo de factores de ajustes de la complejidad .....</b>	<b>147</b>
<b>4.2.2.</b>	<b>Costo del software desarrollado.....</b>	<b>148</b>
<b>4.2.3.</b>	<b>Costo de la elaboración del proyecto .....</b>	<b>150</b>
<b>5.</b>	<b>PRUEBAS Y RESULTADOS .....</b>	<b>152</b>
<b>5.1.</b>	<b>INTRODUCCIÓN.....</b>	<b>152</b>
<b>5.2.</b>	<b>PRUEBA DE HIPÓTESIS.....</b>	<b>152</b>
<b>5.2.1.</b>	<b>Pasos de la prueba de hipótesis.....</b>	<b>152</b>
<b>5.3.</b>	<b>PRUEBAS.....</b>	<b>153</b>
<b>5.3.1.</b>	<b>Resultados.....</b>	<b>156</b>

<b>5.3.2.</b>	<b>Evaluación de resultados.....</b>	<b>157</b>
<b>5.4.</b>	<b>DETERMINACIÓN DE LA REGIÓN CRITICA.....</b>	<b>158</b>
<b>6.</b>	<b>CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>163</b>
<b>6.1.</b>	<b>ESTADO DE HIPÓTESIS .....</b>	<b>163</b>
<b>6.2.</b>	<b>ESTADO DE LOS OBJETIVOS .....</b>	<b>163</b>
<b>6.3.</b>	<b>RECOMENDACIONES .....</b>	<b>164</b>

## ÍNDICE DE FIGURAS

### CAPITULO II

Figura 2. 1 Esquema del desarrollo de algoritmos .....	20
Figura 2. 2 Representación de un algoritmo como un sistema de información.....	21
Figura 2. 3 Modelo Descriptivo y Predictivo.....	23
Figura 2. 4 Clustering o Búsqueda de Conglomerados .....	25
Figura 2. 5 Procesos de Minería de Datos .....	25
Figura 2. 6 Esquema de los cuatro niveles de Crisp-Dm.....	32
Figura 2. 7 Ciclo de vida Crisp-Dm .....	33
Figura 2. 8 Fase de comprensión del negocio.....	34
Figura 2. 9 Fase de comprensión de los datos .....	35
Figura 2. 10 Fase de Preparación de los datos .....	36
Figura 2. 11 Fase de modelado .....	38
Figura 2. 12 Fase de evaluación.....	39
Figura 2. 13 Fase de Implantación .....	41
Figura 2. 14 Características del proceso general de software .....	43
Figura 2. 15 Flujo de proceso lineal .....	44
Figura 2. 16 Flujo de proceso Iterativo .....	44
Figura 2. 17 Flujo de proceso evolutivo .....	45
Figura 2. 18 Flujo de proceso paralelo .....	45
Figura 2. 19 Cliente-Servidor .....	47
Figura 2. 20 Así funciona PHP .....	51
Figura 2. 21 Flujo de datos Codeigniter .....	58
Figura 2. 22 Meta Modelo .....	69
Figura 2. 23 Vista general de Modelos Uwe .....	71
Figura 2. 24 Modelo de Caso de Uso .....	72
Figura 2. 25 Diagrama de contenido .....	73
Figura 2. 26 Diagrama de navegación UML .....	74
Figura 2. 27 Nombre y símbolo de estereotipos Modelo de Navegación.....	75

Figura 2. 28 Nombre y símbolo de estereotipos Modelo de Presentación .....	75
Figura 2. 29 Diseño de presentación UWE.....	76
Figura 2. 30 Cocomo II.....	84

### **CAPÍTULO III**

Figura 3. 1 Esquema del sistema.....	88
Figura 3. 2 Caso de Uso General .....	89
Figura 3. 3 Iniciar Sesión.....	94
Figura 3. 4 Gestión de Usuario .....	96
Figura 3. 5 Gestionar Base de Datos .....	97
Figura 3. 6 Genera Reporte o Informe.....	99
Figura 3. 7 Gestión Búsqueda textual.....	100
Figura 3. 8 Modelo Conceptual.....	102
Figura 3. 9 Modelo Navegacional.....	103
Figura 3. 10 Modelo de presentación de Iniciar Sesión .....	104
Figura 3. 11 Recuperar Contraseña .....	105
Figura 3. 12 Registro de Usuario .....	106
Figura 3. 13 Página Principal.....	107
Figura 3. 14 Preparación de los Datos.....	110
Figura 3. 15 Selección de Datos.....	110
Figura 3. 16 Archivo Excel.....	111
Figura 3. 17 Insertando Registro a la Base de Datos.....	112
Figura 3. 18 Cargado de los datos en RapidMiner.....	113
Figura 3. 19 Normalización en RapidMiner .....	114
Figura 3. 20 Diseño Login .....	115
Figura 3. 21 Diseño de Pagina Principal .....	116
Figura 3. 22 Diseño de Explorar Comentarios.....	117
Figura 3. 23 Diseño de Cargar Base de Datos .....	118
Figura 3. 24 Diseño de Reportes o Información .....	119
Figura 3. 25 Implementación de Login .....	120

Figura 3. 26 Fragmento de Código Login .....	121
Figura 3. 27 Implementación de Página Principal .....	121
Figura 3. 28 Fragmento de Código Pagina principal.....	122
Figura 3. 29 Diseño de Vizualizaciones Perfil.....	122
Figura 3. 30 Fragmento de Código Explorar Comentario .....	123
Figura 3. 31 Resultados de la Minería de Texto .....	123
Figura 3. 32 Fragmento de Código Usuarios que Comentan.....	124
Figura 3. 33 Login API de Facebook .....	124
Figura 3. 34 Fragmento de Código de Conectar con Api de Facebook .....	125
Figura 3. 35 Ranking de Usuarios Frecuentes .....	126
Figura 3. 36 Fragmento de Código Visualizaciones .....	126
Figura 3. 37 Diagrama Entidad Relación .....	127

## **CAPITULO V**

Figura 5. 1 Pruebas del prototipo .....	153
Figura 5. 2 Pruebas del prototipo .....	154
Figura 5. 3 Pruebas del prototipo .....	154
Figura 5. 4 Pruebas del prototipo .....	155
Figura 5. 5 Región critica para la hipotenusa .....	158
Figura 5. 6 Función de distribución.....	159
Figura 5. 7 Distribución de Zc para la toma de decisiones.....	160

## ÍNDICE DE TABLAS

### CAPITULO II

Tabla 2. 1 Criterios de Evaluación .....	77
--	----

### CAPITULO III

Tabla 3. 1 Descripción de Actores del Caso de Uso General .....	90
Tabla 3. 2 Iniciar Sesión .....	94
Tabla 3. 3 Gestionar datos del Usuario .....	96
Tabla 3. 4 Gestionar Base de Datos .....	98
Tabla 3. 5 Generar Reporte o Informe.....	99
Tabla 3. 6 Gestionar Búsqueda Textual.....	101
Tabla 3. 7 Tabla fb_usuario .....	128
Tabla 3. 8 Tabla fb_persona.....	128
Tabla 3. 9 Tabla fb_grupo_usuario .....	129
Tabla 3. 10 Tabla fb_grupo.....	129
Tabla 3. 11 Tabla fb_publicacion.....	129
Tabla 3. 12 Tabla fb_comentario .....	130
Tabla 3. 13 Vista fb_view_usuarios .....	130
Tabla 3. 14 Tabla fb_correo_electronico.....	131
Tabla 3. 15 Tabla fb_media .....	131
Tabla 3. 16 Tabla fb_reaccion_publicacion.....	131
Tabla 3. 17 Tabla fb_data_politic .....	132

### CAPITULO IV

Tabla 4. 1 Usabilidad del sistema.....	134
Tabla 4. 2 Entradas para el cálculo de funcionalidad.....	136
Tabla 4. 3 Cálculo de punto de función sin ajustar .....	137
Tabla 4. 4 Valores de ajuste de complejidad .....	138

Tabla 4. 5 Ajuste de complejidad de punto función.....	138
Tabla 4. 6 Datos .....	142
Tabla 4. 7 Resultado de métricas de calidad.....	143
Tabla 4. 8 Entradas del prototipo .....	144
Tabla 4. 9 Sistemas gestionados al usuario.....	145
Tabla 4. 10 Peticiones que hace el usuario al prototipo .....	145
Tabla 4. 11 Salidas lógicas del prototipo.....	146
Tabla 4. 12 Interfaz parámetros de medición.....	146
Tabla 4. 13 Factor de complejidad .....	147
Tabla 4. 14 Factor LCD/PF de lenguaje de programación .....	148
Tabla 4. 15 Costo de elaboración del proyecto.....	150

## **CAPITULO V**

Tabla 5. 1 Tabla de pruebas realizadas.....	156
Tabla 5. 2 Pruebas realizadas en el usuario .....	156
Tabla 5. 3 Pruebas realizadas emisión de reportes.....	157

# CAPITULO I



## MARCO PRELIMINAR

### 1.1. INTRODUCCIÓN

La minería de datos consiste en la extracción no trivial de información que reside de manera implícita en los datos. Dicha información previamente desconocida puede resultar útil para algún proceso no previsto hasta ese entonces (Choque, 2009).

Las bases de datos posibilitan almacenar información que es útil a una organización entidad, siempre y cuando estas lo sepan descubrir e interpretar de manera adecuada para una toma de decisiones. Sin embargo, dentro de estas enormes masas de datos existe una gran cantidad de información oculta, de gran importancia estrategia y la cual no es accesible por las técnicas clásicas de recuperación de la información (Vallejos, 2006).

El avance tecnológico en los últimos años ha dado paso al uso de internet y con ello se hizo popular las redes sociales. Por ende, ellas han acumulado una gran cantidad de datos, la información se ha convertido en uno de los recursos más importantes en estos tiempos y es necesario estudiarlo con detalle, es por lo cual la minería de texto se convirtió en uno de los temas más relevantes

### 1.2. ANTECEDENTES

#### a) **Ámbito Internacional**

- La tesis realizada por Luis Piñon Ferrer (septiembre de 2018 –Universitat Jaume I) titulada **MINERÍA DE DATOS APLICADA A TWITTER Y ANÁLISIS DE SENTIMIENTOS MEDIANTE ALGORITMOS DE INTELIGENCIA ARTIFICIAL**, En este documento se explican los fundamentos teóricos de las

principales técnicas de minería de datos e inteligencia artificial aplicados al análisis de texto procedente de redes sociales. Se introducen los principales algoritmos de Análisis de Asociaciones, Clustering (Agrupamiento) y Clasificación y Predicción. También se realizan dos aplicaciones prácticas: una para el sondeo de redes sociales para analizar los sentimientos de los usuarios respecto a cierto tema y otra para la búsqueda de potenciales clientes que muestran interés o buscan activamente soluciones a sus problemas.

- La tesis realizada por Ana Gonzales Marcos (18 de diciembre de 2015 España – Universidad de la Rioja) titulada **DESARROLLO DE TÉCNICAS DE MINERÍA DE DATOS EN PROCESOS INDUSTRIALES MODELIZACIÓN EN LÍNEAS DE PRODUCCIÓN DE ACCESO**, se enfoca a las plantas industriales más allá de subsistir buscar líderes en un ambiente competitivo, instrumento con el que cuenta, en característica común de los procesos industriales actuales es el constante y rápido crecimiento de su capacidad para almacenar datos, es decir día a día se dispone de mayores volúmenes de históricos que contienen información acerca de dichos procesos productivos.
  
- La tesis realizada por Magdalena Servente (20 de febrero de 2002 Buenos Aire – Universidad de Buenos Aires) titulada **ALGORITMOS TDIT APLICADOS A LA MINERÍA DE DATOS INTELIGENTES**, se enfoca en que los programas computacionales son sistemas de aprendizaje capaces de adquirir conocimientos de alto nivel o estrategias para la resolución en forma análoga a la mente humana, el conocimiento previo en el sistema de aprendizaje crea descripciones generales de

conceptos. Servente propuso un sistema cuyo objetivo el cual permitir que el usuario realice una Minería de Datos sobre las tablas de su elección.

#### **b) Ámbito Nacional**

- La tesis realizada por Edwin Percy Marquez Granado (08 de Octubre de 2008 La Paz Universidad Mayor de San Andrés) titulada **DESARROLLO DE UN MODELO DE MINERÍA DE DATOS ACADÉMICO**, En el proyecto describe que la mayoría de las organizaciones realizan cientos de operaciones o cual implica el almacenamiento de grandes cantidades de datos utilizando para ello bases de dato entonces surge la necesidad de buscar herramientas que ayuden a la búsqueda de información útil de las bases de datos mediante la minería de datos. Para tal efecto la organización cuenta con una base de datos de gestión académica en el cual almacena datos académicos de los estudiantes, que con la aplicación de un modelo de minería de datos se extrae información útil, la cual ayuda a los administradores en la toma de decisiones.
  
- La tesis realizada por Gliseth Rojas Fernandez (23 de septiembre de 2015 La Paz Universidad Mayor de San Andrés) titulada **MINERÍA DE DATOS EN EL SISTEMA DE INFORMACIÓN EDUCATIVA**, en el trabajo realizado por Rojas se describe la aplicación de minería de datos sobre los datos de educación formal del sistema de información Educativa (SIE) del ministerio de educación y cultural para descubrir patrones de comportamiento acerca del sector educativo. Para tal efecto se lleva a cabo la minería de datos para el descubrimiento de conocimiento contenidos en bases de datos, las etapas del proceso incluyen la selección, el procesamiento y la transformación. Se desarrolló una herramienta de minería de dato que incluye la

visualización de datos de forma multidimensional que constituye una herramienta visual de análisis y exploratorio y el algoritmo k-medios de Clustering.

**c) Ámbito Local**

- La tesis realizada por Enrique Flores Baltazar (febrero de 2013 El Alto - Universidad Pública del Alto UPEA) titulada **GEO CODIFICACIÓN PARA EL RESUMEN AUTOMÁTICO DE TEXTOS BAJO FORMAS DE MAPAS – CASO: GEO BOLIVIA VICEPRESIDENCIA DEL ESTADO PLURINACIONAL DE BOLIVIA**, donde se describe la aplicación de Minería de Datos aplicado al ámbito geográfico, para descubrir patrones en un documento (de diferentes formatos), y luego extraer los nombres propios de lugares descritos en el documento y luego geocodificarlos (posición de un espacio) posteriormente graficarlos en un mapa. Se presenta un modelo de minería de datos aplicado a datos espaciales (información geográfica), y se presenta la interpretación de los resultados encontrados en la tesis

### **1.3. PLANTEAMIENTO DEL PROBLEMA**

#### **1.4. Descripción del problema**

Actualmente la dirección de **Posgrado Universidad Pública de El Alto** no tiene un programa informático o un aplicativo web para el análisis de patrones y relaciones de los comentarios de su página de Facebook que hacen referencia a temas políticos en sus opiniones lo que ocasiona no poder monitorear a los perfiles de Facebook que comentan de manera inapropiada y con intereses políticos en la página de Posgrado.

### 1.4.1. Problema principal

La dirección de **Posgrado Universidad Pública de el Alto** no cuenta con un sistema web que permita el análisis de patrones y relaciones de los comentarios de su qué hacen referencia a temas políticos en sus opiniones, la dirección de Posgrado ubicada en el edificio emblemático.

### 1.4.2. Problemas secundarios

- Búsqueda manual de los comentarios de la página oficial de Posgrado en Facebook
- Inexistencia una de una base de datos estandarizada para guardar el historial de comentarios de cada perfil.
- No se llega a analizar la totalidad de los comentarios realizados en la página de Facebook sobre una publicación.
- Proporcionar un módulo de representación e interpretación de la información obtenida.

### 1.4.3. Formulación de pregunta

¿De qué manera el “Sistema Politic Data Mining” automatizara el monitoreo de los comentarios con orientaciones políticas realizados en página oficial de Posgrado?

## 1.5. OBJETIVOS

### 1.5.1. Objetivo General

Desarrollar un sistema web para la dirección de **Posgrado Universidad Pública de el Alto** para el análisis de patrones relaciones de los comentarios de la página oficial de Facebook que hacen referencia a temas políticos en sus opiniones.

### 1.5.2. Objetivos Específicos

- Mejorar la búsqueda manual de los comentarios de la página oficial de Posgrado en Facebook
- Implementar una base de datos estandarizada para guardar el historial de comentarios de cada perfil.
- Analizar la totalidad de los comentarios realizados en la página de Facebook sobre una publicación.
- Realizar la representación e interpretación de la información obtenida.

### 1.6. HIPÓTESIS

La hipótesis planteada es la siguiente propuesta:

**Hi** = La minería de texto nos permite extraer comentarios con fines políticos para analizar las opiniones de la página de posgrado con una eficiencia del 95%.

**Ho** = La minería de texto no nos permite extraer comentarios con fines políticos para analizar las opiniones de la página de posgrado con una eficiencia del 95%.

#### **Variable independiente**

Minería de texto

#### **Variable dependiente**

Extraer comentarios con fines políticos

## **1.7. JUSTIFICACIÓN**

### **1.7.1. Justificación Económica**

Un aplicativo web para el análisis de los comentarios de las publicaciones de Facebook utilizando minería de texto generaría ahorros para las empresas que hagan uso la red social Facebook ya que este estudio genera una gran fuente de información de los perfiles de Facebook que no se pueden ver a simple vista, datos que están escondidos entre los patrones y relaciones de los datos que podría generar mucha más información para la dirección de Posgrado y así dar a conocer nuevas tendencias e información acerca de cómo los usuarios actuales interactúan con la página de Posgrado.

Por otra parte, el conocimiento de los comentarios de las personas en Facebook ayudara a Posgrado a poder plantear una mejor estrategia para los usuarios interesados y evitar los comentarios con tendencias políticas.

### **1.7.2. Justificación Social**

El presente estudio va dirigido a cualquier empresa pública o privada que quisiera conocer las opiniones de las personas en la red social Facebook además de colaborar a obtener una mejor información de los comentarios que se producen en Facebook.

### **1.7.3. Justificación Científica**

El estudio aporta con un aplicativo web con el cual podemos buscar palabras sea cual sea su orientación y esto servirá para el análisis a toda la comunidad de investigadores en las redes sociales intercalados ya que se utiliza la API de Facebook para obtener la información requerida de esta tesis.

## **1.8. METODOLOGÍA**

### **1.8.1. Método Científico**

El método de investigación empleado en este proyecto será, el método científico que nos ayudará a descubrir propiedades del objeto de estudio a obtener nuevos conocimientos, que ha caracterizado históricamente a la ciencia, y que consiste en la observación sistemática, medición, experimentación, y la formulación, análisis y modificación de hipótesis (Francis Bacon).

### **1.8.2. Método de Desarrollo**

#### **1.8.2.1. Metodología UWE**

UWE es una metodología que permite especificar de mejor manera una aplicación Web en su proceso de creación mantiene una notación estándar basada en el uso de UML (Unified Modeling Language) para sus modelos y sus métodos, lo que facilita la transición. La metodología define claramente la construcción de cada uno de los elementos del modelo (Nora Koch, Andreas Kraus).

### **1.8.3. Método de Ingeniería**

#### **1.8.3.1. Metodología CRISP-DM**

La metodología de herramienta de este trabajo se basa principalmente en las etapas del proceso conocido como Knowledge Discovery in Databases (KDD), implementadas de acuerdo a la metodología CRISP-DM, la cual es usada para el desarrollo de proyectos de minería de datos.

### **1.8.4. Metodología de Evaluación de Software**

#### **1.8.4.1. WebQEM**

Metodología de evaluación de calidad de sitios Web (Web-site Quality Evaluation method), diseñada para la evaluación siguiendo seis fases: planificación y programación de la evaluación de

calidad, definición y especificación de requerimientos de calidad, definición e implementación de la evaluación elemental, definición e implementación de la evaluación global, análisis de resultados, conclusión y documentación, validación de métricas (Olsina, 1999).

### **1.8.5. Estimación de Costos**

#### **1.8.5.1. Cocomo II**

Permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de costo y de escala, posee tres modelos Composición de Aplicación, Diseño Temprano y Post-Arquitectura. Considera los enfoques de desarrollo de construcción de prototipos, desarrollo basado en componentes, bases de datos, modelo de desarrollo en espiral, paradigma orientado a objetos. Considera los enfoques de desarrollo de construcción de prototipos, desarrollo basado en componentes, bases de datos, modelo de desarrollo en espiral, paradigma orientado a objetos.

## **1.9. HERRAMIENTAS**

### **1.9.1. Lenguaje de Programación**

**PHP acrónimo** recursivo en inglés de PHP: Hypertext Preprocessor (preprocesador de hipertexto), es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en un documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera el HTML resultante.

### **1.9.1.1. Framework**

**Codeigniter** es un framework para el desarrollo de aplicaciones en php que utiliza el MVC. Permite a los programadores Web mejorar la forma de trabajar y hacerlo a mayor velocidad. Al igual que cualquier framework está pensado para gente que tiene un dominio, al menos medio, del lenguaje de programación PHP.

### **1.9.2. Base de Datos**

**PostgreSQL** es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto, publicado bajo la licencia PostgreSQL, similar a la BSD o la MIT.

### **1.9.3. Servidor**

**Apache** es un acrónimo de «a patchy server» es un servicio de páginas web HTTP de código abierto que sirve para colocar varias plataformas como Unix, BSD, GNU/Linux, Windows, Macintosh entre otros que implementan el protocolo HTTP y el conocimiento o conceptos de sitios virtual y se basó inicialmente en el código NCSA HTTP.

### **1.9.4. Minería de Datos**

**RapidMiner Studio** es una herramienta perfecta para crear modelos y a posterior la realización de análisis predictivos de grandes volúmenes de datos. Se utiliza para realizar análisis de minería de datos (Data Mining) en aplicaciones empresariales, gobierno y academias.

## **1.10. LÍMITES Y ALCANCES**

### **1.10.1. Limites**

El presente proyecto propuesto no puede resolver otro problema que no se encuentre dentro del contexto mencionado.

El prototipo dependerá de la información que provea Facebook.

### **1.10.2. Alcances**

Al ser la minería de datos una herramienta eficiente podremos llegar a abarcar los diferentes módulos que con lleva esta área: Sistemas expertos, sistemas inteligentes, redes neuronales, etc.

- Modelo de entrada de datos.
- Modelo de análisis y descubrimiento de patrones
- Modelo de evaluación

### **1.11. APORTES**

La presente investigación se orienta a la propuesta de un modelo de minería de datos con el fin de estudiar y analizar información proveniente de redes sociales apoyando a fines políticos.



## CAPITULO II

## MARCO TEÓRICO

En este capítulo se representa la teoría relacionada con la minería de datos su definición, objetivos, métodos, técnicas y modelos, también se describe la relación con el descubrimiento del conocimiento en Bases de datos de las herramientas de desarrollo para la propuesta de la solución, descubrimiento de sus componentes estructura, mecanismo de aprendizaje ventajas y aplicaciones.

### 2.1. INTRODUCCIÓN

Según (Mata, 2017) indica que la minería de datos permite encontrar información escondida en los datos que no siempre resulta aparente, ya que, dado el gigantesco volumen de datos existentes, gran parte de ese volumen nunca será analizado.

Además (Perez Marquez, 2014) señala a minería de datos como un conjunto de técnicas encaminadas al descubrimiento de la información contenida en grandes conjuntos de datos. Se trata de analizar comportamientos, patrones, tendencias, asociaciones y otras características del conocimiento inmerso en los datos.

El autor (Daza Vergaray, 2004) dice que la minería de datos se define como aquel proceso que consiste en extraer conocimiento útil y comprensible previamente desconocido, desde grandes

cantidades de datos almacenados en distintos formatos. En ese sentido, la tarea fundamental de la minería de datos es encontrar modelos inteligibles a partir de los datos recogidos.

Por lo tanto, la minería de datos es un avance de técnicas de recuperación de información que su función principal es localizar temas o conceptos de un documento, labor que se lleva a cabo por las técnicas de extracción de información.

## **2.2. CONCEPTOS BÁSICOS**

El autor (Molina Felix, 2002) indica que data mining surge como una tecnología que intenta ayudar a comprender el contenido de una base de datos. De forma general, los datos son la materia prima bruta. En el momento que el usuario les atribuye algún significado especial pasan a convertirse en información.

Además (Vallejos S. , p. 11) señala que, en los últimos años, ha existido un gran crecimiento en nuestras capacidades de generar y coleccionar datos, debido básicamente al gran poder de procesamiento de las máquinas como a su bajo costo de almacenamiento.

Por lo tanto, la minería de textos busca extraer información útil e importante de formatos de documentos heterogéneos, tales como páginas web, correos electrónicos, medios sociales, artículos de revistas, etc.

### **2.3. DATO**

Según (Molina L. C., 2002) los datos son hechos, medidas u observaciones que pueden presentarse, descritos al interior de un contexto. Datos sin contexto son por ejemplo los números 60, 62, 66,72 los mismos datos, ahora con contexto, podrían representar el peso en kilogramos de Laura, Ana, Juan y pedro, respectivamente. La validez y la efectividad de los datos vienen determinadas principalmente por su exactitud

Por su parte (D'Ambrosio, 2020) afirma que la importancia de los datos está en su capacidad de asociarse dentro de un contexto para convertirse en información. Por si mismos los datos no tienen capacidad de comunicar un significado y por tanto no pueden afectar el comportamiento de quien los recibe.

Por lo tanto, los datos son los hechos que describen sucesos y entidades es una palabra en plural que se refiere a más de un hecho.

### **2.4. INFORMACIÓN**

(Chiavenato, 2006) define la información como un conjunto de datos con su significado, es decir, que reduce la incertidumbre y aumenta el conocimiento de algo, la información es un mensaje con significado en un determinado contexto, disponible para uso inmediato y que proporciona orientación a las acciones por el hecho de reducir el margen de incertidumbre con respecto a las decisiones.

El autor (Thompson, 2008) dice que la información es un conjunto de datos acerca de algún suceso, hecho, fenómeno o situación, que organizados en un contexto determinado tienen su significado, cuyo propósito puede ser el de reducir la incertidumbre o incrementar el conocimiento acerca de algo.

Por lo tanto, denominamos información al conjunto de datos, ya procesados y ordenados para su comprensión, que aportan nuevos conocimientos a un individuo o sistema sobre un asunto.

## **2.5. CONOCIMIENTO**

Según (Molina J. , 2001) afirma que el conocimiento es una mezcla fluida de experiencias, valores, información contextual y apreciaciones expertas (Know-how) que proporcionan un marco para su evaluación e incorporación de nuevas experiencias e información, y es útil para la acción.

También (Molina J. , 2001) afirma que la información puede ser convertida en conocimiento partiendo de patrones históricos. De lo anterior, se infieren dos puntos. El primero es que la información es personal, el segundo conocimiento no es estático, es decir, debe cambiar cuando cambia el entorno de decisión.

Por último (Vladimirou, 2005) señala que la información puede ser convertida en conocimiento partiendo de patrones históricos. De lo anterior, se infieren dos puntos. El primero es que la información es personal, el segundo conocimiento no es estático, es decir, debe cambiar cuando cambia el entorno de decisión.

Por tanto, el conocimiento es adquirir información valiosa para comprender la realidad por medio de la razón, el entendimiento y la inteligencia.

## 2.6. PATRONES DE DISEÑO

(Rufenba, 2014) considera que los patrones de diseño son una herramienta muy útil. Cualquier programador debería conocer, por lo menos, los patrones más utilizados. Y es que tenerlos en nuestra caja de herramientas nos puede ahorrar muchos dolores de cabeza.

Por otra parte (Christopher, 2005) afirma que un patrón de diseño (design pattern) es una solución repetible a un problema recurrente en el diseño de software. Esta solución no es un diseño terminado que puede traducirse directamente a código, sino más bien una descripción sobre cómo resolver el problema, la cual puede ser utilizada en diversas situaciones.

Por lo tanto, los patrones de diseño nomina, abstrae e identifica los aspectos clave de una estructura de diseño común, lo que los hace útiles para crear un diseño orientado a objetos reutilizable.

### 2.6.1. ¿Qué son los patrones de diseño de software?

(Sanchez, 2017) apoya la siguiente concepción, pues ni más ni menos son formas “estandarizadas” de resolver problemas comunes de diseño en el desarrollo de software.

Las ventajas del uso de patrones son evidentes:

- Conforman un amplio catálogo de problemas y soluciones
- Estandarizan la resolución de determinados problemas
- Condensan y simplifican el aprendizaje de las buenas prácticas
- Proporcionan un vocabulario común entre desarrolladores
- Evitan “reinventar la rueda”

### 2.6.2. ¿Por qué son útiles los patrones de diseño?

(Leiva, 2016) destaca lo siguiente, puede parecer una tontería, pero si no encuentras utilidad a las cosas acabarás por no usarlas. Los patrones de diseño son muy útiles por los siguientes motivos.

- **Te ahorran tiempo** Sé que te encantará encontrar una solución ingeniosa a un problema cuando estás modelando tu software, y es normal, a mí también me pasa. Como he comentado alguna vez, el desarrollo es un proceso casi artístico, y ese reto mental que supone revierte en una satisfacción personal enorme una vez que consigues un buen resultado. Pero hay que ser sinceros, buscar siempre una nueva solución a los mismos problemas reduce tu eficacia como desarrollador, porque estás perdiendo mucho tiempo en el proceso.
- **Los patrones de diseño atajan ese punto.** Una vez los conozcas, contarás con un conjunto de «trucos», de reglas, de herramientas muy probadas, que te permitirán solucionar la mayor parte de tus problemas de forma directa, sin tener que pensar en cómo de válidas son, o si puede haber una alternativa mejor.
- **Te ayudan a estar seguro de la validez de tu código** Un poco relacionado con lo anterior, siempre que creamos algo nuevo nos surge la duda de si realmente estamos dando con la solución correcta, o si realmente habrá una respuesta mejor. Y el tema es que es una duda muy razonable y que en muchos casos la respuesta sea la que no deseas: sí que hay una solución más válida, y has perdido tu valioso tiempo en implementar algo que, aunque funciona, podría haberse modelado mejor.

- **Establecen un lenguaje común** Todas las demás razones palidecen ante esta. Modelar tu código mediante patrones te ayudará a explicar a otras personas, conozcan tu código o no, a entender cómo has atajado un problema. Además, ayudan a otros desarrolladores a comprender lo que has implementado, cómo y por qué, y además a descubrir rápidamente si esa era la mejor solución o no.

### 2.6.3. ¿Cómo identificar qué patrón encaja con tu problema?

(Leiva, 2016) Afirma que este es el punto más complicado, y la respuesta más evidente, que es también la que menos nos gusta, es que se aprende practicando. La experiencia es la única forma válida de ser más hábil detectando dónde te pueden ayudar los patrones de diseño.

### 2.6.4. Anti patrones

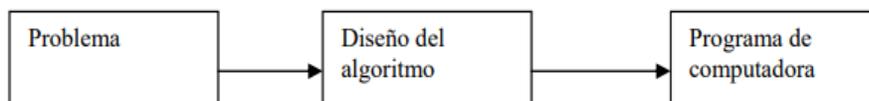
Según (Sanchez, 2017) indica que no todos los patrones iban a ser buenos, existen los llamados anti patrones, una especie de gemelos malvados que nos llevan a implementar malas soluciones.

El autor (Christopher, 2005) dice que los anti patrones, se corresponden a las malas experiencias ya que reúnen soluciones que han producido efectos negativos. Los anti patrones proveen dos soluciones: la problemática (aquella con el impacto negativo), y la refactorizada (aquella que transforma la situación negativa en una más saludable).

Por lo tanto, un anti patrón son las malas prácticas o malas soluciones para los problemas y las estudiamos para poderlos evitar en el futuro.

## 2.7. ALGORITMO

(Balderrama Vasquez, 2017) afirma que un problema es un asunto o conjunto de cuestiones que se plantean para ser resueltas, la naturaleza de los problemas varia con el ámbito o con el contexto donde están planteados: así existen problemas matemáticos, físicos, filosóficos, etc. (véase en la figura 2.1).



*Figura 2. 1 Esquema del desarrollo de algoritmos*

*Fuente: Balderrama Vasquez 2017*

### 2.7.1. Definición

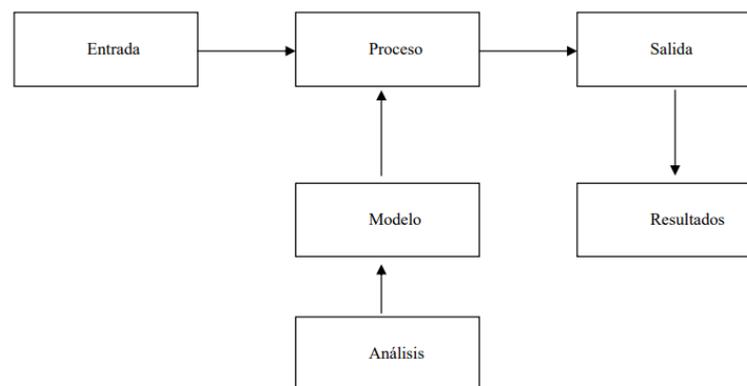
- “Un Algoritmo es una secuencia de operaciones detalladas y no ambiguas que, al ejecutarse paso a paso, conducen a la solución de un problema”. En otras palabras, es un conjunto de reglas para resolver una cierta clase de problema.
- “Algoritmo es un conjunto de instrucciones que especifican la secuencia de operaciones a realizar, en orden, para resolver un sistema específico o clase de problema”.
- “Un Algoritmo es la aplicación de pasos lógicos, secuenciales y metódicamente aplicados para dar solución a un problema en cuestión.” En otras palabras, un algoritmo es una fórmula para resolver problemas.
- “En otras palabras un algoritmo es una fórmula para la solución de un problema.”
- “Todo problema se puede describir por medio de un algoritmo”
- “Todo algoritmo es independiente del lenguaje”

### 2.7.2. Características de los algoritmos.

Las propiedades de un algoritmo son las siguientes:

- El algoritmo debe ser preciso e indicar el orden de realización de cada paso.
- El algoritmo debe ser definido, si se sigue un algoritmo dos veces, se debe obtener el mismo resultado cada vez.
- El algoritmo debe ser finito, si se sigue un algoritmo se debe terminar en algún momento; o sea debe tener un número finito de pasos.

El algoritmo debe ser planteado como un sistema de información (véase en la figura 2.2).



*Figura 2. 2 Representación de un algoritmo como un sistema de información*

*Fuente: Balderrama Vasquez 2017*

## 2.8. MINERÍA DE DATOS

Según (Michalski, 1986) indica que los procesos de minería de datos tienen muchas veces implícitas técnicas de aprendizaje que, de acuerdo con la habilidad de adquirir nuevo conocimiento, desarrollar habilidades para analizar y evaluar problemas mediante métodos y técnicas, así como también por medio de la experiencia propia; se requiere del aprendizaje entendible para un hombre.

Por otra parte (Molina J. , 2001) considera que es el conjunto de técnicas y tecnologías que permiten explorar grandes bases de datos, de manera automática o semiautomática, con el objetivo de encontrar patrones repetitivos, tendencias o reglas que expliquen el comportamiento de los datos en un determinado contexto.

Por último (Gorunescu, 2011) señala que es obvia la incapacidad del hombre de procesar y extraer nueva información de grandes cantidades de datos, por lo que surge un importante campo la Minería de Datos. La Minería de Datos (MD) o KDD como se le comienza a llamar a inicios del año 1996, se define como el proceso de extraer conocimiento útil y comprensible, previamente desconocido, desde grandes cantidades de datos almacenados en distintos formatos.

Por lo tanto, se considera a la Minería de Datos como el proceso de extracción de conocimiento de forma automática o semiautomática, desde grandes cantidades y representa el núcleo fundamental al interior de proceso de descubrimiento de conocimientos en bases de datos.

### **2.8.1. Definición de minería de datos**

(Gonzales Marcos, 2006) define a la minería de datos como el proceso de extraer conocimiento útil y comprensible, previamente desconocido, a partir de grandes volúmenes de datos.

El autor (Piatetsky & Frawley, 2000) dice que la minería de datos es el conjunto de técnicas y herramientas aplicadas al proceso trivial de extraer y presentar el conocimiento implícito, previamente desconocido, potencialmente útil y humanamente comprensible, a partir de grandes conjuntos de datos, con el objeto de predecir de forma autorizada tendencias y comportamientos y/o descubrir de forma automatizada modelos previamente desconocidos.

Por último (Wang, 1999) define que la minería de datos busca patrones de intereses mediante árboles o reglas de clasificación, técnicas de regresión, clusterizado, modelizado secuencial, dependencias.

Por lo tanto, la minería de datos es un proceso de extracción de conocimiento de gran interés de forma automática o semiautomática mediante la aplicación de un conjunto de métodos y técnicas a grandes cantidades de datos.

### 2.8.2. Métodos de minería de datos

(Srikant & Agrawal, 1994) la minería de datos dispone de varios métodos los cuales, aplicados a grandes cantidades de datos, tienen la capacidad de descubrir conocimiento de grandes intereses. La minería de datos genera modelos que pueden ser descriptivos o predictivos (véase en la figura 2.3).

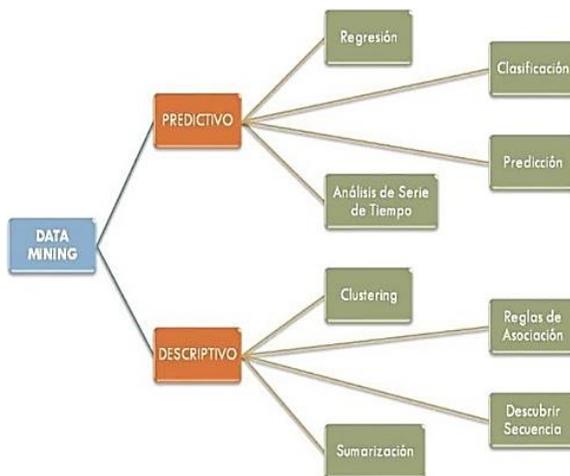


Figura 2. 3 Modelo Descriptivo y Predictivo

Fuente: Rodríguez Oldemar

- **Descriptivos o no supervisados:** este modelo aspira a descubrir patrones y tendencias sobre el conjunto de datos sin tener ningún tipo de conocimiento previo de la situación a la cual se requiere llegar descubrir patrones en los datos analizados y proporciona información sobre las relaciones entre los mismos (Rodríguez, 2013).
  - OLAP (visualización).
  - “Clustering”.
  - Métodos Factoriales como ACP, AFC.
  
- **Predictivos o supervisados:** crean un modelo de una situación donde las respuestas son conocidas y luego es aplicado en otra situación de la cual se desconoce la respuesta. Conociendo y analizando un conjunto de datos, intentan predecir el valor de un atributo (etiqueta), estableciendo relaciones entre ellos. Utiliza algunas de las variables para predecir los valores futuros desconocidos de la misma variable o bien de otras variables (Rodríguez, 2013).

#### 2.8.2.1.1. Clustering

Según (Rodríguez, 2013) clustering es clasificación no supervisada, aprendizaje no supervisado, es similar a la clasificación, excepto que los grupos no son predefinidos. El objetivo es particionar o segmentar un conjunto de datos o individuos en grupos que pueden ser disjuntos o no (véase en la figura 2.4).

Métodos:

- Clasificación Jerárquica (grupos disjuntos).
- Nubes Dinámicas (grupos disjuntos).

- Clasificación Piramidal (grupos NO disjuntos).

### Clustering o Búsqueda de Conglomerados

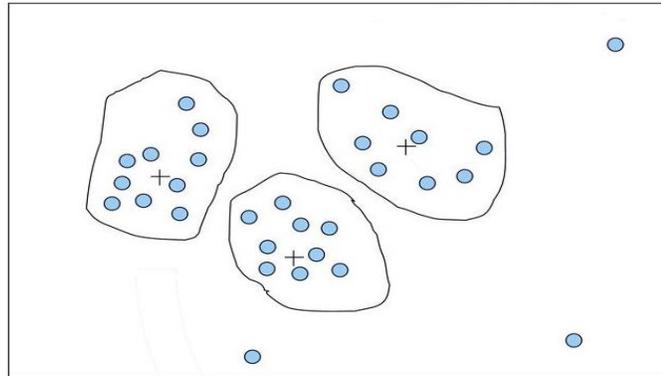


Figura 2. 4 Clustering o Búsqueda de Conglomerados

Fuente: Rodrigues Oldemar

### 2.8.3. Procesos de la Minería de Datos

Según ( Fuentes Alarcon, 2015) El único propósito de proceso de Minería de Datos es extraer conocimientos de grandes bases de datos, esto se lo logra utilizando algoritmos.

Los pasos del proceso de Minería de Datos se describen en la (véase en la figura 2.5).

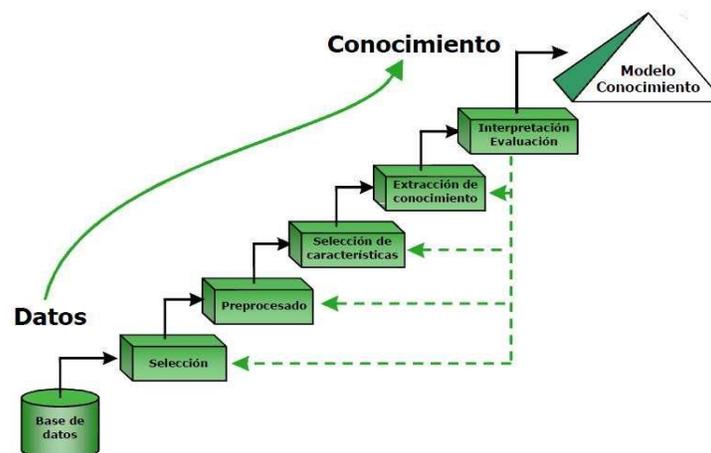


Figura 2. 5 Procesos de Minería de Datos

Fuente: Fuentes Alarcon, 2015

### 2.8.3.1. Selección de Datos

Según ( Fuentes Alarcon, 2015) La selección de datos para el proceso de minería de datos se divide en dos partes, la primera parte consiste en la localización de los datos, esto viene a ser un poco más mecánico en comparación con la segunda parte, que consiste en la identificación de los datos, esta segunda parte requiere de la ayuda de un experto en los datos que se desean analizar. Existen varios métodos para la selección de este subconjunto de datos o atributos.

Entre algunos de ellos se puede citar.

- Selección por pasos hacia adelante: comienza con un conjunto de atributos, en cada paso se agrega al conjunto el mejor atributo.
- Eliminación para los pasos hacia atrás: comienza con un conjunto que posee todos los atributos generales, en cada paso se elimina del conjunto el peor atributo.
- Combinación de selección por pasos hacia adelante y eliminación por pasos hacia atrás: es una combinación de los dos anteriores. Se puede utilizar un umbral de medición para establecer cuando detener la eliminación y agregación de los atributos
- Inducción con árboles de decisión: se utilizan algoritmos como ID y C4.5, son algoritmos cuya misión es la elaboración de un árbol de decisión.

### 2.8.3.2. Preprocesamiento de Datos

Según ( Fuentes Alarcon, 2015) La influencia del preprocesamiento de datos dentro del desempeño de modelos de perfilamiento de clientes es un área de investigación que requiere ser abordada desde el campo de la minería de datos, corresponde a un problema que actualmente es de gran importancia para el mercadeo directo en las organizaciones, se observa que en los últimos

quince años se han realizado estudios con respecto a esta problemática, no obstante estos estudios no identifican una relación directa entre el preprocesamiento de datos aplicado y la calidad de los modelos de perfilamiento, siendo estos modelos aquellos que permiten conocer los rasgos particulares de los clientes en una organización.

Por otra parte (Posted, 2016) El preprocesamiento de datos es un paso preliminar durante el proceso de minería de datos. Se trata de cualquier tipo de procesamiento que se realiza con los datos brutos para transformarlos en datos que tengan formatos que sean más fáciles de utilizar.

Por lo tanto, El objetivo del reprocesador es adecuar los datos para la aplicación a los algoritmos de minería sea optima. Para esto hay que filtrar, eliminar datos incorrectos o no válidos, crear nuevos valores y categorías para atributos e intentar complementar o descartar los valores desconocidos e incompletos

### **2.8.3.3. Extracción del Conocimiento**

Según (Pavez, 2000) conceptualiza el conocimiento se ha convertido en uno de los mejores activos de las organizaciones para obtener una ventaja competitiva en el mercado a largo plazo debido a la poca diferencia en sus desarrollos tecnológicos, a los nuevos paradigmas asociados a la Sociedad de la información y a la Nueva economía basada en el conocimiento. Entre los objetivos de la Gestión del Conocimiento se encuentran los siguientes:

- Formular una estrategia de alcance organizacional para el desarrollo, adquisición y aplicación del conocimiento.
- Implantar estrategias orientadas al conocimiento.

- Promover la mejora continua de los procesos de negocio, enfatizando la generación y utilización del conocimiento.
- Monitorizar y evaluar los logros obtenidos mediante la aplicación del conocimiento.
- Reducir los tiempos de ciclo en el desarrollo de nuevos productos y mejorar los existentes.
- Reducir los costos por repetición de errores.

Por otra parte (Hernandez, 2004) Menciona que una vez que los datos están recopilados integrados y limpios todavía no están listos para aplicar una tarea de Minería de Datos. Es necesario además realizar un reconocimiento o análisis exploratorio de los datos con el objetivo de conocerlos con detalle para tal efecto de cuenta con técnicas diversas algunas técnicas simples del análisis exploratorio de datos técnicas de visualización previa de agrupamiento exploratorio. Según para que un modelo sea válido los patrones que se pueden extraer de él deben tener 3 cualidades: ser precisos, comprensibles (es decir, inteligibles) e interesantes (útiles y novedosos). Si bien este tipo de validación es un tanto subjetiva, se ajusta bien a este tipo de problemas, ya que aquí no tiene sentido otro tipo de validación más matemática como la que se haría con modelos de simulación.

Por lo tanto, mediante una técnica de minería de datos, se obtiene un modelo de conocimiento, que representa patrones de comportamiento observados en los valores de las variables del problema o relaciones de asociación entre dichas variables. También pueden usarse varias técnicas a la vez para generar distintos modelos, aunque generalmente cada técnica obliga a un reprocesados diferente de los datos.

#### **2.8.3.4. Evaluación e Interpretación de Patrones**

Una vez obtenido el modelo, se debe proceder a su validación, comprobando que las conclusiones que arroja son válidas y suficientemente satisfactorias. En el caso de haber obtenido varios modelos mediante el uso de distintas técnicas, se deben comparar los modelos en busca de aquel que se ajuste mejor al problema. Si ninguno de los modelos alcanza los resultados esperados, debe alterarse alguno de los pasos anteriores para generar nuevos modelos (Vallejos S. , 2006).

Menciona que una vez obtenidos los patrones se debe comprobar su validez. Si los modelos son varios, se debe elegir el que se ajuste mejor al problema. Si ninguno de los modelos alcanza los resultados esperados, se debe volver a las etapas anteriores y modificar alguna entrada para generar nuevos modelos (Flores 2002)

#### **2.8.3.5. Descubrimiento de Conocimiento en Bases de Datos**

Según (Lara Tarralbo, 2016) el descubrimiento de las bases de datos (Knowledge Discovery in Databases, KDD), gracias al avance de las herramientas tecnológicas y al incremento del poder de cómputo, permite que se pueda obtener conocimiento a partir de grandes cantidades de información. Este paradigma emergente puede ser utilizado como ayuda para la toma de decisiones.

Por una parte (Fayyad, Piatestky-Shapiro, & Smyth, 1996). Afirma que la etapa de transformación/reducción de datos, se buscan características útiles para representar los datos dependiendo de la meta del proceso. Se utilizan métodos de reducción de dimensiones o de transformación para disminuir el número efectivo de variables bajo consideración o para encontrar representaciones invariantes de los datos

Por otra parte (Dyche, 2000) Conceptualizo que el termino de descubrimiento de conocimiento tienen sus raíces en la inteligencia artificial y el aprendizaje con máquinas.

Por lo encontrar conocimiento relevante y nuevo mediante la utilización de algoritmos eficientes, en grandes bases de datos lo que se considera como conocimiento de acuerdo a la especificación de ciertos parámetros usando una base de datos.

#### 2.8.4. Algoritmos de minería de datos

Con el fin de aproximarnos (Servente, 2002) afirma que la clasificación de datos se desarrolla una descripción o modelo para cada una de las clases presentes en la base de datos. Existen muchos métodos de clasificación como aquellos basados en los árboles de decisión TDIDT como el ID3 y el C4.5, los métodos estadísticos, las redes neuronales, y los conjuntos difusos, entre otros.

Automático que han sido aplicados a la Minería de Datos con cierto éxito:

- **Redes Neuronales:** las redes neuronales imitan la capacidad de la mente humana para encontrar patrones. Han sido aplicadas con éxito en aplicaciones que trabajan sobre la clasificación de los datos.
- **Método del vecino más cercano:** es una técnica que clasifica cada registro de un conjunto de datos en base a la combinación de las clases de los k registros más similares. Generalmente se utiliza en bases de datos históricas.

## **2.9. METODOLOGÍA CRISP-DM**

(Chapman, Clinton, & Kerber, 2000) autores este modelo que es uno de los más utilizados como guía de referencia en el desarrollo de proyectos de minería de datos. La metodología CRISP-DM consiste en un conjunto de tareas que están organizadas en cuatro niveles de abstracción: fases, tareas generales, tareas especializadas e instancias de proceso. Dichos niveles están establecidos respetando jerarquías en tareas; inician en el nivel más general hasta llegar, finalmente, a los casos más específicos

### **2.9.1. Ciclo de vida de la metodología CRISP-DM**

El modelo provee una representación completa del ciclo de vida de un proyecto de minería de datos. El proceso es dinámico e iterativo, por lo que la ejecución de los procesos no es estricta y con frecuencia se puede pasar de uno a otro proceso, de atrás hacia delante y viceversa (Chapman, Clinton, & Kerber, 2000).

Estos dependen del resultado de cada fase o la planeación de la siguiente tarea por ejecutar. Estas fases ayudan a las organizaciones a entender el proceso y proveen de un “mapa del camino” que se debe seguir, así: conocimiento del negocio, conocimiento de los datos, preparación de los datos, modelado, evaluación, despliegue (véase en la figura 2.6).

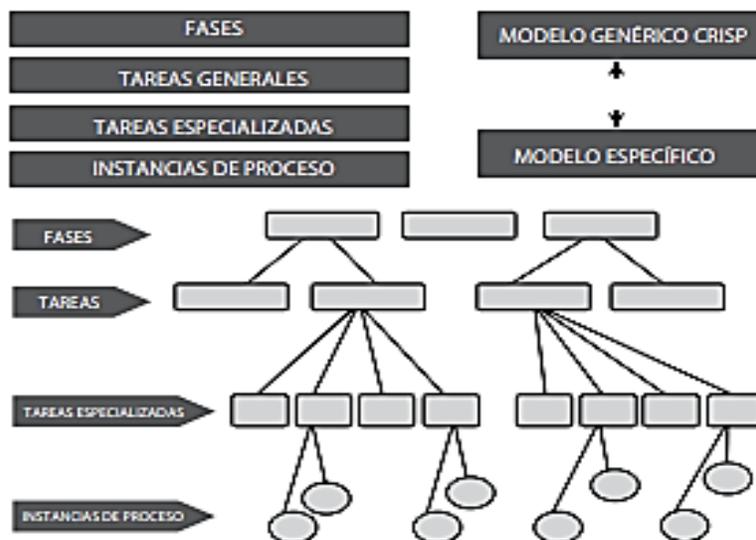


Figura 2. 6 Esquema de los cuatro niveles de Crisp-Dm

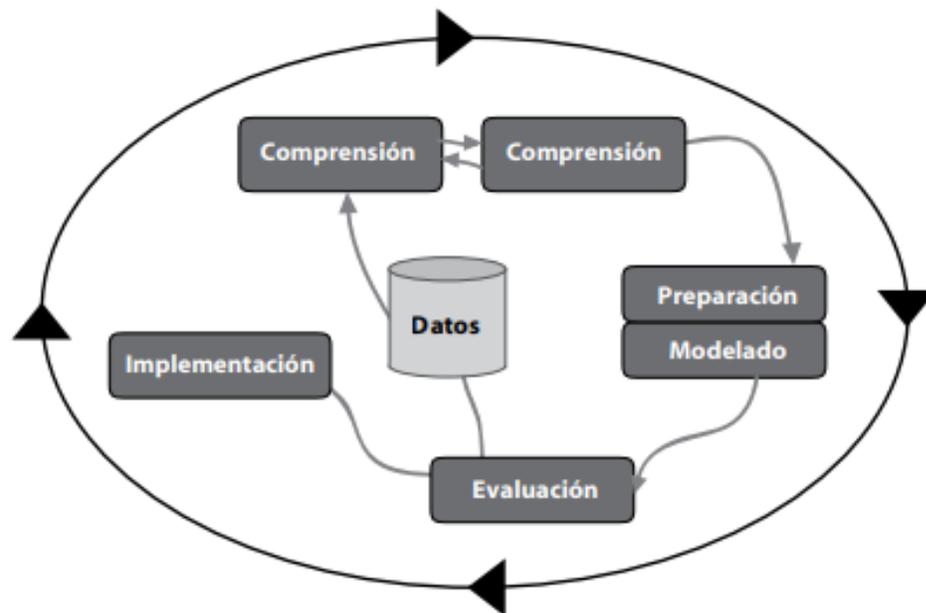
Fuente: Chapman, Clinton, & Kerber, 2000

### 2.9.2. Fases de la metodología

**Fase 1. Comprensión del negocio o problema.** Comprende los requisitos y objetivos del proyecto desde una perspectiva empresarial o institucional para convertirlos en objetivos técnicos y en un plan de proyecto, para lo cual es necesario comprender de manera completa el problema por resolver (Chapman, Clinton, & Kerber, 2000) (Véase en la figura 2.7).

- **Determinar los objetivos.** Se determina cuál es el problema que se quiere resolver y por qué se usa minería de datos para dicho propósito; también se deben fijar los criterios de éxito. En cuanto a estos últimos, pueden ser de tipo cualitativo o de tipo cuantitativo; por ejemplo, si el problema es detectar fraude en el uso de tarjetas de crédito, el criterio de éxito cuantitativo sería el número de detecciones de fraude.

- **Evaluar la situación actual.** En esta tarea se evalúan antecedentes y requisitos del problema, tanto en términos del negocio como en términos de la minería de datos. Algunos de los aspectos por tener en cuenta pueden ser el conocimiento previo acerca del tema, la cantidad de datos requeridos para resolver el problema, ventajas de aplicar minería de datos al problema, entre otros.



*Figura 2. 7 Ciclo de vida Crisp-Dm*

*Fuente: Chapman, Clinton, & Kerber, 2000*

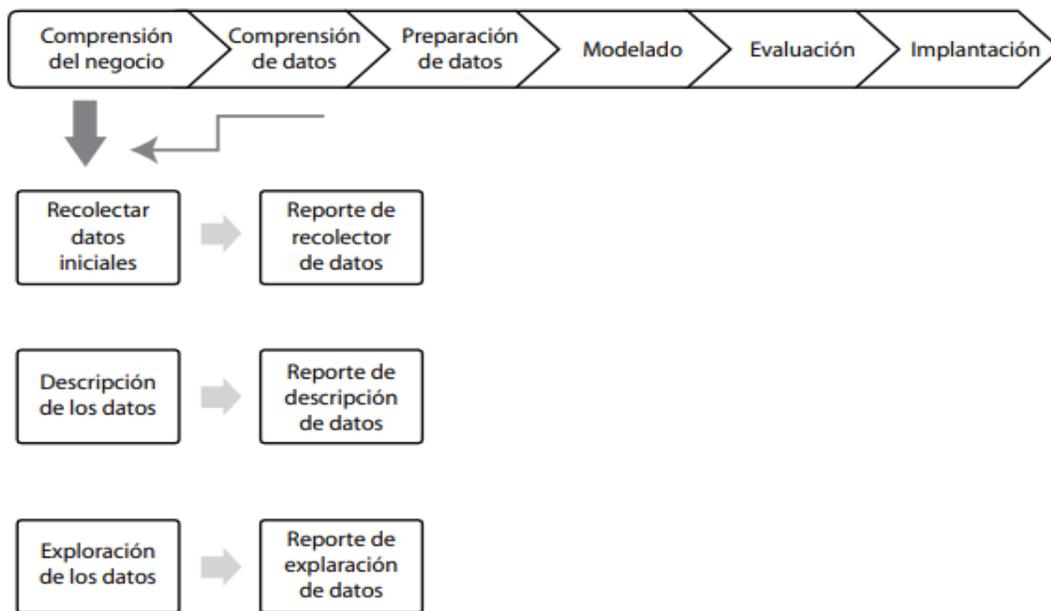
- **Determinar los objetivos de la minería de datos.** El objetivo de esta tarea es representar los objetivos del negocio en términos de las metas del proyecto de minería de datos. Por ejemplo, si el objetivo del negocio es el desarrollo de una campaña publicitaria para incrementar asignación de créditos hipotecarios, la meta de la minería de datos sería determinar el perfil de los clientes respecto a su capacidad de endeudamiento (véase en la figura 2.8).



*Figura 2. 8 Fase de comprensión del negocio*

*Fuente: Chapman, Clinton, & Kerber, 2000*

**Fase 2. Comprensión de los datos.** Corresponde a la recolección inicial de los datos para establecer un primer contacto con el problema; esta fase, junto con la fase 3 y la fase 4, demanda mayor esfuerzo y tiempo (véase figura 2.12). Las principales tareas que se deben desarrollar en la fase de comprensión de los datos son: recolectar datos iniciales, describir los datos, explorar los datos y verificar la calidad de los datos (Chapman, Clinton, & Kerber, 2000) (véase en la figura 2.9).



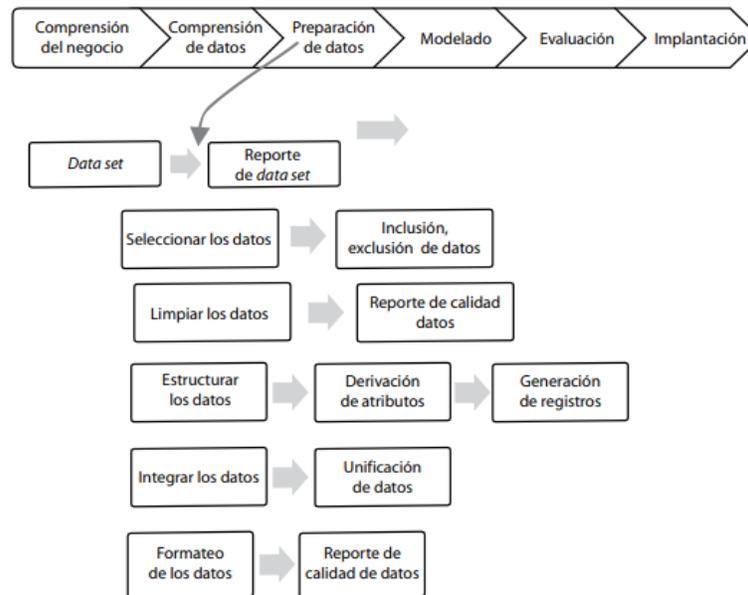
*Figura 2. 9 Fase de comprensión de los datos*

*Fuente: Chapman, Clinton, & Kerber, 2000*

- **Recolectar datos iniciales.** Tiene como objetivo principal la recolección de datos iniciales y su adecuación para su posterior procesamiento. Se deben elaborar informes con una lista de los datos adquiridos, su localización, las técnicas utilizadas en su recolección y los problemas y soluciones inherentes a este proceso.
- **Describir los datos.** Se deben describir los datos iniciales obtenidos, tales como número de registros y campos por registro, su identificación, el significado de cada campo y la descripción del formato inicial.
- **Explorar los datos.** Su finalidad es descubrir una estructura general para los datos. Involucra la aplicación de pruebas estadísticas básicas, que revelen propiedades en los

datos, se crean tablas de frecuencia y se construyen gráficos de distribución. Se crea un informe de exploración de datos.

**Fase 3. Preparación de los datos.** Se usa para adaptarlos a la técnica de minería de datos, mediante la visualización de los datos y la búsqueda de relaciones entre las variables. Esta fase es la de modelado, ya que los datos requieren ser procesados de diferentes formas; por ende, las fases de preparación y modelado interactúan permanentemente (Chapman, Clinton, & Kerber, 2000) (véase figura 2.10).



*Figura 2. 10 Fase de Preparación de los datos*

*Fuente: Chapman, Clinton, & Kerber, 2000*

Los pasos que se consideran para la preparación de los datos son: seleccionar, limpiar, estructurar, integrar y formatear los datos.

- **Seleccionar los datos.** Se selecciona un subconjunto de datos considerando la calidad de los datos, la limitación en el volumen o en los tipos de datos que están relacionados con las técnicas de minería de datos seleccionados.
- **Limpiar los datos.** Existe una diversidad de técnicas aplicables a esta tarea con el fin de optimizar la calidad de los datos en la perspectiva de prepararlos para la fase de modelación. Algunas de las técnicas son: normalización de los datos, discretización de campos numéricos, tratamiento con valores vacíos, reducción del volumen de datos.
- **Estructurar los datos.** Algunas de las operaciones por realizar en esta tarea son la generación de nuevos atributos a partir de atributos ya existentes, integración de nuevos registros o transformación de valores para atributos existentes.

**Fase 4. Modelado.** Corresponde a la selección de un modelo adecuado y específico; para ello se usan técnicas que cumplan los siguientes criterios (Chapman, Clinton, & Kerber, 2000) (véase figura 2.11).

- Ser apropiada para el problema.
- Disponer de datos adecuados.
- Cumplir con los requisitos del problema.
- Técnica adecuada para obtener un modelo.
- **Conocimiento pleno de la técnica.** Por ejemplo, si el problema es de clasificación, podemos elegir entre árboles de decisión, knearest neighbour o razonamiento basado en casos (cbr).

- **Generar plan de prueba.** Se debe generar un plan para probar la calidad y validez del modelo construido; por ejemplo, en una tarea como la clasificación es posible usar la razón de error como medida de la calidad. Entonces, típicamente se separan los datos en dos conjuntos, uno de entrenamiento y otro de prueba.
- **Evaluar el modelo.** Se deben interpretar los modelos de acuerdo con el conocimiento del dominio y los criterios de éxitos preestablecidos.

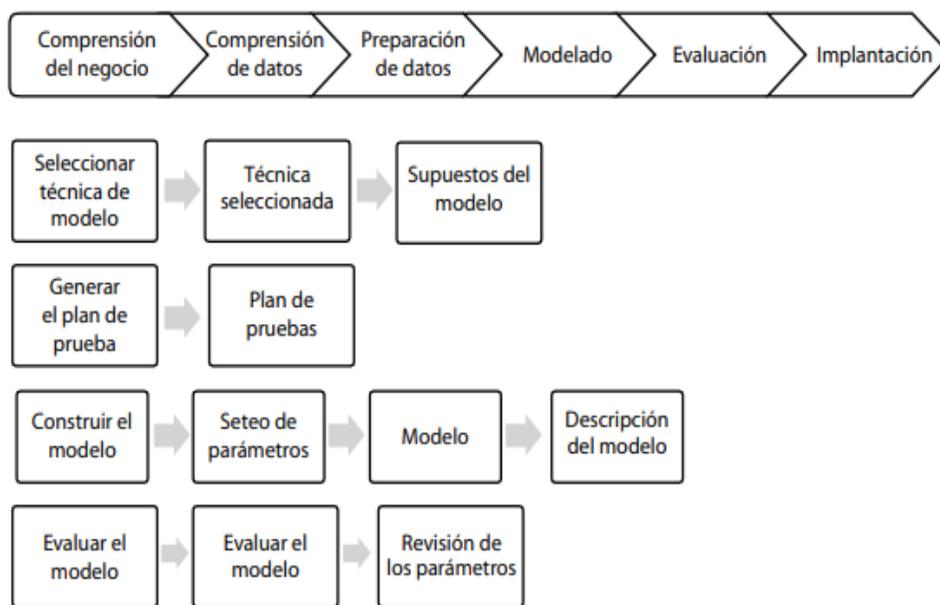


Figura 2. 11 Fase de modelado

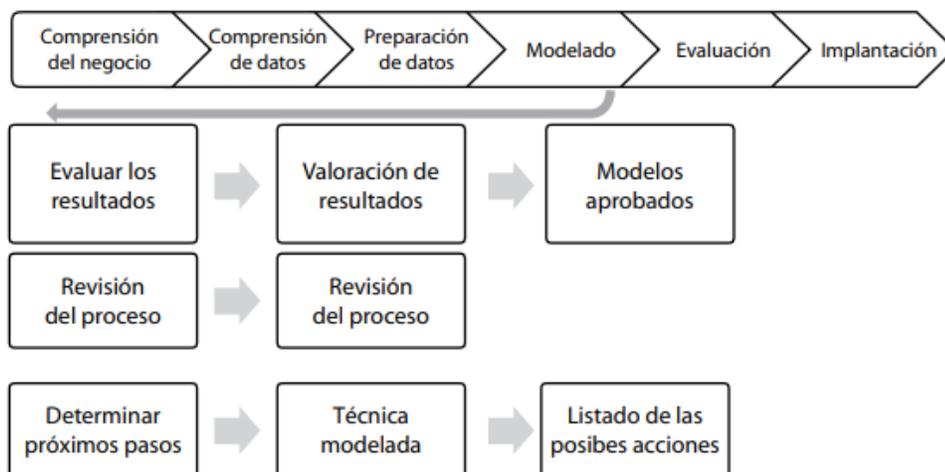
Fuente: Chapman, Clinton, & Kerber, 2000

**Fase 5. Evaluación.** Evalúa el modelo teniendo en cuenta el cumplimiento de los criterios de éxito del problema; para ello se emplean múltiples herramientas para la interpretación de los resultados, entre ellas matrices de confusión Edelstein 1999, que es una tabla que indica cuántas

clasificaciones se han hecho para cada tipo (Chapman, Clinton, & Kerber, 2000) (véase en la figura 2.12).

La diagonal de la tabla representa las clasificaciones correctas. Si es válido lo anterior, se procede a la explotación del modelo, que es el mantenimiento de la aplicación y la posible difusión de los resultados.

Una vez que el modelo ha sido construido y validado, se transforma el conocimiento obtenido en acciones dentro del proceso de negocio; la retroalimentación generada por la monitorización y mantenimiento puede indicar si el modelo está siendo utilizado apropiadamente.



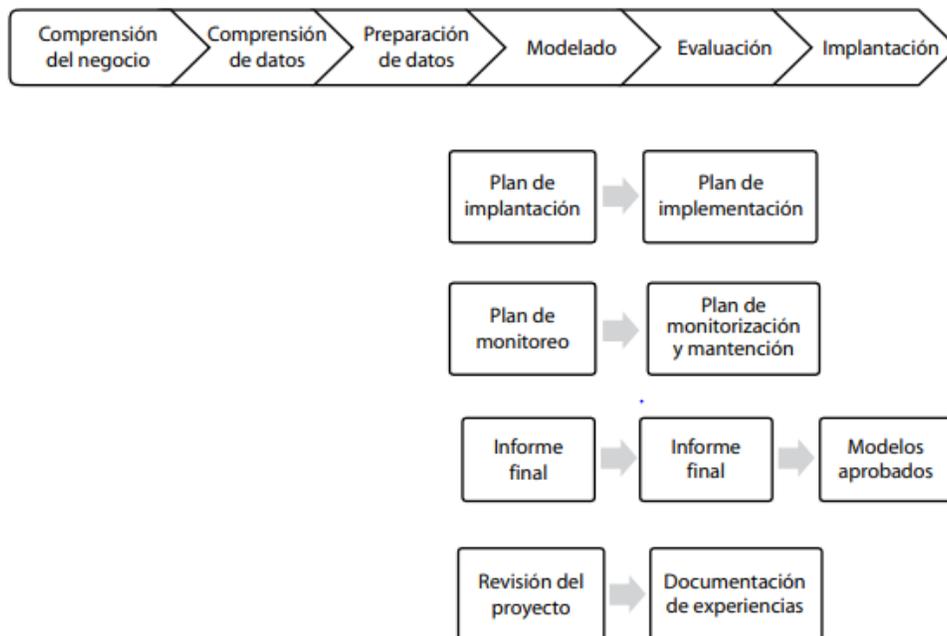
*Figura 2. 12 Fase de evaluación*

*Fuente: Chapman, Clinton, & Kerber, 2000*

**Fase 6. Implementación.** Es aquí donde el conocimiento obtenido se transforma en acciones dentro del proceso de negocio, ya sea observando el modelo y resultados, o aplicándolo a múltiples grupos de datos o como parte del proceso. Las tareas que se efectúan son: planear la

implementación, monitorizar y mantener, informe final y revisar el proyecto (Chapman, Clinton, & Kerber, 2000) (véase en la figura 2.13).

- **Planear la implementación.** Esta tarea toma los resultados de la evaluación y concluye una estrategia para su implementación. Si un procedimiento general se ha identificado para crear el modelo, debe estar documentado para su posterior implementación.
- **Monitorizar y mantener.** Se deben preparar estrategias de monitorización y mantenimiento para ser aplicadas sobre los modelos.
- **Informe final.** Dependiendo del plan de implementación, este puede ser un resumen de los puntos importantes del proyecto y la experiencia lograda, o puede ser una presentación final que incluya y explique los resultados logrados con el proyecto.
- **Revisar el proyecto.** Se evalúa lo correcto y lo incorrecto.



*Figura 2. 13 Fase de Implantación*

*Fuente: Chapman, Clinton, & Kerber, 2000*

Dependiendo del plan de implementación, este informe puede ser sólo un resumen de los puntos importantes del proyecto y la experiencia lograda o puede ser una presentación final que incluya y explique los resultados logrados con el proyecto. Revisión del proyecto: En este punto se evalúa qué fue lo correcto y qué lo incorrecto, qué es lo que se hizo bien y qué es lo que se requiere mejorar

## **2.10. INGENIERÍA DE SOFTWARE**

(Pessman, 2006) aporta con la siguiente conceptualización, en la actualidad la enorme industria del software se ha convertido en un factor dominante en las economías del mundo industrializado. Equipos de especialistas de software, cada uno centrado en una parte de la tecnología que se requiere para llegar a una aplicación compleja, han reemplazado al programador solitario de los primeros tiempos.

El autor (Cortes Morales, 2006) define que primero debemos entenderla en el sentido de que el software debe cumplir con los requisitos exigidos dentro del sistema. Si no cumple con ese cometido, simplemente el producto no servirá. En este caso usted deberá aplicar criterios como: validez, correctitud, verificación, confiabilidad, etc.

(Zavala Ruiz, 2003) menciona que el Ingeniería de Software es la rama de la ingeniería que la ciencia de la computación y las matemáticas para lograr soluciones efectivas eficaces en costo o económico a los problemas relacionados con el desarrollo del software.

Por lo tanto, en la actualidad el uso de la ingeniería del software como disciplina empleando para el desarrollo del sistema de información, para lograr un producto eficaz y económico.

### **2.10.1. Ingeniería de Software Orientado a Objetos**

La orientación a objetos es un buen ejemplo de cómo los pequeños detalles tecnológicos pueden resultar tan significante. Estos detalles se aprecian mejor si se contrastan con la programación tradicional. La programación orientada a objetos define una estructura de más alto nivel llamada objeto, que ofrece dos ventajas sobre la programación tradicional (Weitzenfeid, 2002).

- La primera es permitir al programador que organice su programa de acuerdo con abstracciones de más alto nivel, siendo esta más cercanas a la manera de pensar de la gente. En otras palabras, los objetos son las unidades de representación de la aplicación, por ejemplo, cuentas de banco, reservaciones de vuelos.

- La segunda es que los datos globales desaparecen, siendo estos junto con las funciones parte interna de los objetos. Por lo tanto, cualquier cambio en la estructura de alguno de los datos solo debería afectar las funciones definidas en ese mismo objeto y no los demás.

El análisis orientado a objetos es el proceso que modela el dominio del problema mediante la identificación y la especificación de un conjunto de objetos semánticos que interaccionan y se comportan de acuerdo a los requisitos del sistema (Monarchi & Puhr, 1992).

### 2.10.2. Proceso del Software

Según (Alava, 2015) el desarrollo de un software es un proceso que no solo permite cumplir el objetivo de desarrollarlo, sino que también es un proceso de aprendizaje reiterativo, ya que permite documentar la información además de construir un software de calidad. (Véase en la figura 2.14)

El proceso del software tiene las siguientes características:

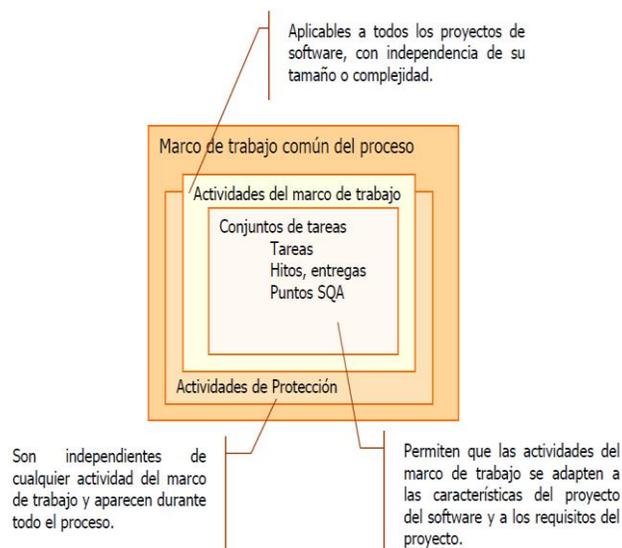


Figura 2. 14 Características del proceso general de software

Fuente: Alava 2015

- **Flujo de proceso**

En el flujo de proceso se describe el orden de cómo se van a ejecutar las actividades y también se describe el tiempo que va durar cada actividad. Hay tres tipos de flujo de proceso

- **Flujo de proceso lineal**

El flujo de proceso lineal se realiza en una secuencia que empieza desde la comunicación hasta y termina en el despliegue (Véase en la figura 2.15).

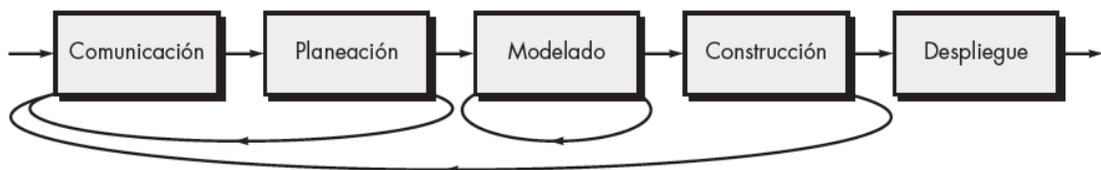


*Figura 2. 15 Flujo de proceso lineal*

*Fuente: Alava 2015*

- **Flujo de proceso iterativo**

Dentro del flujo de proceso iterativo se repiten las actividades una y otra vez mientras sea necesario para avanzar con la siguiente actividad (Véase en la figura 2.16).

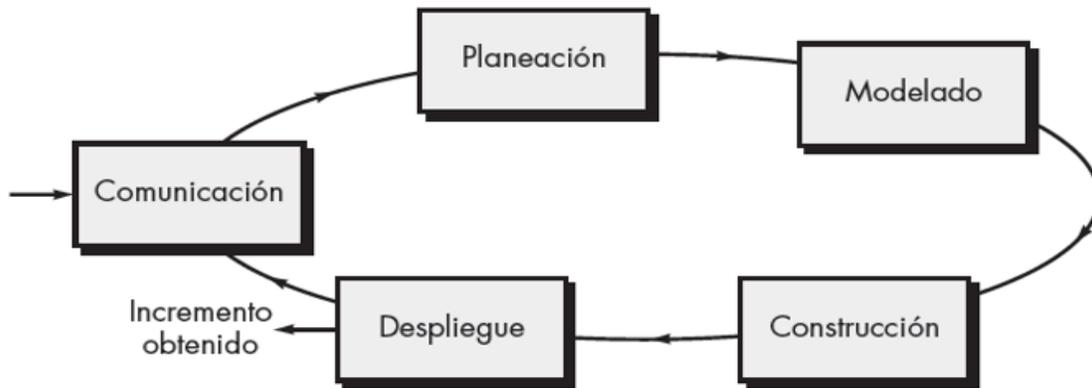


*Figura 2. 16 Flujo de proceso Iterativo*

*Fuente: Alava 2015*

- **Flujo de proceso evolutivo**

Las actividades son ejecutadas en forma circular y en cada círculo que realice es una versión mejorada del producto (Véase en la figura 2.17).

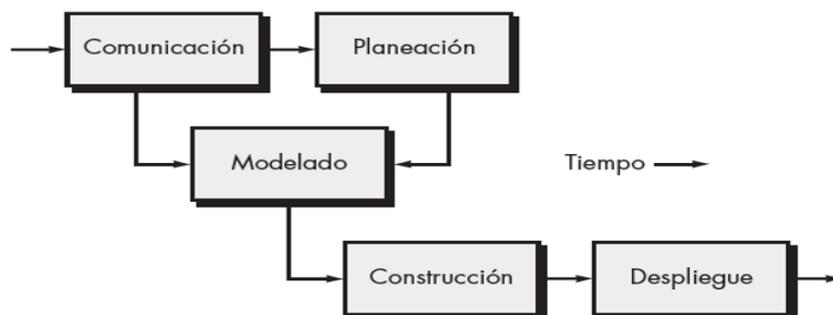


*Figura 2. 17 Flujo de proceso evolutivo*

*Fuente: Alava 2015*

- **Flujo de proceso en paralelo**

Ejecuta una o dos actividades en paralelo, es decir al mismo tiempo que se ejecuta otra actividad (Véase en la figura 2.18).



*Figura 2. 18 Flujo de proceso paralelo*

*Fuente: Alava 2015*

### 2.10.2.1. El ciclo de desarrollo de la arquitectura

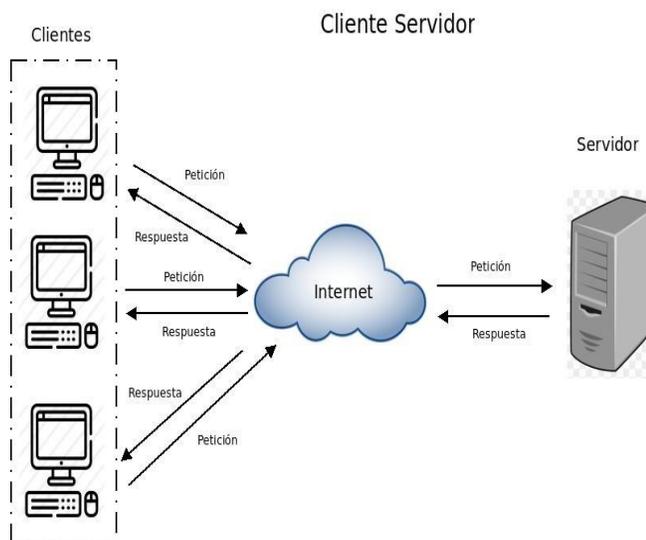
Dentro de un proyecto de desarrollo, e independientemente de la metodología que se utilice, se puede hablar de “desarrollo de la arquitectura de software”. Este desarrollo, que precede a la construcción del sistema, está dividido en las siguientes etapas: requerimientos, diseño, documentación y evaluación.

A continuación, se describen dichas etapas.

- **Requerimientos.** La etapa de requerimientos se enfoca en la captura, documentación y priorización de requerimientos que influyen la arquitectura. Como se mencionó anteriormente, los atributos de calidad juegan un papel preponderante dentro de estos requerimientos, así que esta etapa hace énfasis en ellos.
- **Diseño.** La etapa de diseño es la etapa central en relación con la arquitectura y probablemente la más compleja. Durante esta etapa se definen las estructuras que componen la arquitectura. La creación de estas estructuras se hace en base a patrones de diseño, tácticas de diseño y elecciones tecnológicas.
- **Documentación.** Una vez creado el diseño de la arquitectura, es necesario poder comunicarlo a otros involucrados dentro del desarrollo. La comunicación exitosa del diseño muchas veces depende de que dicho diseño sea documentado de forma apropiada.
- **Evaluación.** Dado que la arquitectura de software juega un papel crítico en el desarrollo, es conveniente evaluar el diseño una vez que este ha sido documentado con el fin de identificar posibles problemas y riesgos.

### 2.10.3. Arquitectura Cliente – Servidor

La estructura cliente - servidor es una arquitectura de computación en la que se consigue un procesamiento cooperativo de la información por medio de un conjunto de procesadores, de tal forma que uno o varios clientes, distribuidos geográficamente o no, solicitan servicios de computación a uno o más servidores (véase en la figura 2.19).



*Figura 2. 19 Cliente-Servidor*

*Fuente: Gonzales Reyes2017*

#### 2.10.3.1. El Esquema de funcionamiento de un Sistema Cliente/Servidor:

- El cliente solicita una información al servidor.
- El servidor recibe la petición del cliente.
- El servidor procesa dicha solicitud.
- El servidor envía el resultado obtenido al cliente.
- El cliente recibe el resultado y lo procesa.

### 2.10.3.2. Componentes de la Arquitectura Cliente – Servidor

El modelo Cliente/Servidor es un modelo basado en la idea del servicio, en el que el cliente es un proceso consumidor de servicios y el servidor es un proceso proveedor de servicios. Además, esta relación está establecida en función del intercambio de mensajes que es el único elemento de acoplamiento entre ambos

- **Nivel de Presentación:** Agrupa a todos los elementos asociados al componente Cliente.
- **Nivel de Aplicación:** Agrupa a todos los elementos asociados al componente Servidor.
- **Nivel de comunicación:** Agrupa a todos los elementos que hacen posible la comunicación entre los componentes Cliente y servidor.
- **Nivel de base de datos:** Agrupa a todas las actividades asociadas al acceso de los datos.

### 2.10.3.3. Elementos principales

#### 2.10.3.3.1. Cliente

Un cliente es todo proceso que reclama servicios de otro. Una definición un poco más elaborada podría ser la siguiente: cliente es el proceso que permite al usuario formular los requerimientos y pasarlos al servidor. Se lo conoce con el término front-end. Las funciones que lleva a cabo el proceso cliente se resumen en los siguientes puntos:

- Administrar la interfaz de usuario.
- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.
- Generar requerimientos de bases de datos.
- Recibir resultados del servidor.

- Formatear resultados.

De este modo el cliente se puede clasificar en:

- **Ciente basado en aplicación de usuario.** Si los datos son de baja interacción y están fuertemente relacionados con la actividad de los usuarios de esos clientes.
- **Ciente basado en lógica de negocio.** Toma datos suministrados por el usuario y/o la base de datos y efectúa los cálculos necesarios según los requerimientos del usuario.

#### 2.10.3.3.2. Servidor

Un servidor es todo proceso que proporciona un servicio a otros. Es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Al proceso servidor se lo conoce con el término back-end. El servidor normalmente maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos. Las principales funciones que lleva a cabo el proceso servidor se enumeran a continuación:

- Aceptar los requerimientos de bases de datos que hacen los clientes.
- Procesar requerimientos de bases de datos.
- Formatear datos para transmitirlos a los clientes.
- Procesar la lógica de la aplicación y realizar validaciones a nivel de bases de datos.

#### 2.10.3.4. Tipos de Arquitectura Cliente – Servidor

- **Servidores de Ficheros** Con un servidor de archivos, un cliente lo que hace es requerimientos de los mismos sobre una red. Esta es una forma muy primitiva de

servicios de datos, la cual necesita intercambio de muchos mensajes sobre una red para hallar el dato requerido.

- **Servidores de Bases de Datos** Este análisis está elaborado desde el punto de vista del modelo Cliente/Servidor, y está directamente relacionado con la arquitectura en dos planos, que se describirá en el apartado siguiente. Obviamente la creación de aplicaciones Cliente/Servidor está asociada a la utilización de servidores de bases de datos relacionales SQL, y dependiendo de los requerimientos y restricciones se debe elegir entre una arquitectura dos o tres planos.
- **Servidores de Transacciones** Estos tipos de sistemas se pueden implementar con cualquiera de las modalidades Cliente/Servidor en dos o tres planos, pero incorporan un elemento principal sobre el cual se elabora y basa toda la fortaleza de este modelo, el concepto de transacción.
- **Servidores de Objetos** Con un servidor de objetos, las aplicaciones Cliente/Servidor son escritas como un conjunto de objetos que se comunican. Los objetos cliente se comunican con los objetos servidores usando un Object Request Broker (ORB).

## 2.11. HERRAMIENTAS DE LA APLICACIÓN

### 2.11.1. PHP

Según (Baquero, Garcia, 2017) nos comenta que PHP corresponde con las iniciales de Personal Home Page Tools. PHP es software libre, nacido en 1994 de la mano de Rasmus Lerdof, que ha ido creciendo gracias a las aportaciones de los miembros de la gran comunidad PHP, que hoy en día cuenta con un potente núcleo de lenguaje y con muchísimas librerías.

Por otra parte (Cobo, Gómez, Pérez, & Rocha, 2005) conceptualiza que es un lenguaje interpretado del lado del servidor que surge dentro de la corriente denominada código abierto (open source). Se caracteriza por su potencia, versatilidad, robustez y modularidad. Al igual que ocurre con tecnologías similares, los programas son integrados directamente dentro del código HTML (véase en la figura 2.20).

Por lo tanto, en definitiva, PHP es uno de los lenguajes más utilizados actualmente en el desarrollo de aplicaciones web y viene experimentando un constante crecimiento en su nivel de utilización en Internet.

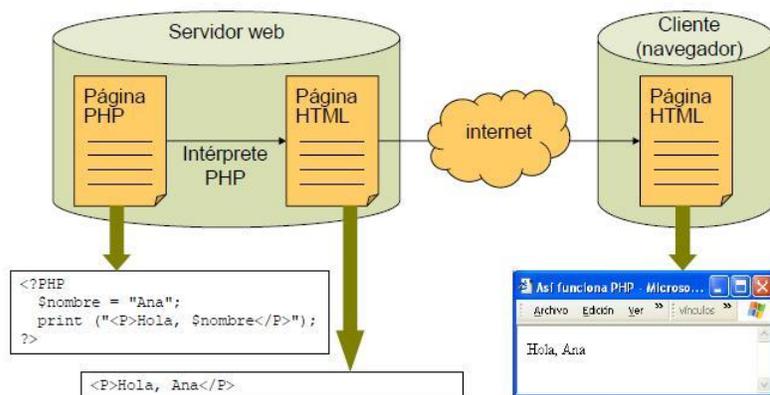


Figura 2. 20 Así funciona PHP

Fuente: Rodríguez Pérez, 2018

- PHP es un lenguaje de script del lado del servidor. Otros lenguajes similares son ASP, JSP o ColdFusion.
- Los scripts PHP están incrustados en los documentos HTML y el servidor los interpreta y ejecuta antes de servir las páginas al cliente.
- El cliente no ve el código PHP sino los resultados que produce.

### **2.11.2. JavaScript**

El lenguaje interpretado orientado a objetos desarrollado por Netscape que se utiliza en millones de páginas web y aplicaciones de servidor en todo el mundo. JavaScript de Netscape es un superconjunto del lenguaje de scripts estándar de la edición de ECMA-262 3 (ECMAScript) que presenta sólo leves diferencias respecto a la norma publicada.

Contrariamente a la falsa idea popular, JavaScript no es "Java interpretativo". En pocas palabras, JavaScript es un lenguaje de programación dinámico que soporta construcción de objetos basado en prototipos. La sintaxis básica es similar a Java y C++ con la intención de reducir el número de nuevos conceptos necesarios para aprender el lenguaje. Las construcciones del lenguaje, tales como sentencias if, y bucles for y while, y bloques switch y try ... catch funcionan de la misma manera que en estos lenguajes (o casi).

JavaScript puede funcionar como lenguaje procedimental y como lenguaje orientado a objetos. Los objetos se crean programáticamente añadiendo métodos y propiedades a lo que de otra forma serían objetos vacíos en tiempo de ejecución, en contraposición a las definiciones sintácticas de

clases comunes en los lenguajes compilados como C++ y Java. Una vez se ha construido un objeto, puede usarse como modelo (o prototipo) para crear objetos similares.

Las capacidades dinámicas de JavaScript incluyen construcción de objetos en tiempo de ejecución, listas variables de parámetros, variables que pueden contener funciones, creación de scripts dinámicos (mediante eval), introspección de objetos (mediante for ... in), y recuperación de código fuente (los programas de JavaScript pueden decompilar el cuerpo de funciones a su código fuente original) (Mozilla, 2020).

### **2.11.3. Bootstrap**

Bootstrap es un kit de herramientas de código abierto para desarrollar con HTML, CSS y JS. Realice rápidamente prototipos de sus ideas o cree toda su aplicación con nuestras variables y mixins Sass, sistema de cuadrícula sensible, componentes precompilados extensos y complementos potentes creados en jQuery (MIT, 2020).

Bootstrap facilita la maquetación de sitios web, además de ser compatible con preprocesadores como Less y Saas, nos ofrece las herramientas para que nuestro sitio web se vea bien en toda clase de dispositivos, ahorrándonos así el trabajo de tener que rediseñar un sitio web (MIT, 2020).

### **2.11.4. Ajax**

AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible

realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página, aunque existe la posibilidad de configurar las peticiones como síncronas de tal forma que la interactividad de la página se detiene hasta la espera de la respuesta por parte del servidor.

JavaScript es un lenguaje de programación (scripting language) en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dado que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

Ajax es una combinación de cuatro tecnologías ya existentes:

- XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño que acompaña a la información.
- Document Object Model (DOM) accedido con un lenguaje de scripting por parte del usuario, especialmente implementaciones ECMAScript como JavaScript y Script, para mostrar e interactuar dinámicamente con la información presentada.

- El objeto XMLHttpRequest para intercambiar datos de forma asíncrona con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto iframe en lugar del XMLHttpRequest para realizar dichos intercambios. PHP es un lenguaje de programación de uso general de script del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico también utilizado en el método Ajax.
- XML es el formato usado generalmente para la transferencia de datos solicitados al servidor, aunque cualquier formato puede funcionar, incluyendo HTML preformateado, texto plano, JSON y hasta EBML

### 2.11.5. jQuery

El propósito de jQuery es hacer que sea mucho más fácil usar JavaScript en su sitio web. jQuery toma muchas tareas comunes que requieren muchas líneas de código JavaScript para lograrlo, y las envuelve en métodos que puede llamar con una sola línea de código. jQuery también simplifica muchas de las cosas complicadas de JavaScript, como las llamadas AJAX y la manipulación DOM (W3Schools, 2020).

La biblioteca jQuery contiene las siguientes características:

- Manipulación HTML / DOM
- Manipulación de CSS
- Métodos de eventos HTML
- Efectos y animaciones.
- AJAX
- Utilidades

JQuery es una librería de JavaScript (JavaScript es un lenguaje de programación muy usado en desarrollo web). Esta librería de código abierto, simplifica la tarea de programar en JavaScript y permite agregar interactividad a un sitio web sin tener conocimientos del lenguaje.

Basados en esta librería, existe una infinita cantidad de plugins (gratis y pagos) creados por desarrolladores de todo el mundo. Estos plugins resuelven situaciones concretas dentro del maquetado de un sitio, por ejemplo: un menú responsive, una galería de fotos, un carrousel de imágenes, un slide, un header que cambia de tamaño, el deslizamiento del scroll al hacer clic en un botón (anclas HTML), la transición entre páginas y miles de efectos más (CHUBURU, 2018).

#### **2.11.6. HTML**

HTML es el acrónimo en inglés de HyperText Markup Language (en español se traduce como lenguaje de marcado de hipertexto). Es un lenguaje abstracto que usan las aplicaciones para representar documentos (se les llama documentos a instancias completas, como lo son las páginas web) y que puede ser transmitido fácilmente por algún medio, como Internet. Los navegadores de Internet procesan e interpretan documentos descritos en HTML usando un analizador de HTML (nextu, 2020).

El lenguaje HTML está definido por lo que se llama etiquetas, cuyo nombre se delimita usando los símbolos < y >, de la siguiente forma: <etiqueta>. Dichas etiquetas se utilizan para describir algo que se quiere representar en una página web.

Por ejemplo: <title>Internet básico, email, descargas y compras en línea</title>

El HTML como buen lenguaje de marcado, se basa en un sistema de etiquetas. Estas etiquetas se usan para que el navegador comprenda que clase de información le estamos dando. Por ejemplo, tenemos etiquetas para indicar títulos, toda la información que metamos en esas etiquetas, el navegador sabrá que son títulos.

HTML es el acrónimo en inglés de HyperText Markup Language (en español se traduce como lenguaje de marcado de hipertexto). Es un lenguaje abstracto que usan las aplicaciones para representar documentos (se les llama documentos a instancias completas, como lo son las páginas web) y que puede ser transmitido fácilmente por algún medio, como Internet. Los navegadores de Internet procesan e interpretan documentos descritos en HTML usando un analizador de HTML.

El lenguaje HTML está definido por lo que se llama etiquetas, cuyo nombre se delimita usando los símbolos < y >, de la siguiente forma: <etiqueta>. Dichas etiquetas se utilizan para describir algo que se quiere representar en una página web.

#### **2.11.7. CodeIgniter**

Según (EllisLab 2011) organización responsable del desarrollo de CodeIgniter, es un entorno de desarrollo rápido, con un grupo específico de herramientas para programadores que generan aplicaciones en lenguaje.

(British Columbia Institute of Technology, 2017) conceptualiza que codeigniter es un marco de desarrollo de aplicaciones - un conjunto de herramientas - para las personas que construyen sitios web utilizando PHP.

Por lo tanto, codeigniter es un framework para aplicaciones web de código abierto; muy útil para crear sitios web dinámicos con PHP. Su objetivo es permitir que los desarrolladores puedan realizar proyectos mucho más rápido que creando toda la estructura desde cero, brindando un conjunto de bibliotecas para tareas comunes, así como una interfaz simple y una estructura lógica para acceder esas bibliotecas.

### 2.11.7.1. Aplicación de CodeIgniter

El siguiente gráfico ilustra como es el flujo de datos en el framework CodeIgniter:

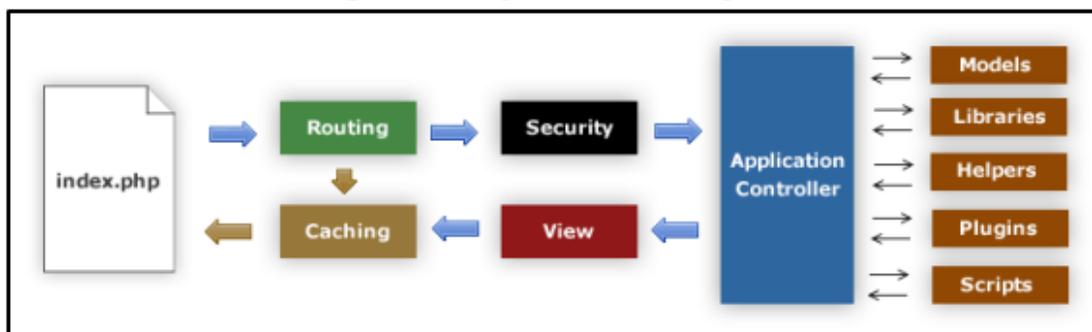


Figura 2. 21 Flujo de datos Codeigniter

Fuente: EllisLab 2011

- El index.php sirve como el controlador frontal, la inicialización de los recursos básicos necesarios para ejecutar CodeIgniter.
- El router examina la solicitud HTTP para determinar qué se debe hacer con él.

- Si existe un archivo de caché, se envía directamente al navegador, sin pasar por la ejecución normal del sistema.
- Seguridad. Antes de que se cargue el controlador de la aplicación, la solicitud HTTP y los datos enviados por los usuarios se filtra para la seguridad.
- El controlador de carga el modelo, las bibliotecas centrales, ayudantes, y cualesquiera otros recursos que se necesitan para procesar la solicitud específica.

La vista finalizada se representa a continuación, envía al navegador web para ser visto. Si el almacenamiento en caché está habilitado, la vista se almacena en caché primero para que en las solicitudes posteriores se puede servir.

#### **2.11.7.2. Definición de MVC según CodeIgniter**

La guía del Framework de CodeIgniter MVC indica que el “framework CodeIgniter se basa en el patrón de desarrollo Model-View-Controller. MVC es un enfoque de software que separa la lógica de aplicación de la presentación. En la práctica, permite que las 30 páginas web que contienen secuencias de comandos mínima ya que la presentación está separada del scripting PHP.

- El modelo representa sus estructuras de datos. Normalmente las clases del modelo contendrán las funciones que le ayudan a recuperar, insertar, y actualizar la información en su base de datos.
- La vista es la información que se presenta a un usuario. Una visión será normalmente una página web, pero en CodeIgniter, una opinión también puede ser un fragmento de la página como un encabezado o pie de página. También puede ser una página de RSS, o cualquier otro tipo de "página".

- El controlador actúa como un intermediario entre el Modelo, la Vista, y todos los demás recursos necesarios para procesar la petición HTTP y generar una página web.

### 2.11.8. PostgreSQL

Según (Camps Paré, Rafael 2005) PostgreSQL proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como DB2 u Oracle. La siguiente es una breve lista de algunas de esas características, a partir de PostgreSQL 7.1.x.

- “Objeto-Relacional: aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi versión, soporte multiusuario, transacciones, optimización de consultas, herencia, y arreglos.
- Altamente extensible: soporta operadores, métodos de acceso y tipos de datos definidos por el usuario.
- Soporte SQL: soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (joins) de SQL92.
- Integridad referencial: soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- API Flexible: la flexibilidad del API de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para el RDBMS PostgreSQL. Estas

interfaces que incluye pueden ser escritas en el lenguaje Object Pascal, Python, Perl, PHP, ODBC, Java/JDBC, Ruby, TCL, C/C++, y Pike.

- Lenguajes procedurales: tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.
- Cliente/Servidor: usa una arquitectura proceso-por-usuario cliente/servidor. Esta es similar al método del Apache 1.3.x para manejar procesos. Hay un proceso maestro que se ramifica para proporcionar conexiones adicionales para cada cliente que intente conectar a PostgreSQL. (González, Carlos D. 2011).
- Restricciones y disparadores: tienen la función de mantener la integridad y consistencia en la BD. Consisten en la ejecución de acciones antes o después de un evento de BD.
- Múltiples tipos de datos predefinidos: como todos los manejadores de bases de datos, PostgreSQL implementa los tipos de datos definidos para el estándar SQL3 y aumenta algunos otros.
- Conectividad TCP/IP, JDBC y ODBC.
- Interfaz con diversos lenguajes: C, C++, Java, Delphi, Python, Perl, PHP, Bash, etc.

**2.11.8.1. Ventajas de PostgreSQL**

El sistema de gestión de bases de datos PostgreSQL ofrece muchas ventajas respecto a otros sistemas de bases de datos, algunas de ellas son:

- Instalación ilimitada, ya que no hay costo asociado a la licencia del software.
- Esto tiene varias ventajas adicionales:
- Modelos de negocios más rentables con instalaciones a gran escala.
- No existe la posibilidad de ser auditado para verificar cumplimiento de licencia en ningún momento.
- Flexibilidad para hacer investigación y desarrollo sin necesidad de incurrir en costos adicionales de licenciamiento
- Mejor soporte, pues cuenta con una importante comunidad de profesionales.
- Ahorros considerables en costos de operación ya que el software ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que los productos de los proveedores comerciales, conservando todas las características, estabilidad y rendimiento.
- Estabilidad y confiabilidad, puesto que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad.
- Extensible ya que el código fuente está disponible para todos sin costo.
- Multiplataforma pues PostgreSQL está disponible en casi cualquier Unix y para Windows.

- Diseñado para ambientes de alto volumen ya que usa una estrategia de almacenamiento de filas llamada MVCC para conseguir una mejor respuesta en ambientes de grandes volúmenes.
- Herramientas gráficas para administrar las bases de datos (pgAdmin, pgAccess) y para hacer diseño de bases de datos (Tora, Data Architect).

### **2.11.9. Servidor**

Según (Fumás, 2014) apache es un poderoso servidor web de código abierto, un programa diseñado para la transferencia de datos de hipertexto, utiliza el protocolo de transferencia de hipertexto denotado como HTTP el cual se encarga de la transferencia de datos en la World Wide Web.

Apache presenta ventajas como ser un software de código abierto, costo gratuito, alta aceptación en la red y popularidad para la contribución de su evolución, multiplataforma, capacidad de manejar más de un millón de visitas. Así como presenta ventajas, presenta también inconvenientes como la falta de integración, formatos de configuración no estándar y carece de un panel de configuración.

### **2.11.10. Minería de Datos**

RapidMiner es un entorno para el aprendizaje de máquinas y procesos de minería de datos. A través de un concepto modular, permite el diseño de modelos de aprendizaje empleando operadores de cadena para diversos problemas. El manejo de los datos es transparente para los diferentes operadores implementados, ya que no tienen que manejar los datos reales directamente, sino hace uso de transformaciones para su manejo.

Hoy en día, RapidMiner es el líder en el mundo de código abierto para procesos de minería de datos y aprendizaje de máquinas, y es ampliamente utilizado por los investigadores y empresas.

Entre las características más sobresalientes tenemos:

- El proceso de KDD es modelado como simples operadores de árbol.
- Tiene una representación interna basada en archivos XML.
- Tiene una interfaz gráfica de usuario (GUI) para el diseño de prototipos Interactivos.
- Tiene una línea de comandos (modo batch) para ser automatizado a gran escala aplicaciones.
- Tienen un API de Java que facilita el uso de RapidMiner para propios programas

#### **2.11.10.1. Formato de Archivos**

RapidMiner puede leer diversos formatos de entrada, como también leer y escribir modelos, conjuntos de parámetros y atributos. Los formatos más importantes que maneja son:

- Leer desde base de datos usando el operador “DatabaseExampleSource”
- Formatos de archivos separados por un carácter de línea, para ello emplea el operador “ExampleSource”. En este tipo de formato, la descripción de los atributos debe guardarse en un archivo XML con extensión. aml y los datos en un archivo de texto con el separador respectivo.

#### **2.11.10.2. Operadores**

Un operador es un proceso o algoritmo que ejecuta una tarea en especial. Cada operador se subdivide en varias partes:

- Grupo e icono del operador
- Una enumeración de las entradas y salidas de los objetos. Los objetos de entrada son generalmente consumidos por el operador y no son parte de la salida. En algunos casos, este comportamiento se puede modificar mediante un parámetro de mantener los datos en el modelo diseñado.
- Los parámetros de configuración del operador
- Una lista de valores que se pueden registrar usando el operador “ProcessLog”
- Si el operador representa un esquema de aprendizaje, las capacidades del aprendizaje son descritas.
- Si el operador representa un operador de cadena (chain) una breve descripción de los operadores internos es dada.
- Una descripción (corta y larga) del operador.
- De acuerdo a los grupos de operadores se tienen los siguientes:
- Operadores Básicos

### **2.11.10.3. Interfaz Grafica**

La interfaz gráfica para el diseño de modelos, se basa en las siguientes partes fundamentales:

- Árbol de procesos. En esta parte se define el modelo y los operadores a usar en este.
- Área de resultados. En esta parte se visualizan los resultados producto de las ejecuciones de los modelos diseñados, al igual que los parámetros de configuración de los diferentes operadores.
- Área de Compilación y Ejecución. En esta área se muestra el log de los procesos ejecutados en el árbol de procesos.

## 2.12. MÉTODO CIENTÍFICO

El método científico nos ayudará a descubrir propiedades del objeto de estudio a obtener nuevos conocimientos, que ha caracterizado históricamente a la ciencia, y que consiste en la observación sistemática, medición, experimentación, y la formulación, análisis y modificación de hipótesis (Francis Bacon).

Según el campo de acción y las implicancias del estudio, existe una serie de métodos que ayudan al descubrimiento. No será igual el método histórico al método lógico, así como no será igual el inductivo o el deductivo. Sin embargo, existe un método científico que predomina, y que se puede extrapolar a casi todas las ciencias. Tal vez, la primera característica que pueda decirse de este método es que se sustenta en leyes que dedujo el hombre, por lo que la efectividad del método es contrastable en el propio uso. Es un método racional, produce ideas que se combinan y pueden generar nuevas ideas y conceptos, hasta incluso un propio cambio en el método. Por estas características se puede decir que el método científico es verificable y explicativo. dos características surgen los dos preceptos que fundan al método, falsabilidad y reproducibilidad (Gargantilla, 2019).

- **Falsabilidad.** Apunta a que las proposiciones que este método considere como verdaderas no pueden dejar de estar sometidas a ser reevaluadas como falsas.
- **Reproducibilidad.** Quiere decir que la veracidad de una proposición puede ser replicada en otras condiciones, a menos que hubiera sido aclarado en la propia afirmación.

### 2.12.1. Fases del Método Científico

(Klein, 2016) definió el Método Científico como un proceso con los pasos siguientes:

- a) **Observación:** aplicar los sentidos a un objeto o a un fenómeno, para estudiarlos tal como se presentan en realidad, puede ser ocasional o causalmente;
- b) **Inducción:** extraer el principio fundamental de cada observación o experiencia;
- c) **Hipótesis:** explicación provisional de las observaciones o experiencias y sus posibles causas;
- d) Comprobación de la hipótesis por experimentación;
- e) Demostración o refutación (antítesis) de la hipótesis;
- f) Tesis o teoría científica.

El método científico lo utilizamos mucho más de lo que podríamos pensar a prioridad en nuestra vida. Así, por ejemplo, si observamos que un libro ha desaparecido de la estantería establecemos una hipótesis, es posible que se lo haya llevado alguien o bien que lo haya dejado en otro sitio sin darme cuenta.

### 2.13. METODOLOGÍA UWE

Según (Schwabe, Olsina, & Rossi, 2008) **UWE** nació a finales de la década de los 90 con la idea de encontrar una forma estándar para analizar y diseñar modelos de sistemas web. El objetivo por lo cual nació esta metodología fue utilizar un lenguaje común o por lo menos definir un metamodelo basado en el mapeo a lo largo de las diferencias etapas. En esa época UML prometía convertirse en un estándar para el modelamiento de sistemas. Por este motivo, UWE se adhirió a UML y no a otra técnica de modelado.

UWE se ha adaptado a las nuevas características de los sistemas web como transacciones, personalizaciones y aplicaciones asíncronas, y por otro lado ha evolucionado para incorporar técnicas de ingeniería de software como el modelamiento orientado aspectos y nuevos lenguajes de transformación para mejorar la calidad del diseño.

### **2.13.1. Características**

La metodología UWE define vistas especiales representadas gráficamente por diagramas en UML, tales como el modelo de navegación y el modelo de presentación. Los diagramas se pueden adaptar como mecanismos de extensión basados en estereotipos que proporciona UML. Estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son los que finalmente se utilizarán en las vistas especiales para el modelado de aplicaciones Web. De esta manera, se obtiene una notación UML adecuada para un dominio específico a la que se conoce como “Perfil UML”.

### **2.13.2. Meta Modelo de UWE**

Según (LMU – Ludwig-Maximilians, 2016) El meta modelo de UWE es una extensión del meta modelo de UML 2.0, esto quiere decir que UWE utilizados los elementos de los modelos sin ninguna modificación. Los elementos de los modelos que añade UWE están relacionados por herencia con al menos un elemento del modelo de UML. Para crear modelos de UWE para aplicaciones Web se puede utilizar cualquier herramienta CASE que soporte perfil e extensiones UML (Véase en figura 2.22).

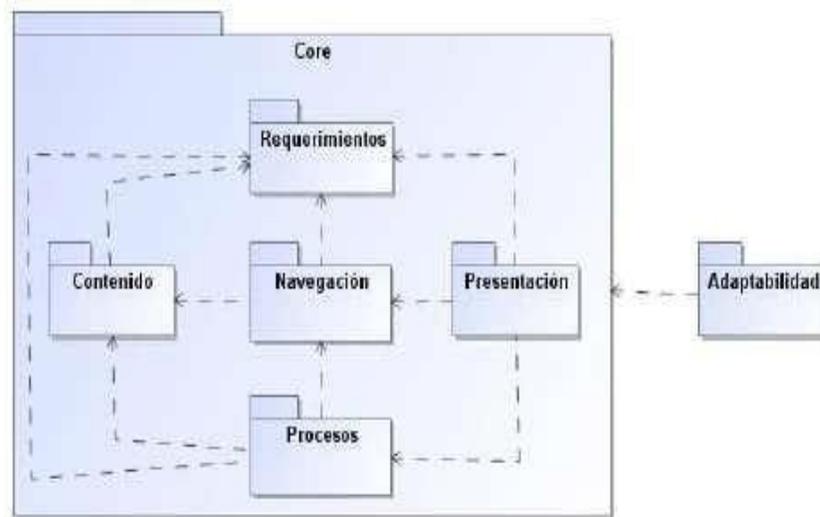


Figura 2. 22 Meta Modelo

Fuente: Rossi, Schwabe, & D. Olsina, 2008

- El paquete de Requerimientos comprende la extensión de UWE para los Casos de Uso
- Los paquetes de Navegación y Presentación corresponden a los modelos iguales de UML.
- Los paquetes de Contenido y Procesos actualmente se los utiliza como una parte para mostrar que UWE permite al diseñador desarrollar modelos de contenido y proceso usando todas las características de UML.
- El paquete de Adaptabilidad representa la característica más importante de las aplicaciones Web.

### 2.13.3. Actividades de modelado de UWE.

Las actividades base de modelado de UWE son el análisis de requerimientos, el modelo conceptual, el modelo navegacional y el modelo de presentación. A estos modelos se pueden sumar

otros modelos como lo son el modelo de interacción y la visualización de Escenarios Web (Galiano, 2012).

El proceso de UWE está dividido en cuatro actividades, que en el siguiente capítulo será profundizado con claridad, estos son los siguientes:

- Análisis de Requerimientos
- Modelo Conceptual
- Modelo de Navegación
- Modelo de Presentación

#### **2.13.3.1. Escenarios Web**

Permiten detallar la parte dinámica del modelo de navegación, especificando los eventos que disparan las situaciones, definen condiciones y explícitamente incluyen las acciones que son realizadas. Junto con el modelo de interacción temporal, los escenarios Web proveen la representación funcional dinámica del modelo de navegación (Galiano, 2012).

#### **2.13.3.2. Diagramas**

Los diagramas usados por UWE, son diagramas UML puro. Entre los más importantes tenemos: Diagramas de estado, de Secuencia, de colaboración y diagramas de Actividad (Galiano, 2012).

#### **2.13.4. Fases de Uwe**

Según (LMU – Ludwig-Maximilians, 2016)Uwe es una metodología dirigida o enfocada al modelado de aplicaciones Web, ya que está basada estrictamente en UML, esta metodología nos

garantiza que sus modelos sean fáciles de entender para los que manejan UML. (Véase en la figura 2.23) podemos ver una vista general de UWE, con las fases que tiene como.

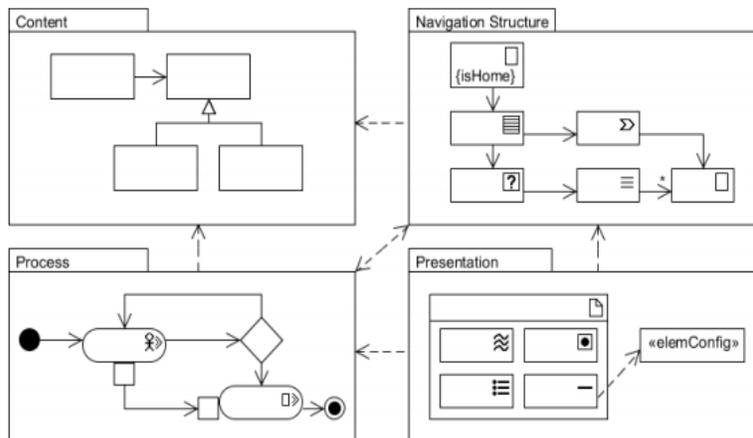


Figura 2. 23 Vista general de Modelos Uwe

Fuente: Nolivos y Coronel

#### 2.13.4.1. Fase Modelo de Análisis de Requerimiento

Según (LMU – Ludwig-Maximilians, 2016) El modelo de requerimiento permite diferenciar los procesos de navegación de los procesos de negocio mismo mediante el uso de diagramas de casos de uso para la captura de requisitos, que da como resultado un modelo de casos de uso acompañado de documentación que describe las reglas de adaptación, los usuarios y las interfaces (Véase en la figura 2.24).

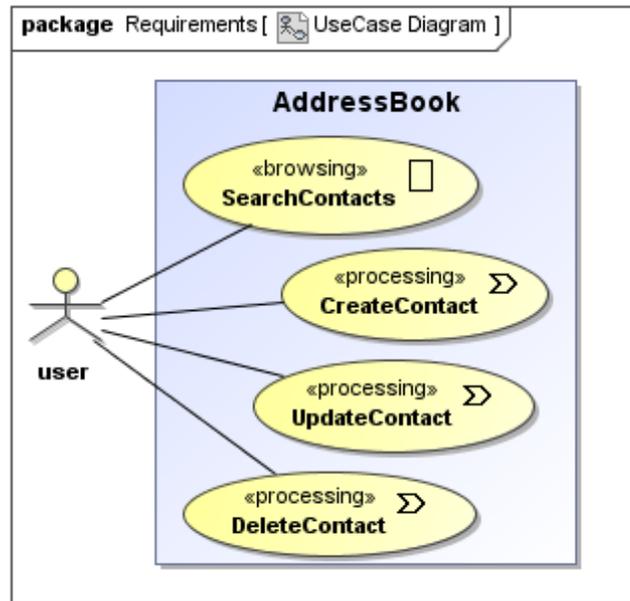


Figura 2. 24 Modelo de Caso de Uso

Fuente: LMU – Ludwig-Maximilians, 2016

#### 2.13.4.2. Fase Modelo de Conceptual

Según (LMU – Ludwig-Maximilians, 2016) Caracterizado por un modelo de dominio, que utiliza los requisitos que se detallan en los casos de uso. En esta etapa se representa el dominio del problema con un diagrama de clases de UML, que permiten determinar, métodos y atributos. El propósito de este diagrama es construir un modelo del dominio que intenta no considerar el paseo de la navegación, la presentación y los aspectos de interacción. Aspectos que se analizarán en los pasos respectivos de navegación y presentación de la planificación (Véase en la figura 2.25).

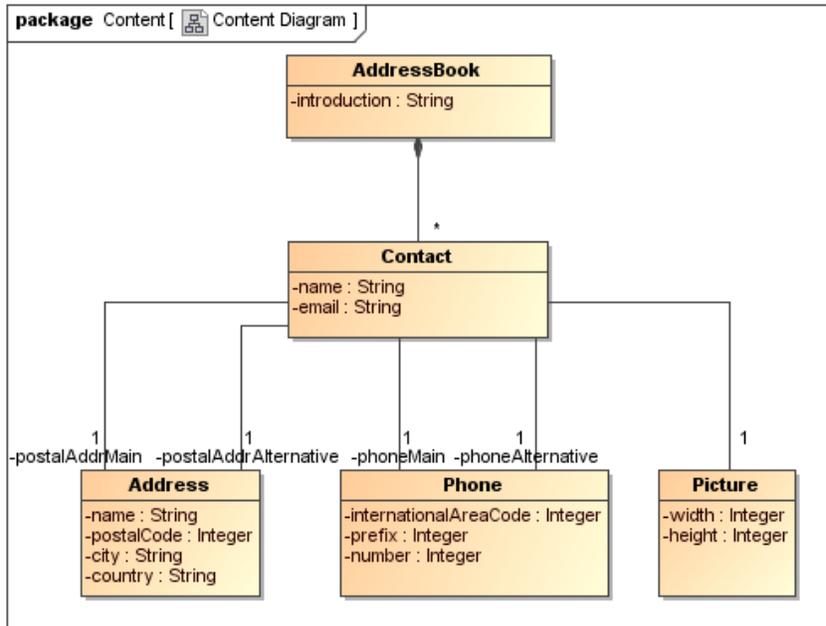


Figura 2. 25 Diagrama de contenido

Fuente: LMU – Ludwig-Maximilians, 2016.

### 2.13.4.3. Fase Modelo de Navegación

Según (LMU – Ludwig-Maximilians, 2016) Basado en el diagrama de la fase conceptual, donde se especifica los objetos que serán visitados dentro de la aplicación web y la relación entre los mismos. Su objetivo principal es representar el diseño y estructura de las rutas de navegación al usuario para evitar la desorientación en el proceso de navegación. Este modelo se destaca en el marco de UWE como el más importante, ya que representa elementos estáticos, a la vez que se pueden incorporar lineamiento semántico de referencia para las funcionalidades dinámicas de una aplicación Web (Véase en la figura 2.26).

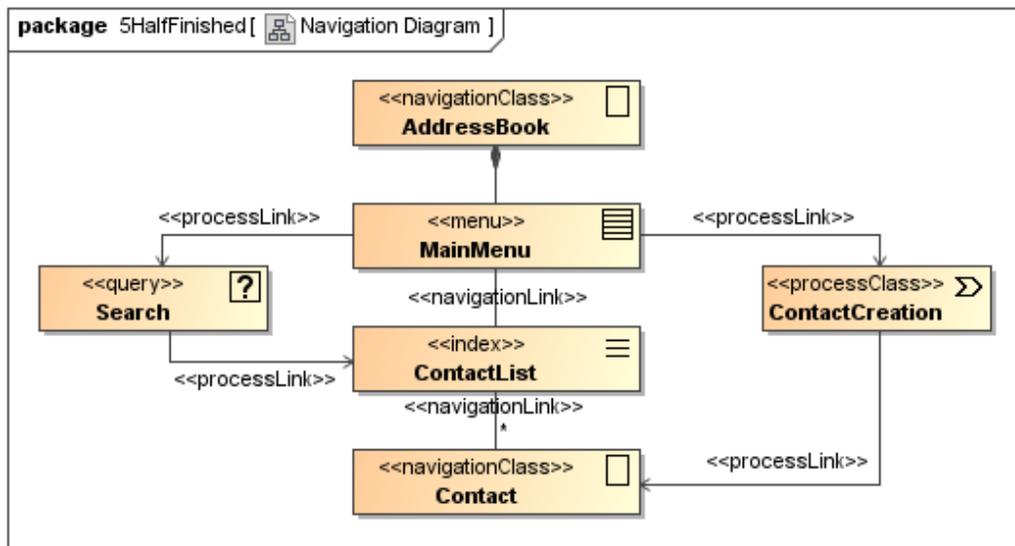


Figura 2. 26 Diagrama de navegación UML

Fuente: LMU – Ludwig-Maximilians, 2016

#### 2.13.4.3.1. Modelo del espacio de navegación

Basada en lo estructurado en la fase de conceptualización, es decir en los diagramas de clases

#### 2.13.4.3.2. Modelo de la estructura de navegación

Según (LMU – Ludwig-Maximilians, 2016) Muestra la forma de navegar ante el espacio de navegación. Están constituidas por menús, índices, visitas guiadas, y formularios (Véase en la figura 2.27).

- Los índices es la colección de objetos permitiendo una navegación directa.
- Las visitas guiadas compuesta por grupo de referencias, permitiendo una navegación secuencial.
- Un menú es un elemento parte de la navegación con un número específico de conexiones a otros objetos.

- Un formulario facilita al usuario ingresar información para completar las condiciones de selección de objetos pertenecientes a las colecciones de índices y visitas guiadas.

 clase de navegación	 menú
 índice	 pregunta
 visita guiada	 clase de proceso
 nodo externo	

Figura 2. 27 Nombre y símbolo de estereotipos Modelo de Navegación

Fuente: Nolivos y Coronel, [T-ESPE, 2013]

#### 2.13.4.4. Fase Modelo de Diseño de Presentación

Según (LMU – Ludwig-Maximilians, 2016) La fase de diseño de presentación tiene como objetivo la representación de las vistas del interfaz del usuario final, la representación gráfica de esta fase se encuentra basada en los diagramas realizados en las fases anteriores. Las clases del modelo de presentación representan páginas Web o parte de ellas, organizando la composición de los elementos de la interfaz de usuario y las jerarquías del modelo de presentación (Véase en la figura 2.28).

 grupo de presentación	 página de presentación
 texto	 entrada de texto
 ancla	 fileUpload
 botón	 imagen
 formulario	 componente de cliente
 alternativas de presentación	 selección

Figura 2. 28 Nombre y símbolo de estereotipos Modelo de Presentación

Fuente: Nolivos y Coronel, [T-ESPE, 2013]

El diagrama de esta fase representa los objetos de navegación y elementos de acceso, por ejemplo, en que marco o ventana se encuentra el contenido y que será remplazado cuando se accione un enlace. En la siguiente imagen podremos observar un ejemplo de un diagrama de presentación mediante UWE (Véase en la figura 2.29).

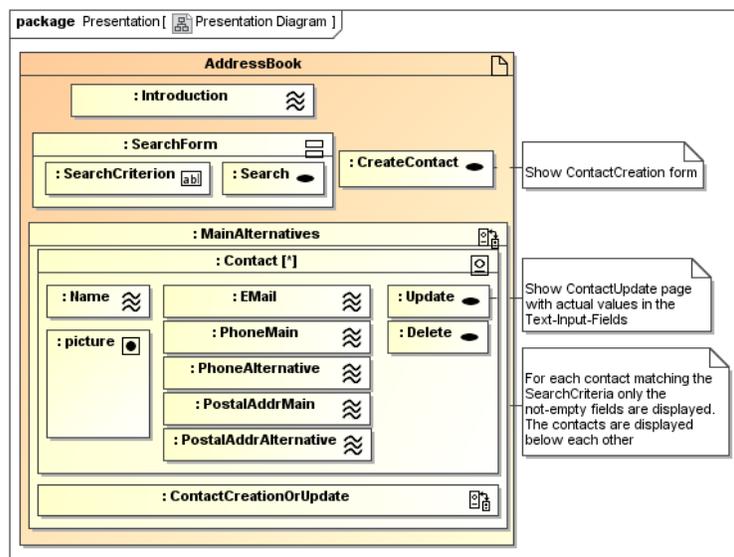


Figura 2. 29 Diseño de presentación UWE

Fuente: LMU – Ludwig-Maximilians, 2016

## 2.14. MÉTRICAS DE CALIDAD DE SOFTWARE

### 2.14.1. WebQem

Según (Medina Sanes, 2014) WebEQM es conceptualizado de la siguiente forma, es un modelo de evaluación propuesto por Luis Olsina en 1999 y está orientado en un inicio a la evaluación de la calidad de producto de software y está basado en modelos de calidad como ISO/IEC 9126-1 y modelos mixtos, el cual le otorga una gran flexibilidad pues se pueden evaluar diferentes perspectivas de calidad en un ente, sin perder los beneficios originales.

Los pasos básicos para la implementación de una evaluación WebEQM son:

- Definición de las metas de evaluación y selección del perfil de usuario.
- Definición y especificación de los requerimientos de calidad.
- Definición de criterios de los indicadores elementales y procedimientos de medición.
- Definición de modelos de agregación e implementación de la evaluación global.
- Análisis de resultados y recomendaciones.

#### 2.14.1.1. Características de calidad

En esta etapa se definirá una base de criterios para evaluación elemental, y realizar el proceso de medición y puntuación de los valores que se calcule para medir la calidad del software en tal caso se definirá los siguientes criterios (Gutierrez Aleman, 2017):

Tabla 2. 1  
*Criterios de Evaluación*

CARACTERÍSTICAS DE CALIDAD	MALA	REGULAR	BUENO	MUY BUENO
<b>Funcionalidad</b>	0-30%	31-50%	51-90%	91-100%
<b>Confiabilidad</b>	0-30%	31-50%	51-90%	91-100%
<b>Usabilidad</b>	0-30%	31-50%	51-90%	91-100%
<b>Mantenibilidad</b>	0-30%	31-50%	51-90%	91-100%
<b>Portabilidad</b>	0-30%	31-50%	51-90%	91-100%
<b>Total</b>	0-30%	31-50%	51-90%	91-100%

Esta tabla nos indica que nuestras características de calidad serán evaluadas según al rango al que se encuentren (Fuente: Gutierrez Aleman 2017).

**2.14.1.1.1. Usabilidad**

Según (Guitierrez Aleman, 2017) Es considerada como el grado de eficacia, eficiencia y satisfacción con la que usuarios específicos pueden lograr objetivos específicos, en contextos de uso específicos al utilizar un producto. Es uno de los factores más importantes que existen para elevar la calidad de los portales web, ya que juega un papel clave para generar experiencias positivas, para la mejora en la toma de decisiones y para contribuir a la creación de sitios web más eficientes. Por tanto, la usabilidad no es más que el arte y la ciencia de crear sistemas o sitios web que brindan una experiencia positiva al usuario, lo que implica facilidad en el manejo, facilidad en la búsqueda de información específica, efectividad del uso del tiempo que el usuario utiliza, error tolerante y acoplamiento.

**2.14.1.1.2. Funcionalidad**

Según (Guitierrez Aleman, 2017) Pues se refiere al correcto funcionamiento técnico del sitio. Al margen de que un sitio sea accesible y fácil de usar, si no lleva a cabo su función sin errores, no resultará de valor alguno para el usuario. Como se observa, la funcionalidad tiene una estrecha relación con la utilidad, pudiendo definirse funcionalidad con utilidad objetiva, es decir, la capacidad técnica de la aplicación para soportar las tareas que el usuario desea realizar. Su interrelación con la usabilidad se fundamenta en que muchos errores o fallos de funcionalidad son considerados al mismo tiempo propio de la usabilidad, debido a la frustración que provocan en el usuario durante la interacción.

Funcionalidad es una métrica orientada a la función del sistema y al proceso por el cual se desarrolla. Se centra en la funcionalidad o utilidad del programa.

- **Entradas del usuario**, se toma en cuenta cada entrada del usuario que el sistema proporciona a medida que ingresa al sistema.
- **Salidas del usuario**, se refleja las salidas que tiene el sistema tanto reportes como estadísticas que tiene el sistema.
- **Número de peticiones del usuario**, una petición se define como una entrada interactiva que resulta de la generación de algún tipo de respuesta en forma de salida. Número de archivos, se define cada archivo lógico.
- **Número de interfaces externas**, se definen todas aquellas interfaces legibles por el ordenador que solicitan transmitir información a otro sistema.

La relación que permite calcular los puntos de función es la siguiente:

$$PF = Cuenta\_total * (Grado\_de\_confiabilidad + tasa\_de\_error * \sum Fi)$$

Donde:

- **PF** = Medida de funcionalidad.
- **Cuenta\_total** = Es la suma de los siguientes datos: Número de entradas, número de salidas, número de peticiones, número de archivos y número de interfaces externas.
- **Grado\_de\_confabilidad** = Es la confiabilidad estimada del sistema
- **Tasa\_de\_error** = Probabilidad subjetiva estimada del dominio de la información, este error estimado es de 1%.
- **Fi** = Son valores de ajuste de complejidad que toman los valores

Con la obtención de los anteriores datos y considerando un grado de confiabilidad mínimo del 65% es que a continuación calculamos el valor de PF

$$PF = Cuenta\_total * (Grado\_de\_confiabilidad + tasa\_de\_error * \sum Fi)$$

Si consideramos el máximo valor de ajuste de complejidad como  $\sum Fi = 70$

Entonces si  $\sum Fi$  es considerada como el 100% la relación obtenida entre los puntos será

$$Funcionalidad = \left( \frac{PF}{PF_{maximo}} \right)$$

#### 2.14.1.1.3. Confiabilidad

Según (Guitierrez Aleman, 2017) Se puede definir como la capacidad de un sitio de realizar sus funciones de la manera prevista. De otra forma, la confiabilidad se puede definir también como la probabilidad en que un producto realizará su función prevista sin incidentes por un período de tiempo especificado y bajo condiciones indicadas. Se consideran tres características para este atributo tales como: proceso correcto de enlace, recuperación de errores y la validación y recuperación de la entrada del usuario. Lo que indica que, si un sitio web contiene varios enlaces rotos o inválidos, degrada la confiabilidad del mismo desde el punto de vista del usuario.

- Enlaces
  - Enlaces Rotos
  - Enlaces Inválidos
  - Enlaces no Implementados
- Paginas
  - Páginas Muertas

- Páginas bajo Construcción
- Errores de Ortografía

Entonces tenemos las siguientes probabilidades:

$$P(T \leq t) = F(t) \quad (1) \text{ Probabilidad de fallas}$$

$$P(T \leq t) = 1 - F(t) \quad (2) \text{ Probabilidad de trabajo sin fallas}$$

Donde a que se tienen tiempos de inicio como de fin, para el cálculo de estas probabilidades se utiliza la distribución exponencial. Entonces la función  $F(t)$  está dada por:

$$F(t) = \text{funcionalidad } e^{-\lambda * t} \quad \text{Ec. De confiabilidad}$$

Para el cálculo de error se consideran 10 ejecuciones en un mes en un periodo de 12 meses para obtener una probabilidad de fallas. En ese caso, el margen de error  $\lambda$  será de 1/10. Además, se utiliza el resultado obtenido del punto función igual a 84% para la distribución en un periodo  $t= 12$  meses.

Obteniendo la portabilidad de fallas se tiene:

$$P(T \leq t) = F(t)$$

$$F(t) = \text{funcionalidad} * e^{-\frac{1}{10} * 12}$$

#### 2.14.1.1.4. Mantenimiento

Según (Guitierrez Aleman, 2017) Capacidad del producto software de ser modificado y probado. La mantenibilidad es la facilidad con que una modificación puede ser realizada. Está indicada por los siguientes sub atributos.

- Facilidad de análisis
- Facilidad de cambio

- Estabilidad
- Facilidad de prueba

El índice de madurez del software se calcula con la siguiente formula

$$IMS = \frac{[Mt - (Fa + Fb + Fc)]}{Mt}$$

Donde

***Mt*** = Número de módulos en la versión actual

***Fa*** = Número de módulos en la versión actual que se han cambiado

***Fb*** = Número de módulos en la versión actual que se han añadido

***Fc*** = Número de módulos en la versión anterior que se han borrado en la versión actual

#### 2.14.1.1.5. Portabilidad

Según (Guitierrez Aleman, 2017) Capacidad del producto software de ser transferido de un ambiente a otro. La portabilidad es la capacidad que tiene el sistema para ser trasladado de un entorno a otro. Para poder medir la portabilidad del sistema usaremos la siguiente formula que indica el grado de portabilidad que tiene un software.

$$GP = 1 - \frac{ET}{ER}$$

Donde:

***ET*** = Es la medida de los recursos necesarios para llevar el sistema a otro entorno.

***ER*** = Es la medida de los recursos necesarios para crear el sistema en el entorno residente.

Si  $GP > 0$ , la portabilidad es más rentable que el redesarrollo

Si  $GP = 1$ , la portabilidad es perfecta

Si  $GP < 0$ , el re-desarrollo es más rentable que la portabilidad.

## 2.15. COSTOS DE SOFTWARE

### 2.15.1. Cocomo II

Según (Boehm, 2011) Los objetivos principales que se tuvieron en cuenta para construir el modelo COCOMO II fueron:

- Desarrollar un modelo de estimación de costo y cronograma de proyectos de software que se adaptara tanto a las prácticas de desarrollo de la década del 90 como a las futuras.
- Construir una base de datos de proyectos de software que permitiera la calibración continua del modelo, y así incrementar la precisión en la estimación.
- Implementar una herramienta de software que soportara el modelo.
- Proveer un marco analítico cuantitativo y un conjunto de herramientas y técnicas que evaluaran el impacto de las mejoras tecnológicas de software sobre los costos y tiempos en las diferentes etapas del ciclo de vida de desarrollo.

Según (Pressman, 2010) COCOMO original se convirtió en uno de los modelos de estimación de costo más ampliamente utilizados y estudiados en la industria. Evolucionó hacia un modelo de estimación más exhaustivo, llamado COCOMO II.

Como su predecesor, COCOMO II en realidad es una jerarquía de modelos de estimación que aborda las áreas siguientes:

- **Modelo de composición** de aplicación. Se usa durante las primeras etapas de la ingeniería de software, cuando son primordiales la elaboración de prototipos de las interfaces de usuario, la consideración de la interacción del software y el sistema, la valoración del rendimiento y la evaluación de la madurez de la tecnología.

- **Modelo de etapa de diseño temprano.** Se usa una vez estabilizados los requisitos y establecida la arquitectura básica del software.
- **Modelo de etapa post-arquitectura.** Se usa durante la construcción del software

### 2.15.1.1. Nivel Post-arquitectura

Según (Boehm, 2011) Es el modelo de estimación más detallado y se aplica cuando la arquitectura del proyecto está completamente definida. Este modelo se aplica durante el desarrollo y mantenimiento de productos de software incluidos en las áreas de Sistemas Integrados, Infraestructura y Generadores de Aplicaciones.

El esfuerzo nominal se ajusta usando 17 factores multiplicadores de esfuerzo. El mayor número de multiplicadores permite analizar con más exactitud el conocimiento disponible en las últimas etapas de desarrollo, ajustando el modelo de tal forma que refleje fielmente el producto de software bajo desarrollo. La fórmula para el cálculo del esfuerzo es la siguiente (Véase en la figura 2.32):



Figura 2. 30 Cocomo II

Fuente: Boehm 1995

**Punto de función**

$$\text{Punto Función} = \text{Cuenta Total} * \text{Factor de Ajustes}$$

**Costo del software desarrollado**

Se utiliza como punto (PF)  $PF = Total$ , realizaremos la conversión de punto función a miles de líneas de código mediante la siguiente tabla

$$E_D = 2,4(KLDC)^{1.05}$$

$$T_D = 2,5(E_D)^{0.38}$$

Donde:

$E_D$  = Esfuerzo aplicado en personas por mes

$T_D$  = Tiempo de desarrollo en mes

KLDC = número estimado de líneas de código distribuidas.

$$LCD = PF * Factor = LCD$$

Calculando el esfuerzo en personas mes:

$$E_D = 2.4 * (KLDC)^{1.05} = 5.78 \text{ personas/mes}$$

Calculando el tiempo de desarrollo mes

$$T_D = 2.5 * (E_D)^{0.38}$$

Calculando el personal requerido para el desarrollo del proyecto se obtiene con la siguiente formula:

$$\text{Numero de programadores} = \frac{E_D}{T_D}$$

Estimación del costo del software se calculará con la siguiente formula.

$$\text{Costo Software} = \text{programadores} * \text{salario de programador} * \# \text{ de mes}$$



# CAPITULO III

**MARCO APLICATIVO**

El propósito del capítulo es poner en práctica lo mencionado en los anteriores capítulos, que se analiza, desarrolla el modelo de minería de datos que satisfaga las necesidades en el análisis de redes sociales. La implementación del modelo y la arquitectura del algoritmo de Minería de datos

**3.1. DESCRIPCIÓN DEL MODELO**

La minería de datos o exploración de datos, es un proceso que descubre patrones de grandes volúmenes de conjuntos de datos, hasta ahora desconocidos. Estos patrones pueden ser vistos como una especie de resumen de los datos de entrada y pueden ser utilizados en el análisis.

Por esta razón un paso metodológico para la construcción de un modelo adecuado evidencia la necesidad de contar la construcción por etapas. También el proceso de minería de datos exige una cobertura de datos confiables y actualizados. Para un análisis exitoso es necesario que la base de datos contenga mayor cantidad existente de información.

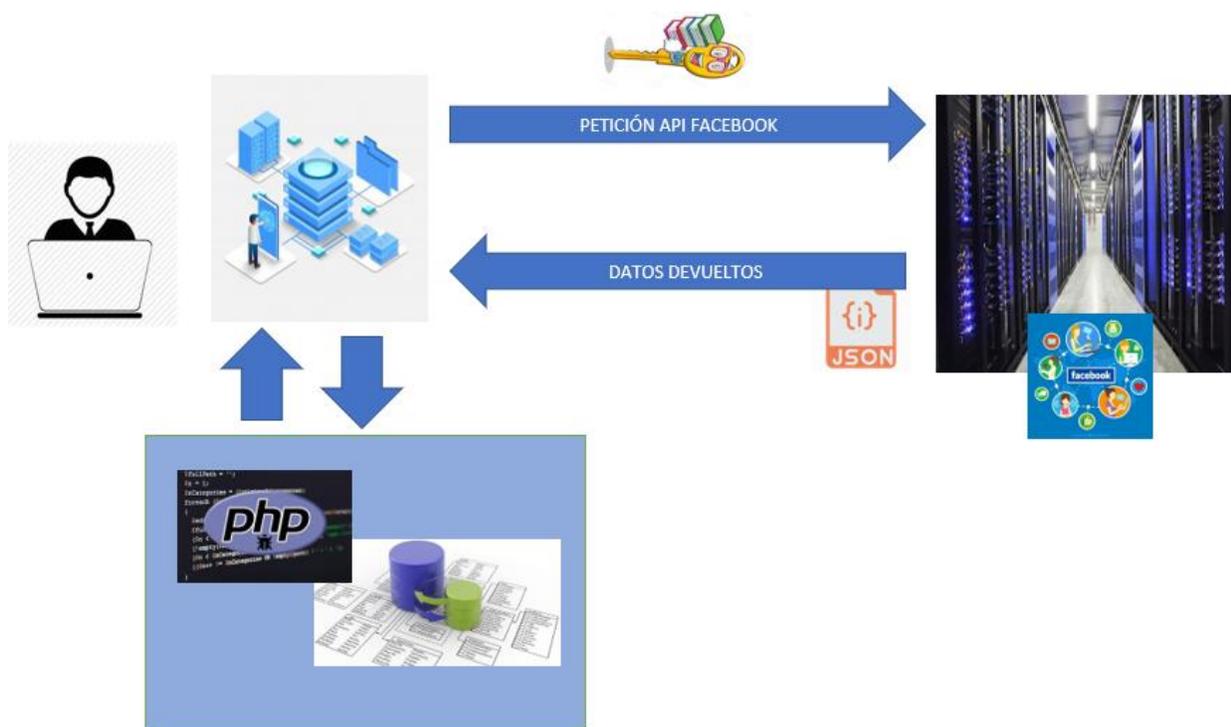


Figura 3. 1 Esquema del sistema

Fuente: Elaboración Propia

### 3.2. FASE DE INICIO

Esta fase tiene como propósito principal definir y acordar el alcance del proyecto, identificar los riesgos asociados al proyecto, proponer una visión muy general de la arquitectura de software. Cuyo objetivo es el modelado de requerimientos de alto nivel. Esta fase es muy breve si se trata de un problema conocido o se ha decidido realizar el proyecto de todas formas.

### 3.2.1. Descripción de Actores del Caso de Uso General

Identificación y descripción de los actores del diagrama de caso de uso General (véase en la tabla 3.1)

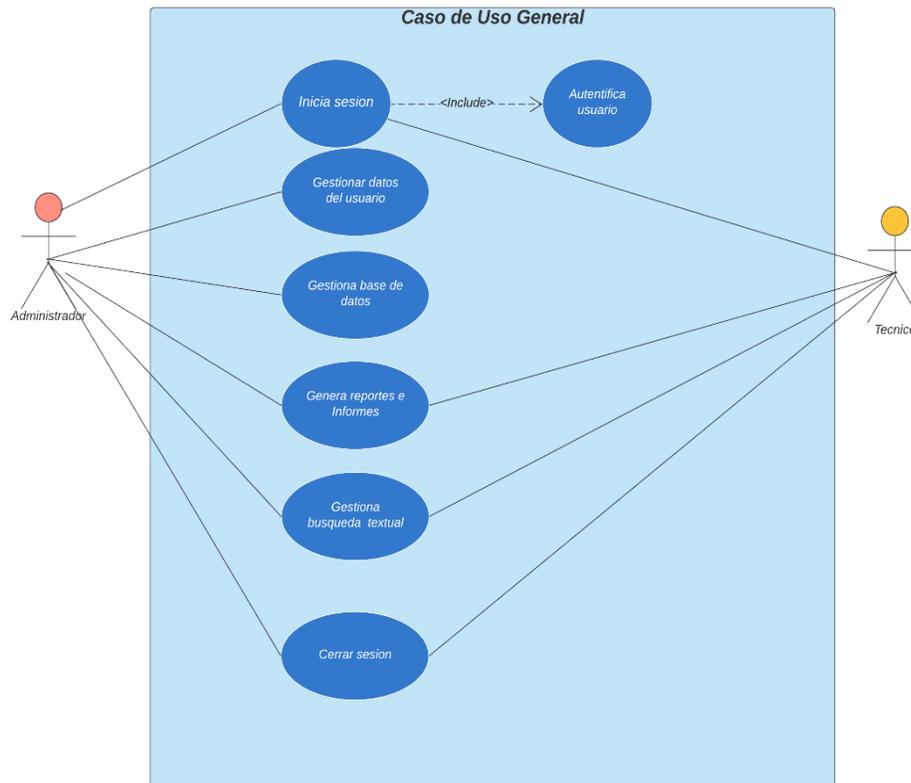


Figura 3. 2 Caso de Uso General

Fuente: Elaboración Propia

Tabla 3. 1  
*Descripción de Actores del Caso de Uso General*

ACTOR	DESCRIPCIÓN
<b>ADMINISTRADOR</b>	Es la persona que interactúa directamente con el sistema. Empezando inicialmente con la búsqueda en redes sociales. Administra toda la información con respecto a las actividades realizadas en el sistema como ser inserta o actualiza para luego hacer un informe digital y guardar toda la información obtenida.
<b>TÉCNICO</b>	Es encargado de iniciar sesión, buscar palabras textuales y generar un reporte sobre la publicación en temas políticos.

---

Fuente: Elaboración Propia

### 3.2.2. Modelo De Requerimientos

Es el conjunto de técnicas y procedimientos que nos permiten conocer los elementos principales para luego a pasar a desarrollar un proyecto de software. El desarrollo de nuestro proyecto se basa en el método científico que nos servirá de apoyo para la organización del proceso de investigación, lo cual se cubrirá los requerimientos para el cumplimiento de los objetivos planteados

### **3.2.2.1. Descripción de Requerimientos a Nivel Técnico**

El sistema nuestro es un Sistema Web por lo tanto lo primero es analizar es cliente /servidor. Donde el cliente realiza peticiones de cualquier operación. Para el desarrollo del sistema a nivel de software y hardware se utilizarán distintas herramientas que se detallan a continuación:

- La pieza fundamental del sistema (Extraer patrones).
- En la parte de codificación o programación para el desarrollo web utilizaremos la plataforma PHP cuyo código es interpretado por un servidor web con un módulo de procesador que genera la página web.
- En la parte de autenticación de usuario se utilizará JQuery con bibliotecas de Java Script. Y CSS que utilizaremos para organizar la presentación y aspectos de una página web.
- Para realizar reportes se utilizará librerías o módulos de PHP cuyos reportes serán exportados en formato pdf.
- Servidor de base de datos PostgreSQL, para gestionar datos relacionales.
- Opcional, tener un servidor en la red.

### **3.2.2.2. Descripción de Requerimientos a Nivel de Sistema**

Los requerimientos a nivel de sistema son estrictamente utilizando el extraer patrones, para realizar las distintas operaciones que se detallan a continuación:

- Para poder utilizar el sistema debe tener al iniciar un interfaz de autenticación o login de usuario, solo para personal autorizado.
- El sistema debe una interfaz para el registro completo de datos personales.

- El sistema debe contar con una búsqueda que permita mostrar palabras a elegir de política tales como: elecciones y otros
- El sistema debe tener otra interfaz, que visualizará todo un reporte de la publicación en Facebook.

### **3.2.2.3. Descripción de Requerimientos a Nivel de Usuario**

Los requerimientos a nivel de usuario en su mayoría son los reportes e informes de exigencia y estos son:

- Registro de datos personales de usuarios para su ingreso al sistema.
- Optimizar el control de cada publicación

### **3.3. FASE DE ELABORACIÓN**

En esta fase se determinarán las soluciones técnicas del proyecto, por sobre todo se profundiza la extracción de la publicación en Facebook que se elaborarán a nivel de diseño, mediante el modelo de análisis realizando los modelos de casos de uso del sistema, casos de uso extendidos, diagramas de secuencia, estado, clases y diagrama de navegación, finalmente con el modelo de diseño se valida la arquitectura. lo propio de fase de inicio porque es ahí donde definimos nuestros requerimientos principales para luego en esta fase elaborar.

### 3.3.1. Modelado de Análisis

#### 3.3.1.1. Modelo De Casos De Uso

Los casos de uso representan la iteración entre los usuarios y el sistema. Este modelo de casos de uso se desarrolla a lo largo de varias iteraciones, aquí además se establecerá lo que el sistema debe realizar, definir los límites del sistema; añadiendo nuevos casos de uso describiremos los actores del sistema, los casos de uso priorizado que se seleccionaron en el modelo del sistema.

#### 3.3.1.2. Diagrama de Caso De Uso Priorizado O de Alto Nivel

Después de haber realizado un conjunto de casos de uso y haberlos analizado, es necesario formalizar o priorizar los casos de uso, sin afectar los requerimientos del sistema a continuación se observa estos casos de uso y los actores que intervienen en el sistema. (Ver figura 3.2)

#### 3.3.1.3. Descripción de Casos de Uso

En esta sección se presentan los casos de uso del sistema, los cuales describen la secuencia de eventos que realiza un actor cuando el sistema lleva a cabo un proceso. Además, proporciona un medio por el cual las personas involucradas en el sistema, tanto los usuarios finales como el equipo de desarrollo, lleguen a una comprensión de éste.

- a) **Caso De Uso: Iniciar Sesión** El iniciar sesión describe como un usuario ingresa al sistema web, previamente accede a la página de ingreso del sistema, ingresa su nombre de usuario y contraseña, si los datos son correctos ingresa al sistema, si no despliega login (inicio) (ver figura 3.2).

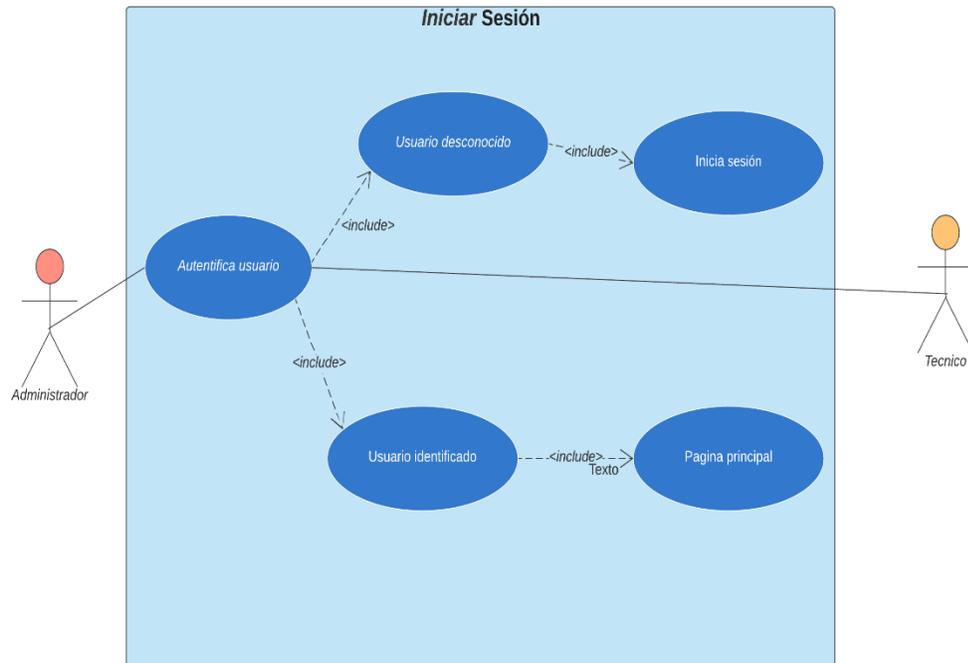


Figura 3. 3 Iniciar Sesión

Fuente: Elaboración propia

Tabla 3. 2  
Iniciar Sesión

CASO DE USO: INICIO DE SESIÓN	
<b>ACTOR</b>	<b>Personal (ADMINISTRADOR)</b> <b>Personal (TÉCNICO).</b>
<b>DESCRIPCIÓN</b>	Este caso de uso muestra como un usuario ingresa al sistema.
<b>PROPÓSITO</b>	Permite el acceso solo a personas autorizadas
<b>REQUISITOS</b>	El encargado debe estar registrado como técnico vigente. Exista el visto bueno por Administrador y asignar la nueva sesión.

<b>FLUJO DE EVENTOS</b>	Evento de Actor	Evento del Sistema
	1. El usuario ingresa su nombre de usuario y contraseña propia para el uso del sistema.	1. En caso el nombre y/o la contraseña sean incorrectos se redireccionará al inicio.
	2. El sistema verifica que exista el usuario y que la contraseña concuerde con la asignada.	2. En caso el usuario exista, pero la contraseña sea incorrecta o se olvidó se
	4. Se verifica el estado del usuario.	indicará esto mediante un mensaje email.
<b>POSTCONDICIÓN</b>	El usuario es reconocido por el sistema ingresa a la página principal.	

Fuente: Elaboración Propia

- b) Caso de uso: Gestionar datos del usuario** En este caso se realiza la tarea de obtener datos del usuario y hace su registro correspondiente para su posterior comparación de datos guarda datos en la base de datos al momento de realizar su registro por primera vez, todo esto mediante login o inicio de sesión que es muy eficiente la hora de realizar el registro (ver figura 3.3).

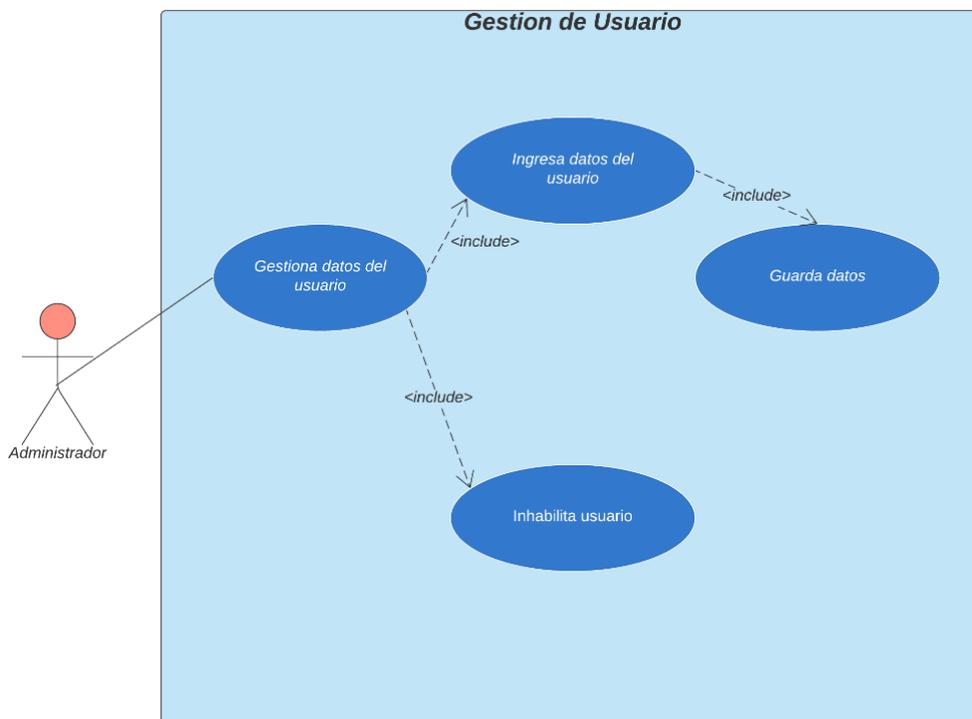


Figura 3. 4 Gestión de Usuario

Fuente: Elaboración propia

Tabla 3. 3  
Gestionar datos del Usuario

CASO DE USO: GESTIONAR DATOS DEL USUARIO		
<b>ACTOR</b>	Personal (ADMINISTRADOR)	
<b>DESCRIPCIÓN</b>	Este caso de uso permite el control de las cuentas de usuario, es decir, el registro de nuevo usuario, la modificación, y la desactivación de una cuenta de usuario	
<b>PROPÓSITO</b>	Permite el acceso solo a personas autorizadas	
<b>REQUISITOS</b>	Ser personal	
<b>FLUJO DE</b>	<b>Evento de Actor</b>	<b>Evento de sistema</b>
	1. El usuario ingresa al inicio de sesión.	1. El sistema valida los datos.

<b>EVENTOS</b>	2. El usuario asigna contraseña y un nombre al nuevo registro. 3. El usuario ingresa a la página principal y guarda los datos	2. El sistema guarda los datos en la base de datos 3. Si el sistema encuentra datos que no concuerdan con los requeridos, el sistema despliega una alerta de informe que no se introdujo los datos.
<b>POSTCONDICIÓN</b>	El usuario asigna una contraseña y un nombre de usuario al nuevo registro, los datos son almacenados	

Fuente: Elaboración propia

c) **Caso de uso: Gestionar base de datos** El administrador guardará todos los datos del usuario donde extraerá los datos (ver figura 3.4).

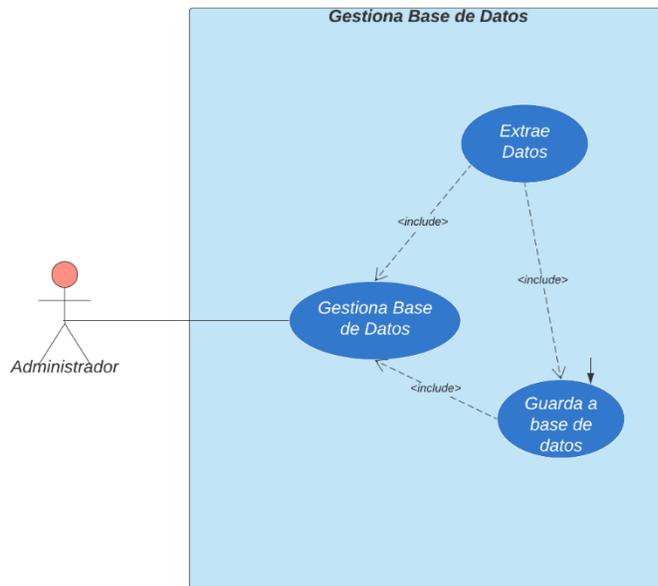


Figura 3. 5 Gestionar Base de Datos

Fuente: Elaboración Propia

Tabla 3. 4  
*Gestionar Base de Datos*

<b>CASO DE USO GESTIONAR BASE DE DATOS</b>		
<b>ACTOR</b>	Personal (ADMINISTRADOR).	
<b>DESCRIPCIÓN</b>	Almacenara los datos del usuario	
<b>PROPÓSITO</b>	Recargar información de Facebook	
<b>REQUISITOS</b>	Ser administrador	
<b>FLUJO DE EVENTOS</b>	<b>Evento Actor</b>	<b>Evento de Sistema</b>
	Ve el registro de datos ingresados en la base de datos	Muestra todo el resultado solo al administrador
<b>POSTCONDICIÓN</b>	Administrador guarda los datos de información	

---

Fuente: Elaboración Propia

- d) **Caso de uso: Genera reporte e informe** Al finalizar cada proceso u operación es muy necesario realizar un informe detallando de la extracción en Facebook realizadas y otros datos como ser: Información de la publicación (ver figura 3.5).

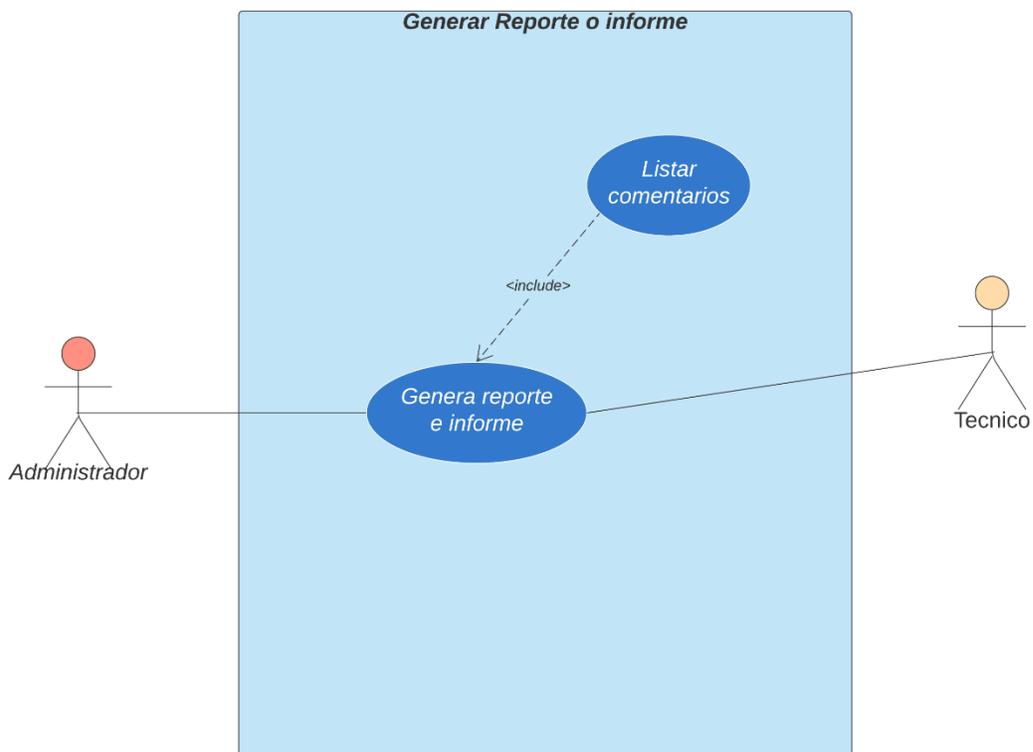


Figura 3. 6 Genera Reporte o Informe

Fuente: Elaboración Propia

Tabla 3. 5  
Generar Reporte o Informe

CASO DE USO: GENERA REPORTE E INFORME		
<b>ACTOR</b>	Personal (ADMINISTRADOR), Personal (TÉCNICO).	
<b>DESCRIPCIÓN</b>	El sistema realiza informes y reportes de cada publicación	
<b>PROPÓSITO</b>	Generar informes y reportes.	
<b>REQUISITOS</b>	Deben existir datos válidos y certeros.	
<b>FLUJO</b>	<b>Eventos del Actor</b>	<b>Eventos del Sistema</b>

<b>DE</b>	1. El administrador elige	1. El sistema muestra la
<b>EVENTOS</b>	la opción de generar informes y reportes en formatos pdf. Para luego ser impresos.	interfaz de generar informes y reportes. 2. El sistema genera informes en formatos pdf. 3. Informes que deben estar a disposición en cualquier momento que se los necesite.
<b>POSTCONDICIÓN</b>	Informes y reportes entregados al directorio	

Fuente: Elaboración Propia

e) **Caso de uso: Gestionar búsqueda textual** Puedes usar palabras en tu búsqueda ejemplo: Vicerrectorado, U.P.E.A., Posgrado, programas etc., y otros para obtener resultados más precisos (ver figura 3.6).

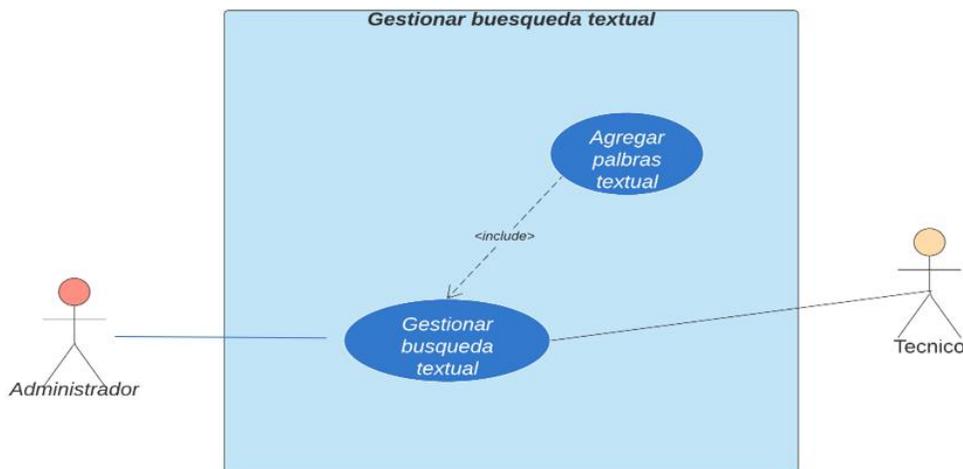


Figura 3. 7 Gestión Búsqueda textual

Fuente: Elaboración Propia

Tabla 3. 6  
*Gestionar Búsqueda Textual*

<b>CASO DE USO: GESTIONAR BÚSQUEDA TEXTUAL</b>		
<b>ACTOR</b>	Personal (ADMINISTRADOR), Personal (TÉCNICO).	
<b>DESCRIPCIÓN</b>	Busca palabras específicas	
<b>PROPÓSITO</b>	Extraer información textual	
<b>REQUISITOS</b>	Ser personal autorizada	
<b>FLUJO</b>	<b>Eventos de actor</b>	<b>Evento de sistema</b>
<b>DE</b>	El administrador y Técnico	1. El sistema muestra el
<b>EVENTOS</b>	elige la opción de generar resultado de la búsqueda en la búsqueda textual y para luego ser mostrados.	página principal.
<b>POSTCONDICIÓN</b>	La búsqueda textual se mostrará en la página principal	

Fuente: Elaboración propia

3.3.2. Modelo Conceptual

El modelo conceptual se basa en el análisis de requisitos en los casos de uso realizados anteriormente. Incluye los objetos implicados en la interacción entre el usuario y la aplicación, el diseño conceptual tiene como objetivo construir el diagrama de clases. El diagrama de diseño conceptual, describe cada una de las clases de dominio del sistema web y la relación con cada una de las clases, se presenta (ver figura 3.7).

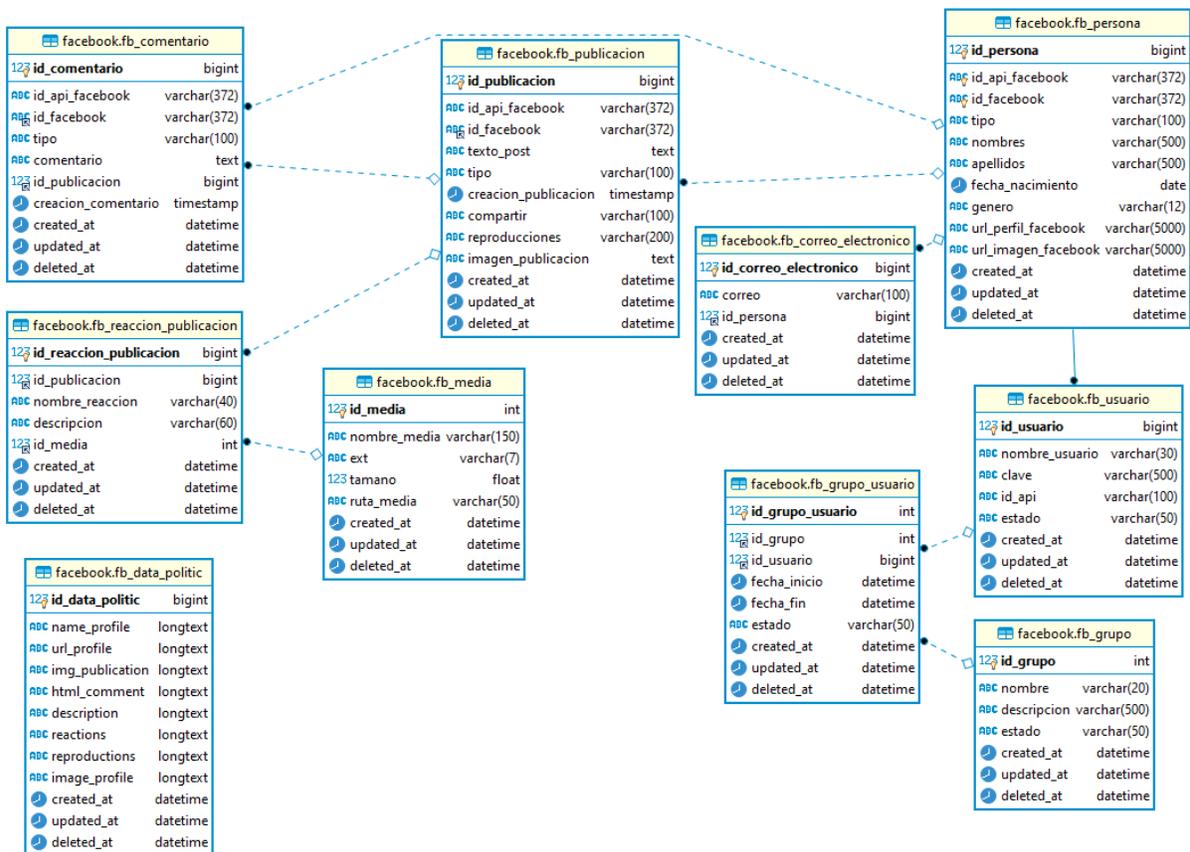


Figura 3. 8 Modelo Conceptual

Fuente: Elaboración Propia

### 3.3.3. Modelo De Navegación

En la fase de diseño de navegación la metodología UWE, selecciona los diagramas apropiados para mejorar la expresión o visualización de las construcciones del dominio de la aplicación web. Adicionalmente UWE introduce “clases navegacionales” que son parte del modelo de navegación y otros elementos de acceso (ver figura 3.8).

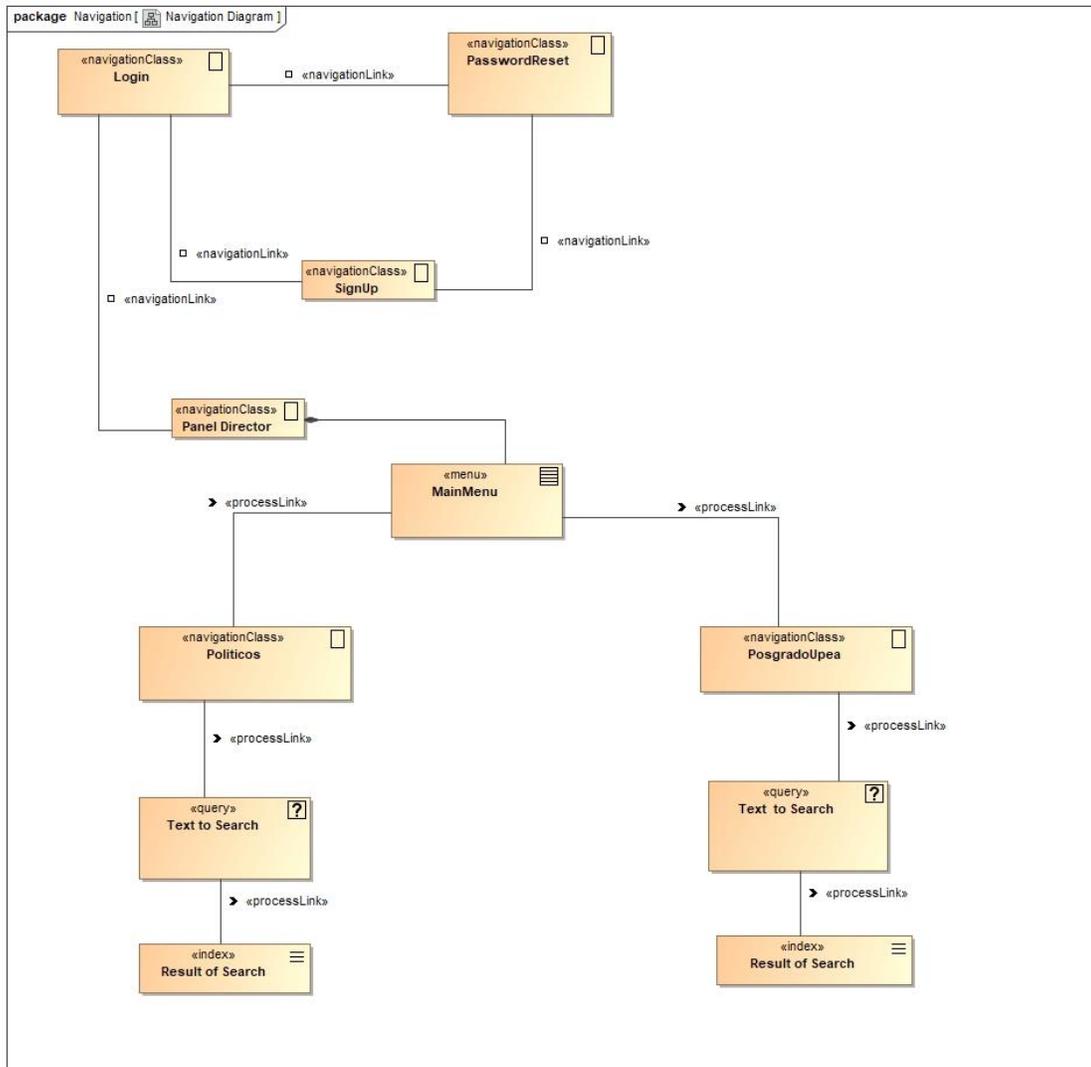


Figura 3. 9 Modelo Navegacional

Fuente: Elaboración Propia

### 3.3.4. Modelo De Presentación

El modelo de presentación pretende proporcionar una representación abstracta de la interfaz de usuario final y definir la interacción de las clases navegables con el usuario. Basada en el modelo de navegación (ver figura 3.9)

#### 3.3.4.1. Modelo De Presentación De Iniciar Sesión

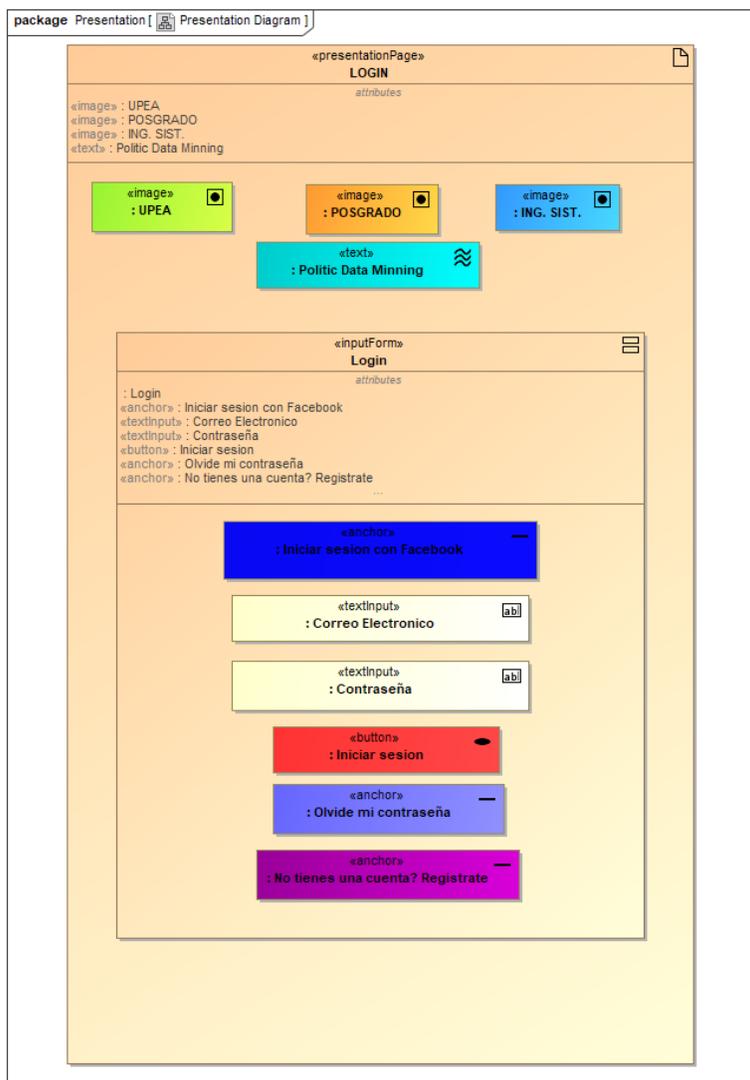


Figura 3. 10 Modelo de presentación de Iniciar Sesión

Fuente: Elaboración Propia

### 3.3.4.2. Recuperar Contraseña

Si el usuario se olvidó la contraseña el administrador podrá enviarle la nueva contraseña a su correo electrónico (ver figura 3.10).

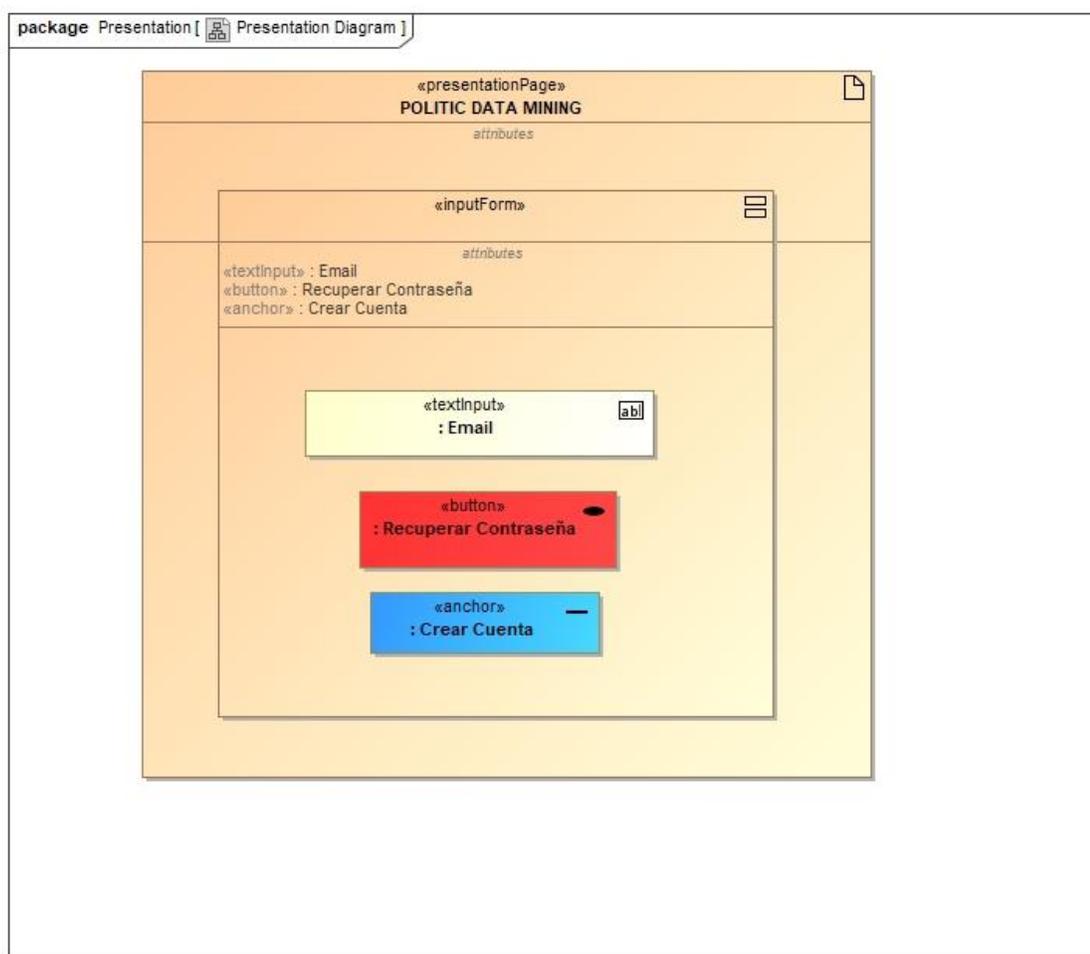


Figura 3. 11 Recuperar Contraseña

Fuente: Elaboración Propia

### 3.3.4.3. Registro de Usuario

El diagrama de presentación de la administración del usuario, permite visualizar las interfaces que implica el registro del usuario de sus datos personales (ver figura 3.11).

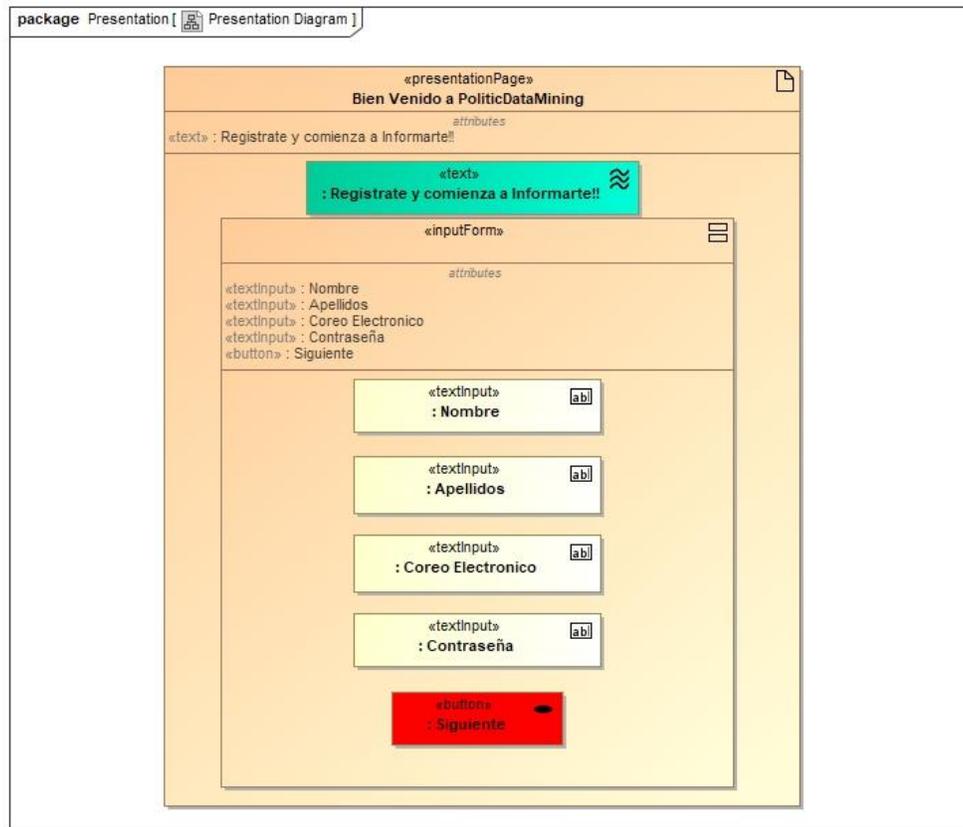


Figura 3. 12 Registro de Usuario

Fuente: Elaboración Propia

### 3.3.4.4. Página Principal

Donde se mostrará la vista de todo de los comentarios de la publicación, los usuarios registrados en Facebook, la base de datos en este caso solo el administrador tendrá el acceso y el reporte de la publicación (ver figura 3.12).

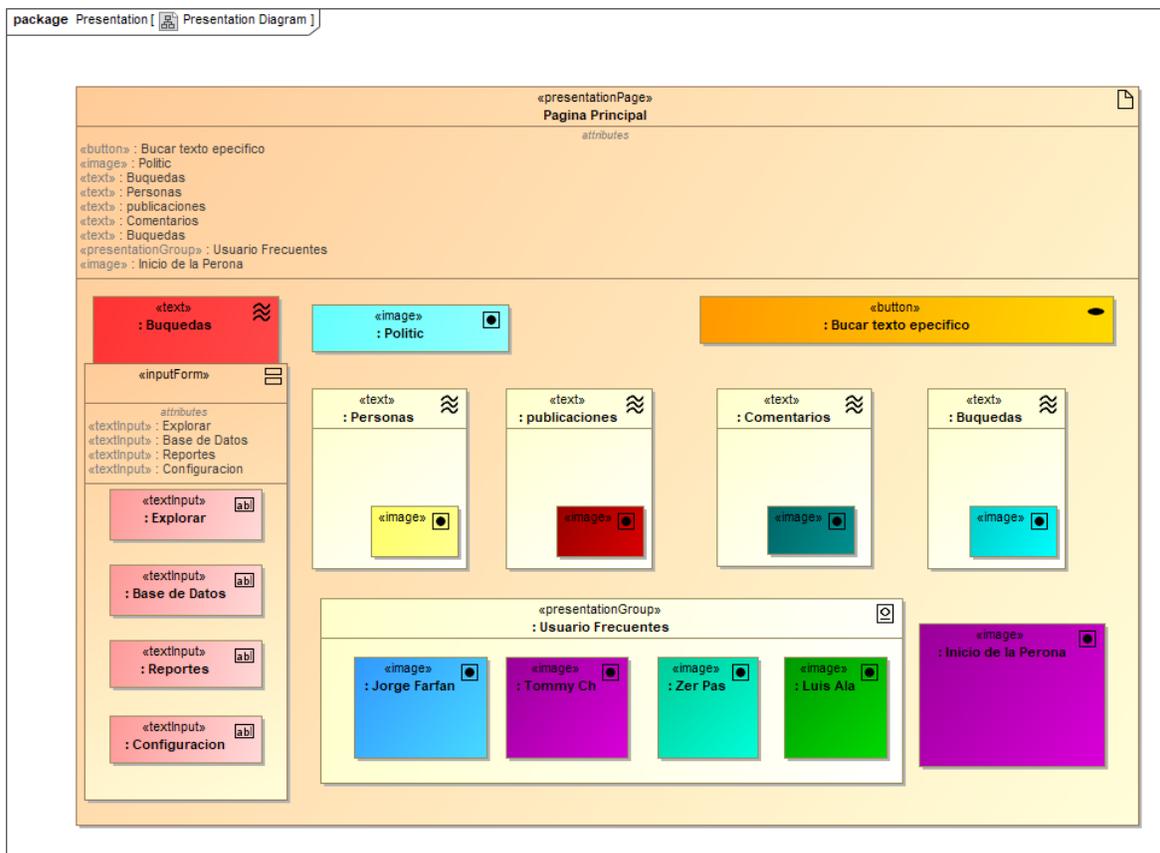


Figura 3. 13 Página Principal

Fuente: Elaboración Propia

### **3.4. MODELO CRISP-DM**

La metodología CRISP-DM permitirá realizar la preparación de los datos, así como el modelado y posteriormente podremos realizar un análisis de los resultados que sirve para la verificación de la hipótesis planteada

#### **3.4.1. COMPRENSIÓN DEL NEGOCIO**

##### **3.4.1.1. Contexto**

De todos es sabido que tener presencia en las redes sociales es fundamental, los políticos intensifican su movimiento en ellas o se abren un perfil, donde el debate político es más intenso. Facebook se convierten en un camino directo de comunicación con las personas donde los candidatos políticos se lanzan a la actualización masiva de sus perfiles. Pero, deben tener presente que las redes sociales son un canal para el debate, para la conversación, y que tendrán que escuchar, responder y analizar las voces opuestas que puedan surgir los políticos.

##### **3.4.1.2. Objetivos de Negocio**

Los políticos tienen como principal objetivo el tratar de colocar más perfiles a favor de su partido para atacar a sus opositores. En tal sentido esta investigación servirá en el apoyo de su partido que usuario comenta mal de su partido.

##### **3.4.1.3. Inventario de Recursos**

Para un mejor trabajo en la investigación se necesitarán los siguientes recursos:

- Computacionales Equipo de computación. Cuyas capacidades mínimas deberán de ser 4 GB de memoria Ram, Procesador Core i3 mínimo, sistema operativo Windows 7 x64

### **3.4.2. COMPRESIÓN DE LOS DATOS**

#### **3.4.2.1. Recolección de datos iniciales**

La información que se utilizara son los siguientes:

- Información de Usuario: Nombre, Apellidos, Correo Electrónico y contraseña. Que sirve como datos base para la otorgación el ingreso del sistema.
- Información sobre el comportamiento sistema: Estado de la Publicación investigado servirán para tomar como base para la extracción de comentarios.

#### **3.4.2.2. Descripción de los datos**

Esos lazos de conexión se representan en las redes sociales donde es posible crear una identidad propia partiendo de sus publicaciones, sus fotos, sus mensajes o videos, y entretejer una red social donde se comparte, genera y se difunden distintos mensajes. Los partidos políticos que comentan de la publicación en redes sociales filtran y modifican los mensajes políticos antes de que ingresen al cerebro de interesado y constituyen un complemento importante para la estrategia de un político candidato.

### **3.4.3. PREPARACIÓN DE LOS DATOS**

De acuerdo a la metodología, se han seleccionado los siguientes algoritmos para las diferentes etapas del tratamiento de los datos (Véase figura 3.13) y (Véase figura 3.14).

#### **3.4.3.1. Selección de datos**

Donde se mostrará la selección de los datos en Facebook

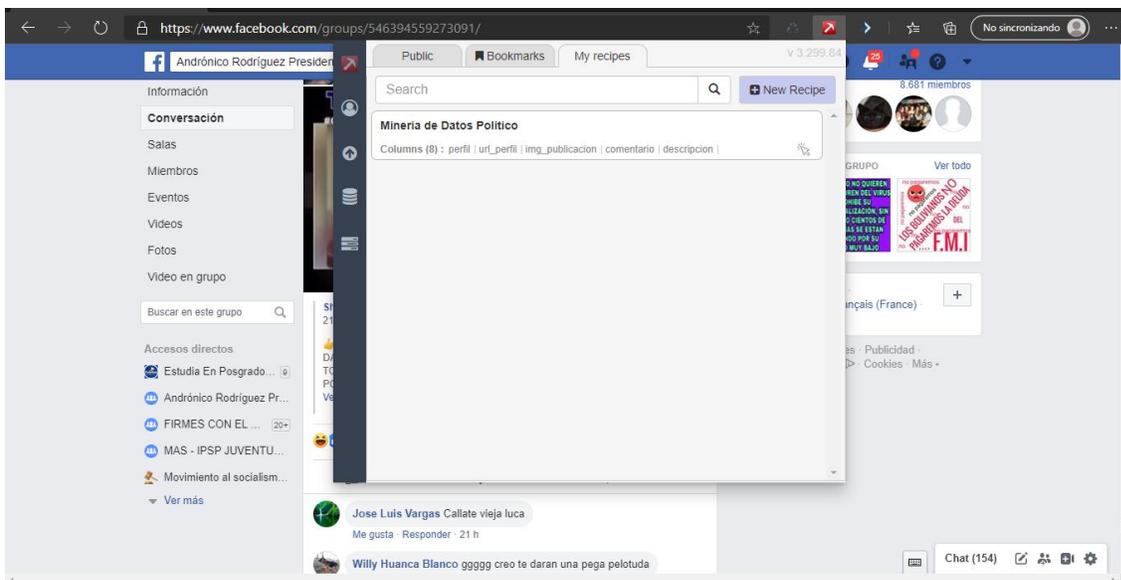


Figura 3. 14 Preparación de los Datos

Fuente: Elaboración Propia

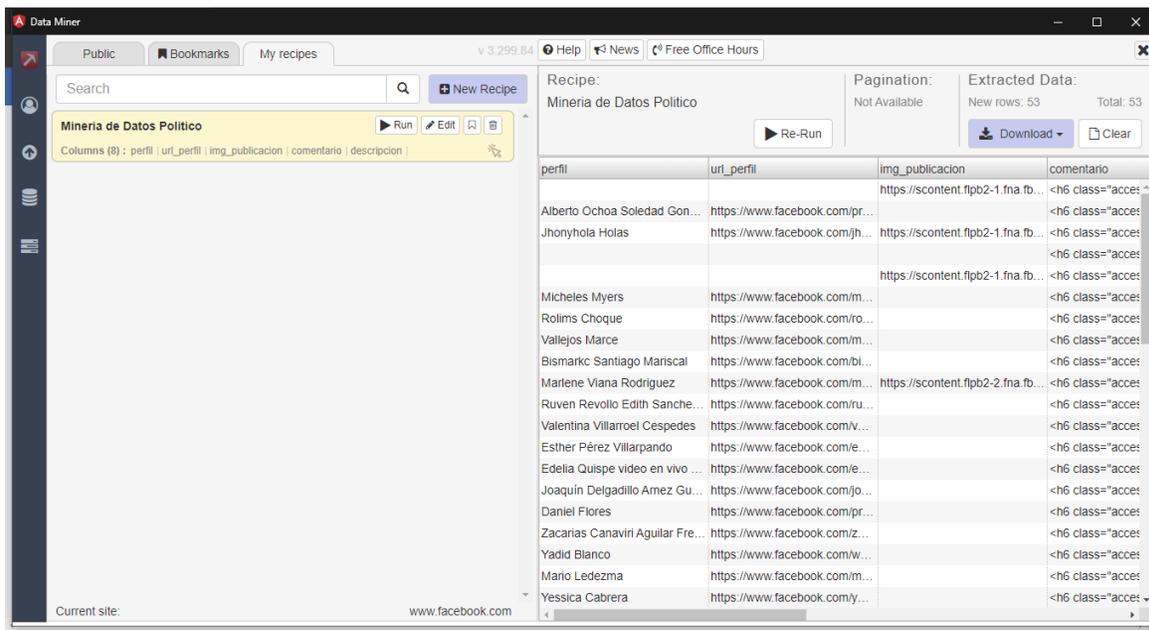


Figura 3. 15 Selección de Datos

Fuente: Elaboración Propia

Los datos son extraídos y preparados para su procesamiento en RapidMiner, en primera instancia se carga un archivo Excel y posteriormente se carga a la base de datos (ver figura 3.15), (ver figura 3.16) y (ver figura 3.17).

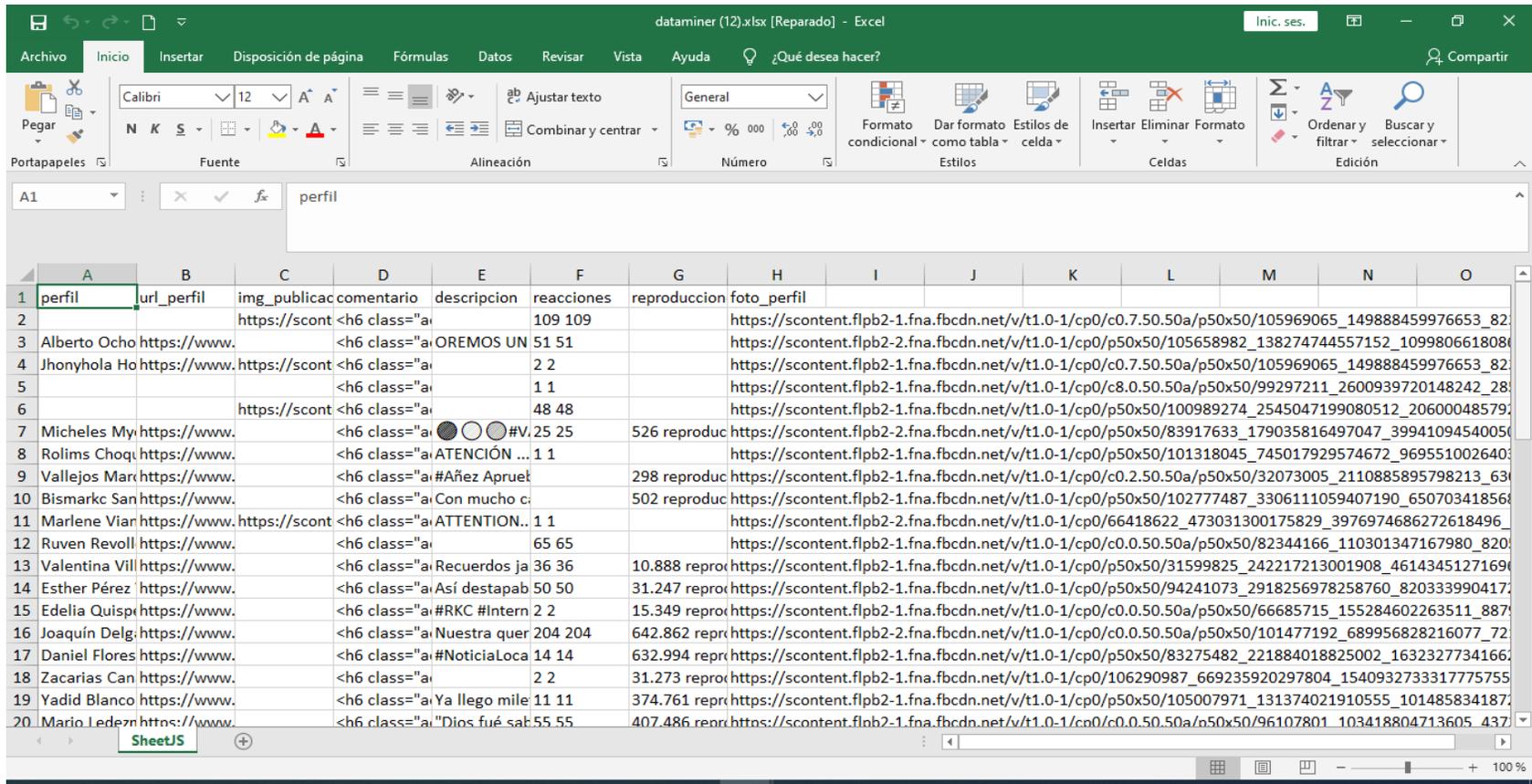


Figura 3. 16 Archivo Excel

Fuente: Elaboración Propia

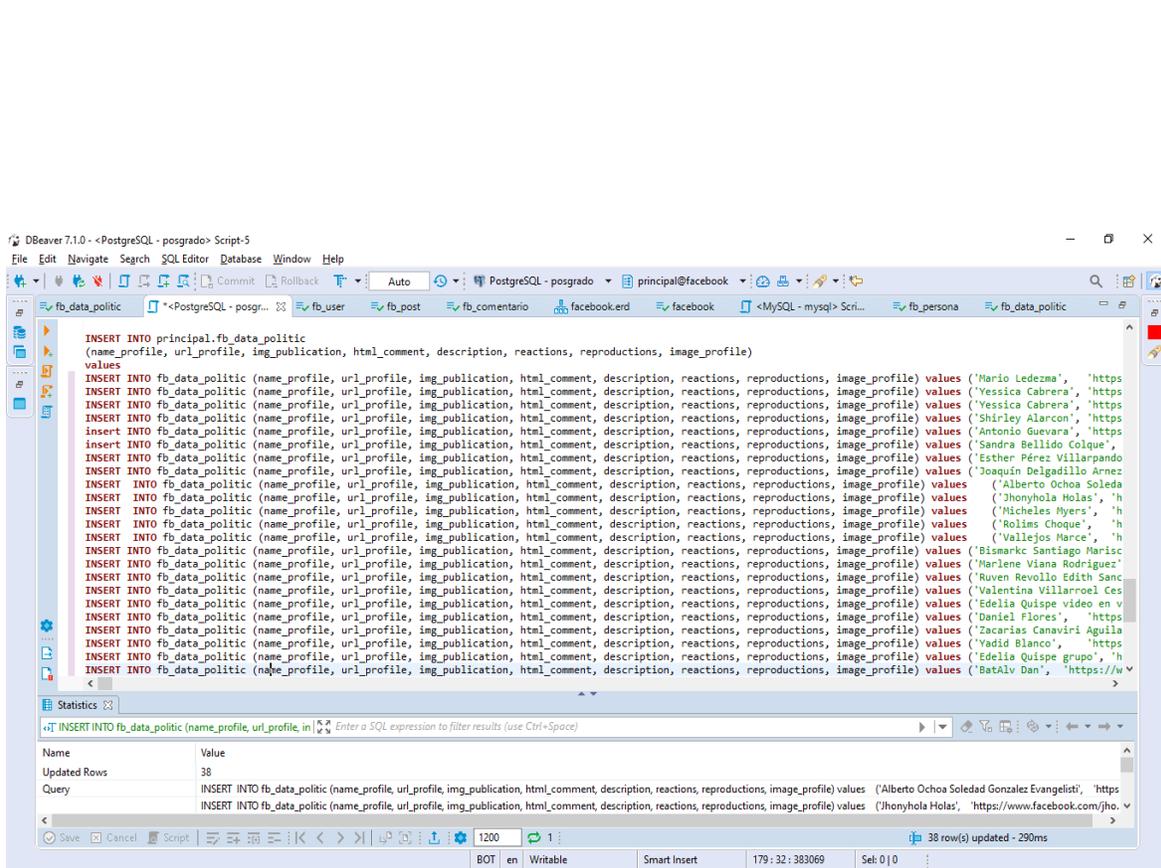


Figura 3. 17 Insertando Registro a la Base de Datos

Fuente: Elaboración Propia

ExampleSet (/Local Repository/comments) x WordList (Process Documents from Data) x

Filter (410 / 410 examples): all

Row No.	id_comenta...	comentario	id_post	id_facebook	created_at	updated_at
1	1108	Siga adelante porq el goberno defacto quiere conegt...	8190	sule.coro	Jun 22, 2020 ...	Jun 22, 2020 ...
2	1109	Eso dale Eva	8190	amirfabio.con...	Jun 22, 2020 ...	Jun 22, 2020 ...
3	1110	A INVESTIGAR...@#@#@#@	8198	liliana.arce.3...	Jun 22, 2020 ...	Jun 22, 2020 ...
4	1111	Grande EVO grande...	8199	jorge.farfan.5...	Jun 22, 2020 ...	Jun 22, 2020 ...
5	1112	Fuera añez	8201	ramyenybabi...	Jun 22, 2020 ...	Jun 22, 2020 ...
6	1113	Fueza compañeros	8201	juvenal.mallku	Jun 22, 2020 ...	Jun 22, 2020 ...
7	1114	Vendidos el pueblo no olvida	8201	1000499686...	Jun 22, 2020 ...	Jun 22, 2020 ...
8	1115	Cob vendidos	8201	1000499686...	Jun 22, 2020 ...	Jun 22, 2020 ...
9	1116	Fuera añez	8201	pitejunior.sa...	Jun 22, 2020 ...	Jun 22, 2020 ...
10	1117	Fuera añez el alto se respeta	8201	yhosele.cand...	Jun 22, 2020 ...	Jun 22, 2020 ...
11	1118	Arce siempre	8206	julian.mama...	Jun 22, 2020 ...	Jun 22, 2020 ...
12	1119	Por Arce presidente obio	8207	1000506398...	Jun 22, 2020 ...	Jun 22, 2020 ...
13	1120	Arce	8207	marianita.cha...	Jun 22, 2020 ...	Jun 22, 2020 ...
14	1121	Arce	8207	1000130385...	Jun 22, 2020 ...	Jun 22, 2020 ...

ExampleSet (410 examples, 0 special attributes, 7 regular attributes)

*Figura 3. 18 Cargado de los datos en RapidMiner*

*Fuente: Elaboración Propia*

Una Se tienen los datos cargados RapidMiner se consideran dos algoritmos básicos acorde al tipo de información obtenida en la base de datos (ver figura 3.18) los cuales son:

- Normalización: RapidMiner realiza la normalización de todas las variables cargadas.

The screenshot displays the RapidMiner interface with a data table. The table has the following columns: Row No., id\_comenta..., id\_post, comentario, id\_face..., created\_at, and updated\_at. The data rows show various comment IDs and their corresponding timestamps.

Row No.	id_comenta...	id_post	comentario	id_face...	created_at	updated_at
381	1.481	1.645	Quien pues esta en el gobierno , acaso sigue Evo de presid...	1000004599...	Jun 23, 2020 ...	Jun 23, 2020 ...
68	-1.160	-1.013	No no a salir a protestar todo bolivia	1000044828...	Jun 22, 2020 ...	Jun 22, 2020 ...
46	-1.346	-1.294	Es editado, no creo q ese criminal que tiene tanto odio a Ev...	1000046373...	Jun 22, 2020 ...	Jun 22, 2020 ...
61	-1.219	-1.148	Es editado, no creo q ese criminal que tiene tanto odio a Ev...	1000046373...	Jun 22, 2020 ...	Jun 22, 2020 ...
69	-1.152	-0.997	La COB siempre estuvo a lado del pueblo y ahí claramente ...	1000046373...	Jun 22, 2020 ...	Jun 22, 2020 ...
384	1.506	1.739	Y ESTE PAYASO VIEJO HECHO AL CHIBOLO O PEÑA...JAJ...	1000049824...	Jun 23, 2020 ...	Jun 23, 2020 ...
197	-0.072	-0.193	Amén	1000088175...	Jun 22, 2020 ...	Jun 22, 2020 ...
280	0.629	0.402	Que vieja condenada	1000092131...	Jun 22, 2020 ...	Jun 22, 2020 ...
342	1.152	1.180	Vieja maleante	1000092131...	Jun 22, 2020 ...	Jun 22, 2020 ...
354	1.253	1.258	Para que sirve ese maleante carajo que son simberguenza	1000095678...	Jun 22, 2020 ...	Jun 22, 2020 ...
155	-0.426	-0.464	Murillo y añez son los delincuentes, con este imagen de ev...	1000105024...	Jun 22, 2020 ...	Jun 22, 2020 ...
153	-0.443	-0.475	Arce presidente	1000115907...	Jun 22, 2020 ...	Jun 22, 2020 ...
119	-0.738	-0.652	Fuerza todos juntos por una verdadera democracia	1000117750...	Jun 22, 2020 ...	Jun 22, 2020 ...
229	0.198	0.021	Elecciones	1000117750...	Jun 22, 2020 ...	Jun 22, 2020 ...

Figura 3. 19 Normalización en RapidMiner

Fuente: Elaboración propia

### 3.5. DESARROLLO E IMPLEMENTACIÓN

#### 3.5.1. Diseño de vistas del sistema

- Inicio

Preparamos la vista principal del sistema **Politic Data Mining** consta de dos entradas de texto y dos botones uno para inicio de sesión vía Facebook y el otro inicio de sesión normal.



Figura 3. 20 Diseño Login

Fuente: Elaboración propia

La vista que se nos presenta una vez que tengamos una sesión activa dentro del sistema de **Politic Data Mining**, consta con detalles para poder ver cantidad de personas registradas, publicaciones, comentarios, búsquedas además cuenta con una galería de personas con más comentarios.

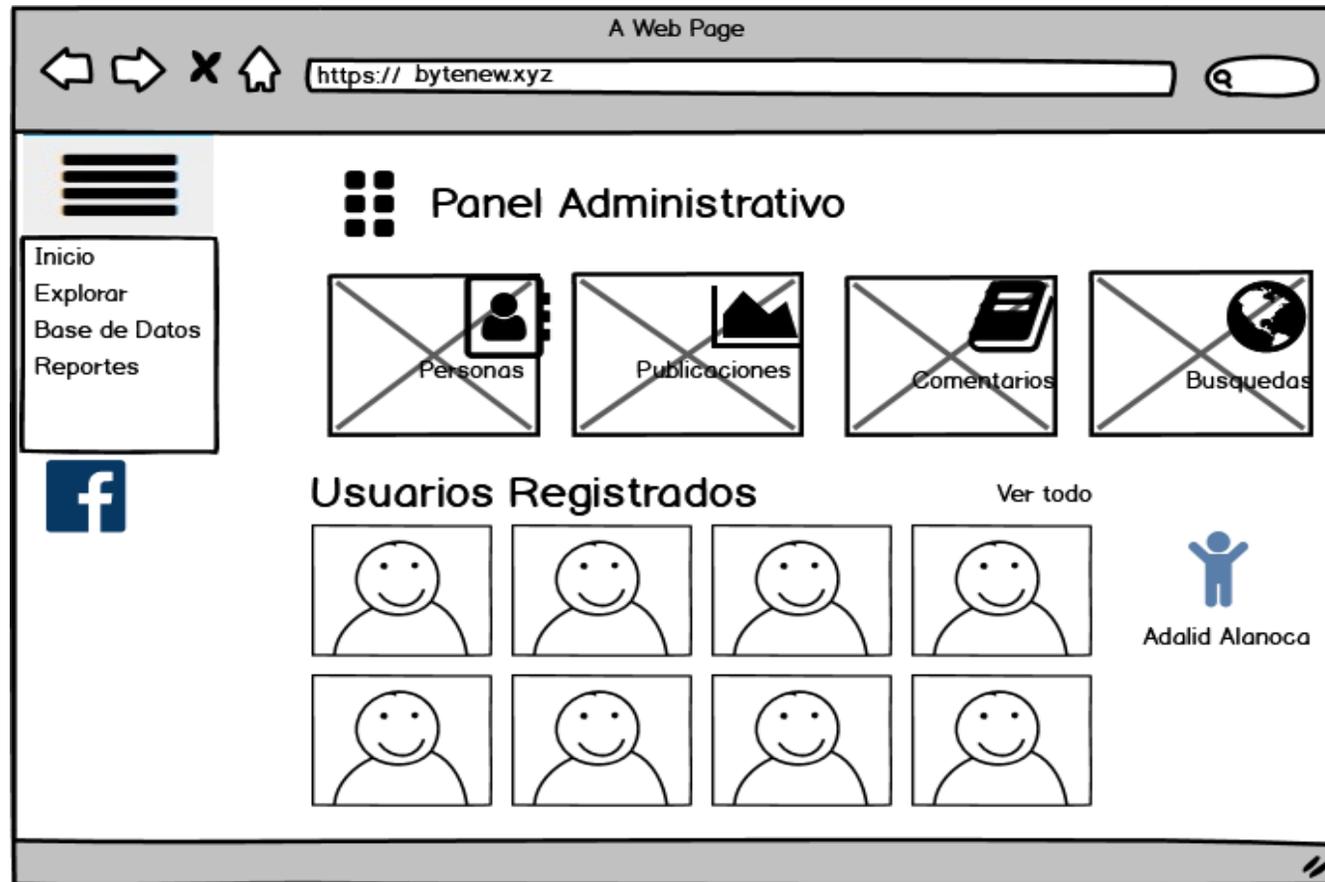


Figura 3. 21 Diseño de Pagina Principal

Fuente: Elaboración propia

- **Explorar**

La vista más importante del sistema la cual se encargará de hacer el proceso de minería de texto, detalla un campo de texto y una lista donde se mostrarán los patrones encontrados.

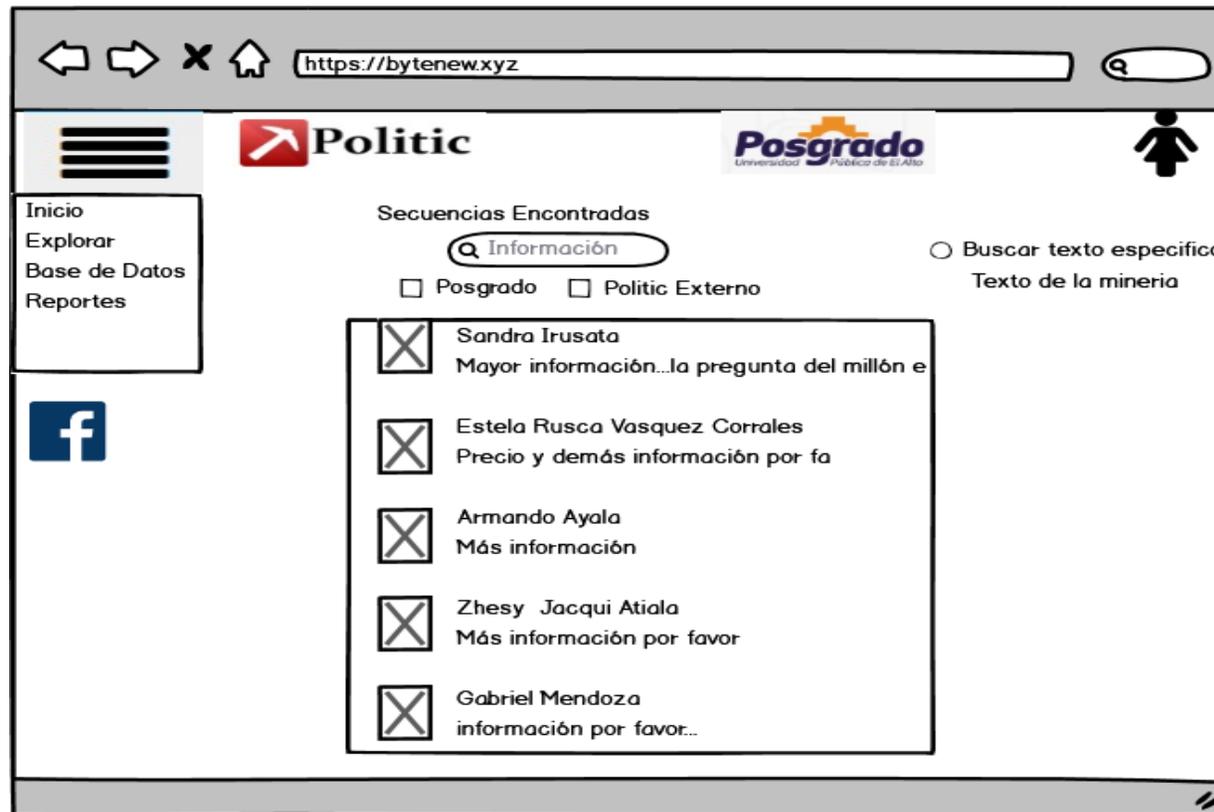


Figura 3. 22 Diseño de Explorar Comentarios

Fuente: Elaboración propia

- **Base de Datos**

Vista que tendrá la función de recargar la base de datos.

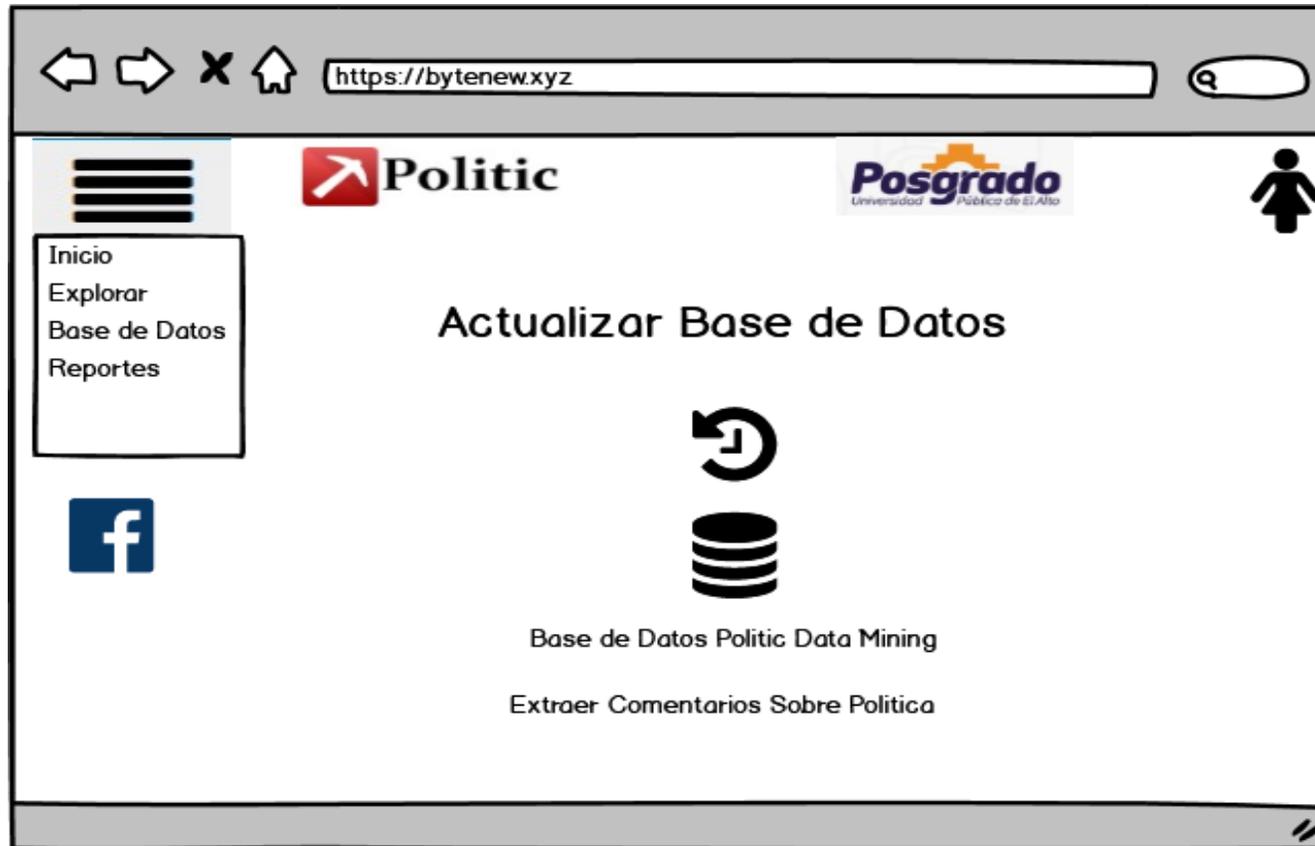


Figura 3. 23 Diseño de Cargar Base de Datos

Fuente: Elaboración propia

## Reportes

Vista que nos detallara el ranking mostrando a perfiles que más comentaron en la página de Posgrado, también podremos generarlo en PDF para poder descargarlo.

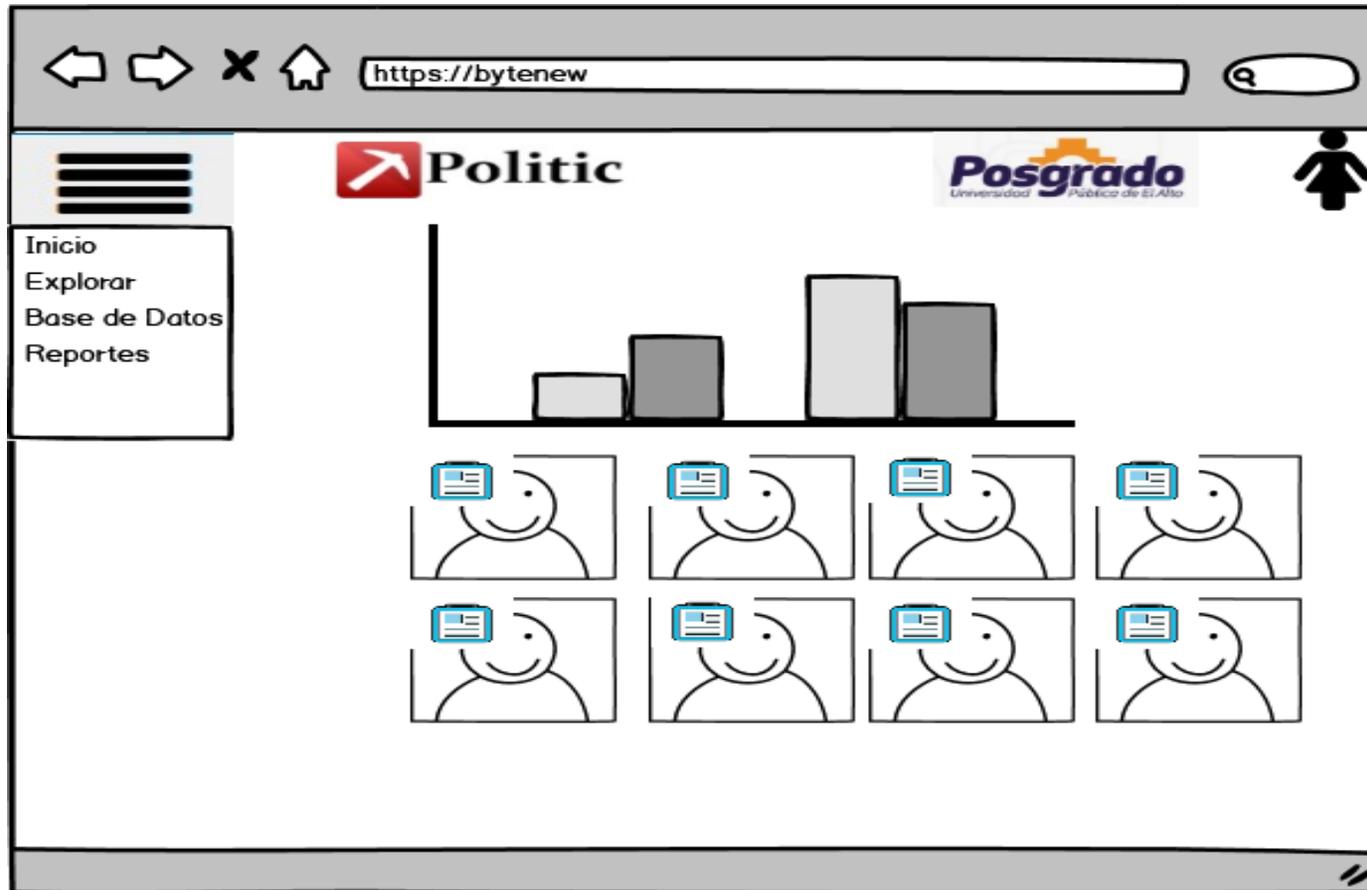


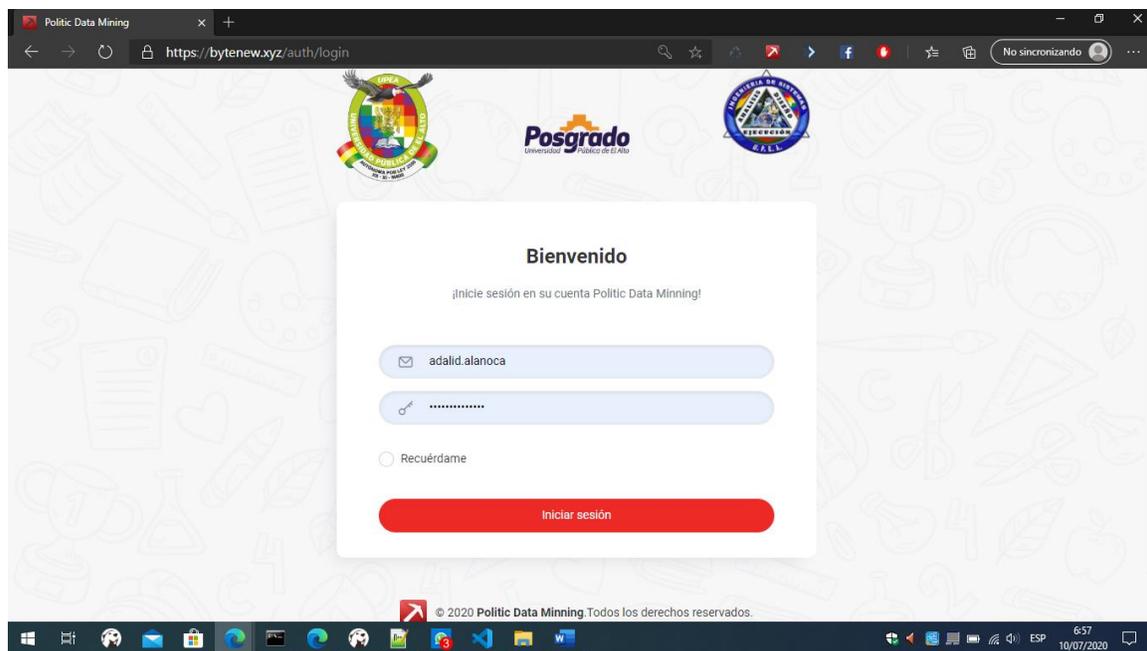
Figura 3. 24 Diseño de Reportes o Información

Fuente: Elaboración propia

### 3.5.2. Implementación de las vistas del sistema

- **Inicio**

En primera instancia se diseña el inicio de sesión del modelo que consta de dos entradas de texto uno para el usuario y el siguiente para la contraseña y un botón para el inicio de sesión.



*Figura 3. 25 Implementación de Login*

*Fuente: Elaboración propia*

Mostramos el código de lado del servidor encargado de autenticar si el usuario y la contraseña son correctas.

```
#authenticate = autenticar al usuario
public function authenticate()
{
    #Recibimos parametros username y password
    $username = trim($this->request->getPost('username'));
    $password = $this->request->getPost('password');
    #Buscamos en la base de datos los 2 datos que nos mando el Login
    $userSearched = $this->authModel->where(array('username' => $username, 'pass' => md5($password)))->findAll();

    #Contamos si $userSearched es igual a 1 si lo es entendemos que podemos aprobar el inicio de sesion
    if (count($userSearched) == 1) {
        # Agregamos una sesion al navegador
        $this->session->set(array('id_user' => $userSearched[0]['id_user']));
        # Redireccionamos a la pagina principal
        return redirect()->to(base_url('/'));
    }
    #Si $userSearched no es igual a 1 debemos devolverlo al mismo login
    else {
        $this->session->destroy();
        return redirect()->to(base_url('/auth/login'));
    }
}
```

Figura 3. 26 Fragmento de Código Login

Fuente: Elaboración propia

Creamos la vista de panel principal del usuario que muestra los datos como ser número total de personas, publicaciones, comentarios. En la parte derecha mostramos un menú en el cual se seleccionarán cada pestaña.

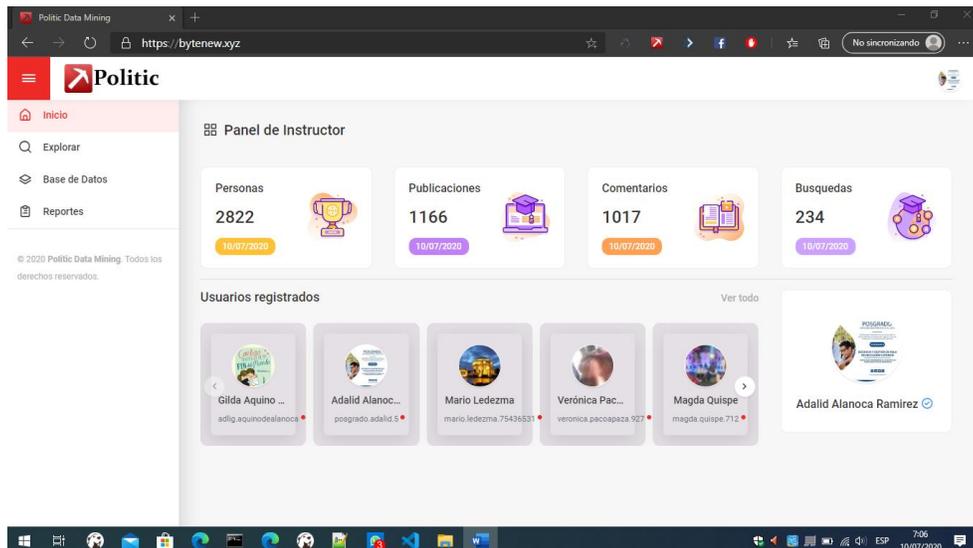


Figura 3. 27 Implementación de Página Principal

Fuente: Elaboración propia

Detallamos el código que da la funcionalidad al panel principal, mostraremos solo el código que da funcionalidad al panel principal dejando de lado los modelos y las vistas.

```
public function index()  
{  
    $this->data['personas'] = $this->persona->findAll(10, 0);  
    $this->data['number_people'] = count($this->persona->findAll());  
    $this->data['number_posts'] = count($this->post->findAll());  
    $this->data['number_comments'] = count($this->comentario->findAll());  
    You, 5 days ago * null  
    return $this->templater->view('main', $this->data);  
}
```

Figura 3. 28 Fragmento de Código Pagina principal

Fuente: Elaboración propia

Se diseño un apartado para visualizar todos los perfiles que cuentan con más comentarios, también

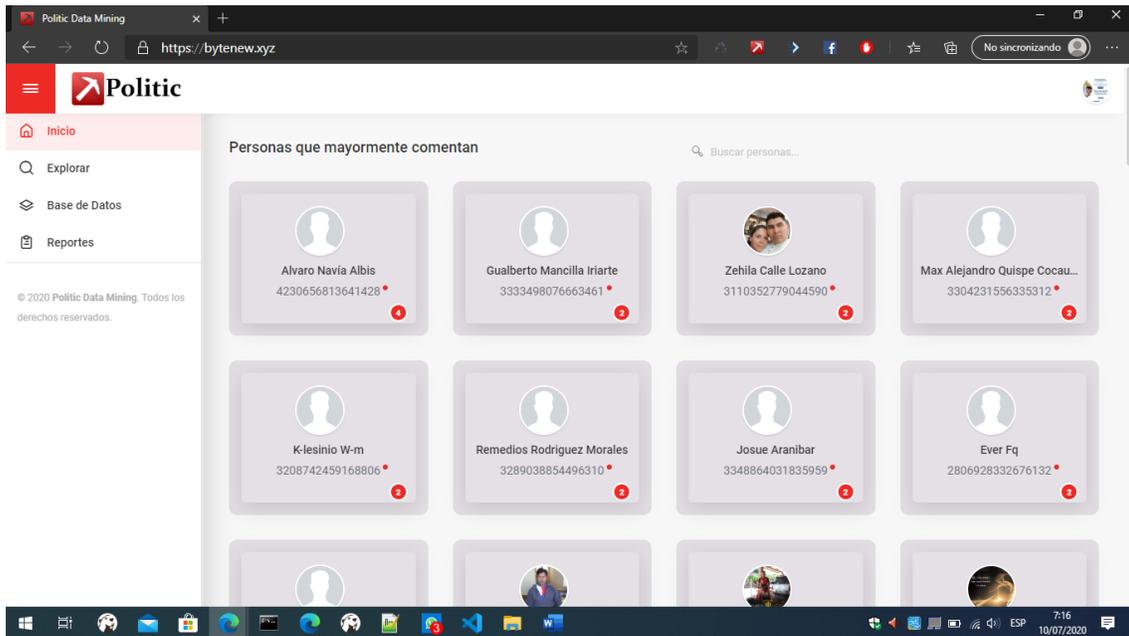


Figura 3. 29 Diseño de Visualizaciones Perfil

Fuente: Elaboración propia

Detallamos el código que hace posible esta selección y la contabilización de las personas, publicaciones, etc.

```

public function index()
{
    $this->data['personas'] = $this->persona->findAll(10, 0);
    $this->data['number_people'] = count($this->persona->findAll());
    $this->data['number_posts'] = count($this->post->findAll());
    $this->data['number_comments'] = count($this->comentario->findAll());

    return $this->templater->view('main', $this->data);
}

public function listPeople()
{
    $this->request->getPost('buscar_personas') == null
    ? $this->data['personas'] = $this->querys->obtenerCantidadComentarioPersonaPosgrado(60)
    : $this->data['personas'] = $this->querys->persona('search', null, null, strtolower(trim($this->request->getPost('bu
    return $this->templater->view('explore/profiles', $this->data);
}
    
```

Figura 3. 30 Fragmento de Código Explorar Comentario

Fuente: Elaboración propia

- **Explorar**

Se diseño con la vista para visualizar la minería de datos y una entrada de texto en el cual se debe escribir el patrón de texto que desea encontrar.

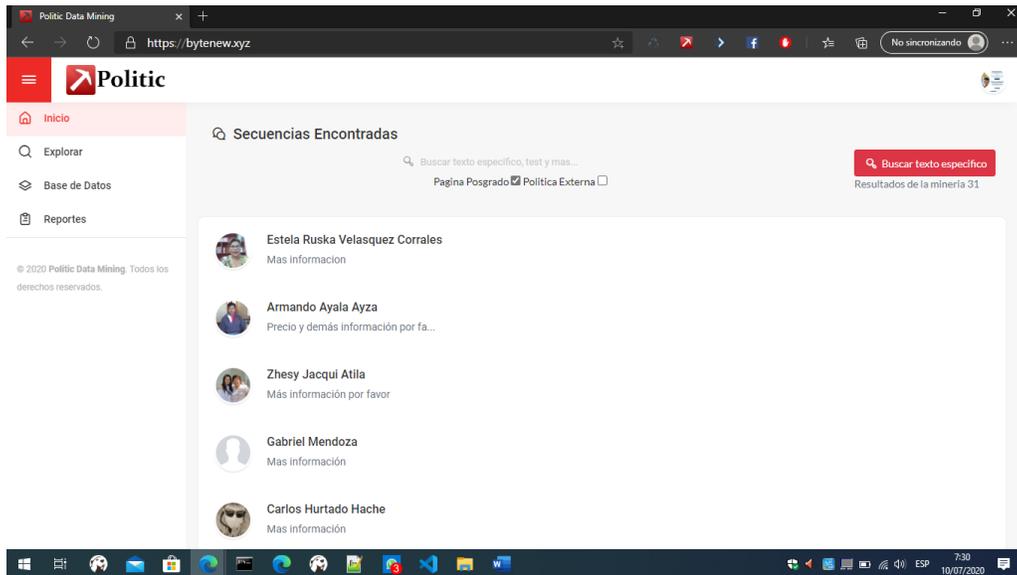


Figura 3. 31 Resultados de la Minería de Texto

Fuente: Elaboración propia

El código necesario en lado del servidor para hacer la minería de datos.

```

public function getContentFilters()
{
    if (empty($this->contentFilters)) {

        $lambdaFunc = function ($word) {
            return preg_replace('/[\x00-\x1F\x80-\xFF]/u', ' ', $word);
        };

        $this->contentFilters = [
            new Filters\StripTagsFilter(),
            new Filters\LowerCaseFilter(),
            new Filters\NumbersFilter(),
            new Filters\EmailFilter(),
            new Filters\UriFilter(),
            new Filters\PossessiveNounFilter(),
            new Filters\QuotesFilter(),
            new Filters\PunctuationFilter(),
            new Filters\CharFilter(),
            new Filters\LambdaFilter($lambdaFunc),
            new Filters\WhitespaceFilter()
        ];
    }
    return $this->contentFilters;
}

```

Figura 3. 32 Fragmento de Código Usuarios que Comentan

Fuente: Elaboración propia

- **Base de Datos**

En esta vista se visualiza la carga de base de datos desde la API de Facebook posteriormente se ingresará los datos a nuestra base de datos.

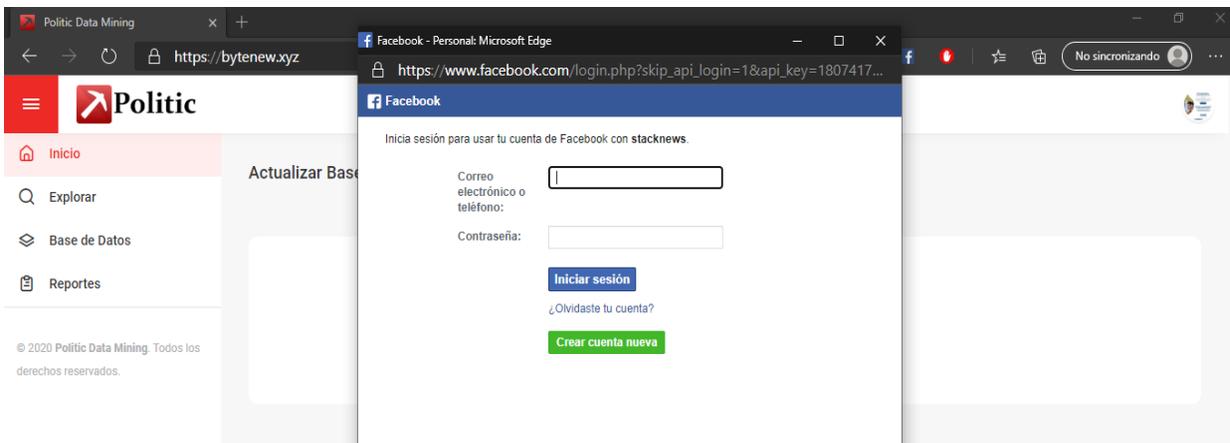


Figura 3. 33 Login API de Facebook

Fuente: Elaboración propia

El código necesario para conectarse a la API de Facebook.

```
function testAPIPage() {
  $.ajax({
    url: 'https://graph.facebook.com/v7.0/me?fields=posts.limit(10)%7Bfrom%2Cshares%2Ccomments%7Bmessage%2Cfrom%2Cid%7D',
    method: 'GET',
  }).done(function (response) {
    $.ajax({
      type: 'post',
      url: 'data/rechargeDataBasePosgrado',
      data: {
        respuesta: JSON.stringify(response),
      },
      dataType: 'json',
    }).done(function (response) {
      $.ajax({
        url: '/data/rechargeDataBasePolitic',
      })
      .done(function (data) {
        if (typeof data.success !== 'undefined') {
          simpleAlert('CORRECTO', data.success, 'top-right', 'success', 6000);
        } else {
          simpleAlert('ERROR', data.error, 'top-right', 'error', 6000);
        }
      })
      .fail(function (jqXHR, textStatus) {
        simpleAlert(jqXHR.statusText, jqXHR.status, 'top-right', 'error', 3000);
        console.log(jqXHR.responseText);
      });
    });
  });
}
function testAPIPage() {
```

Figura 3. 34 Fragmento de Código de Conectar con Api de Facebook

Fuente: Elaboración propia

- **Reportes**

Nos muestra un ranking de los perfiles con más interacción en la página.

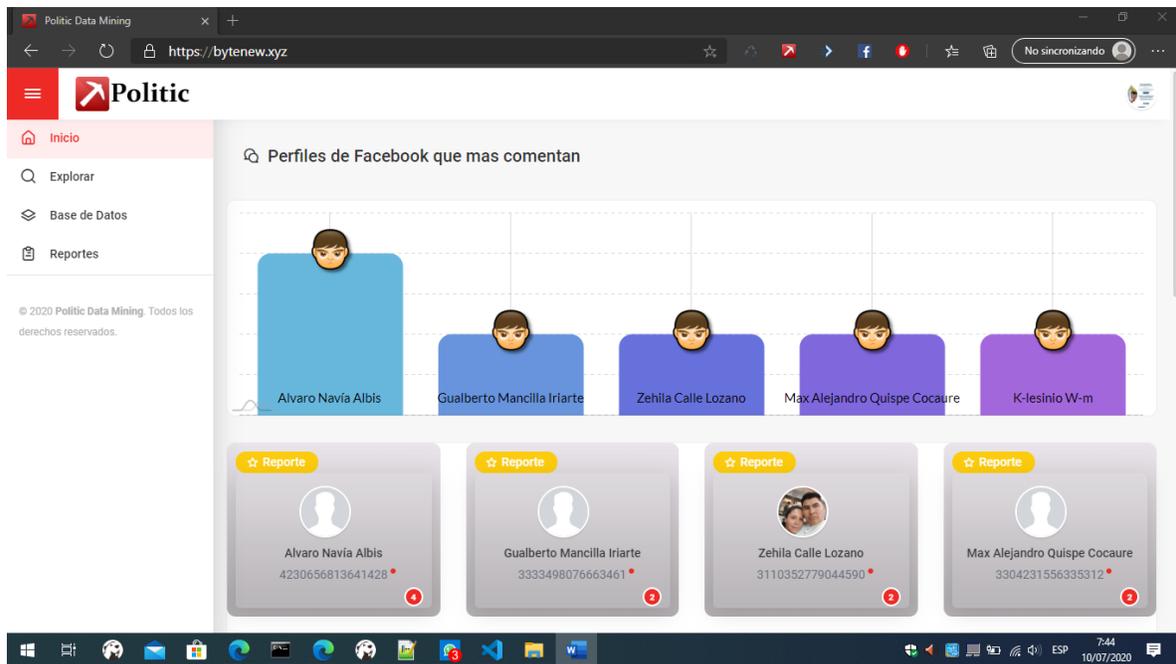


Figura 3. 35 Ranking de Usuarios Frecuentes

Fuente: Elaboración propia

El código en lado del servidor que hace posible estas visualizaciones.

```

public function index()
{
    $this->data['personas'] = $this->querys->obtenerCantidadComentarioPersonaPosgrado(20, 'desc');
    return $this->templatere->view('report/report', $this->data);
}
public function ajaxCantidadComentarioPersona()
{
    if ($this->request->isAJAX()) {
        $cantidadComentarios = $this->querys->obtenerCantidadComentarioPersonaPosgrado(5, 'desc');
        if (is_array($cantidadComentarios)) {
            return $this->response->setJSON(array('exito' => true, 'cantidad_comentarios' => $cantidadComentarios));
        } else {
            return $this->response->setJSON(json_encode(array('error' => $cantidadComentarios)));
        }
    }
}

```

Figura 3. 36 Fragmento de Código Visualizaciones

Fuente: Elaboración propia

3.5.3. Implementación de la Base de Datos

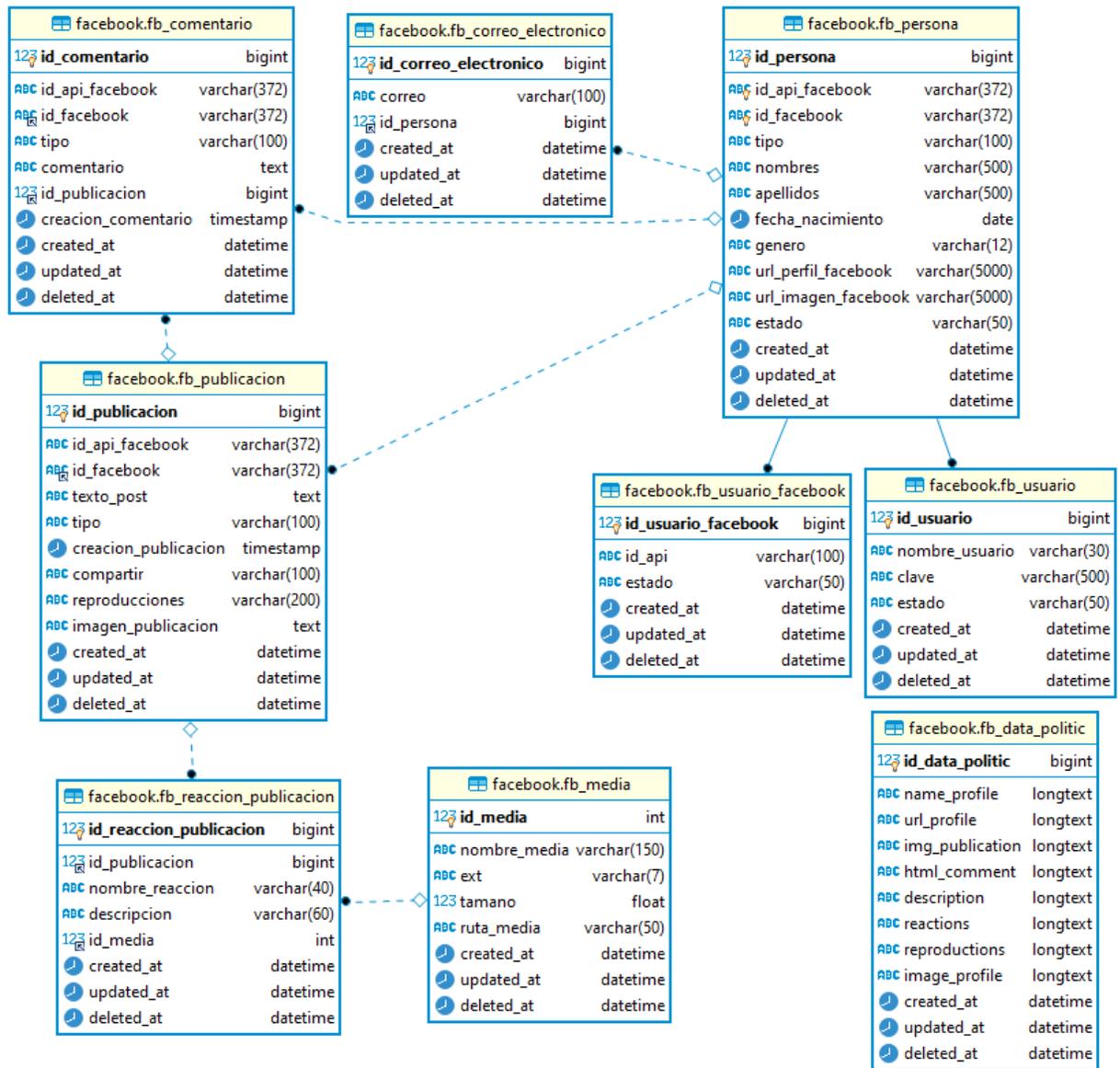


Figura 3. 37 Diagrama Entidad Relación

Fuente: Elaboración propia

### 3.5.3.1. Diccionario de datos

Tabla 3. 7

Tabla *fb\_usuario*

Nombre de Tabla <b>fb_usuario</b>				
Descripción	Contiene todos los registros de los usuarios que acceden al sistema			
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
<b>id_usuario</b>	bigint	PK		Llave primaria de la tabla
<b>nombre_usuario</b>	varchar		30	Nombre de usuario
<b>clave</b>	varchar		150	Contraseña de usuario
<b>estado</b>	Varchar		50	Estado del usuario

Fuente: Elaboración propia

Tabla 3. 8

Tabla *fb\_persona*

Nombre de Tabla <b>fb_persona</b>				
Descripción	Contiene todos los registros de perfiles de Facebook			
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
<b>id_persona</b>	bigint	PK		Llave primaria de la tabla
<b>id_api_facebook</b>	varchar		372	Identificador API Facebook
<b>tipo</b>	varchar		100	Tipo de persona
<b>nombres</b>	varchar		500	Nombre del perfil
<b>apellidos</b>	varchar		500	Apellido del perfil
<b>fecha_nacimiento</b>	datetime			Fecha de nacimiento
<b>genero</b>	varchar		12	Genero de perfil
<b>id_facebook</b>	varchar		372	Identificador Id Facebook
<b>url_perfil_facebook</b>	longtext			Link de Facebook
<b>url_imagen_perfil_facebook</b>	varchar		5000	Link de imagen de perfil Facebook

Fuente: Elaboración propia

Tabla 3. 9

*Tabla fb\_grupo\_usuario*

Nombre de Tabla fb_grupo_usuario				
Descripción	Contiene todos los registros del nivel de acceso del usuario			
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
id_grupo_usuario	bigint	PK		Llave primaria de la tabla
id_grupo	int			Referencia a tabla fb_grupo
id_usuario	bigint			Referencia a tabla fb_usuario
fecha_inicio	datetime			Fecha inicio del usuario
fecha_fin	datetime			Fecha fin del usuario
estado	varchar		50	Estado grupo usuario

Fuente: Elaboración propia

Tabla 3. 10

*Tabla fb\_grupo*

Nombre de Tabla fb_grupo				
Descripción	Contiene todos los registros de los accesos			
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
id_grupo	bigint	PK		Llave primaria de la tabla
nombre	varchar		20	Nombre de grupo
descripcion	varchar		500	Descripción de grupo
estado	varchar		50	Estado del grupo

Fuente: Elaboración propia

Tabla 3. 11

*Tabla fb\_publicacion*

Nombre de Tabla fb_publicacion				
Descripción	Contiene todos los registros de la publicación			
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
id_publicacion	bigint	PK		Identificador de la publicación
id_api_facebook	varchar		1000	Identificador API Facebook
texto_post	longtext			Descripción de la publicación
tipo	varchar		100	Tipo de publicación
compartir	varchar		100	Cantidad de compartidos
reproducciones	varchar		200	Cantidad de reproducciones
imagen_post	longtext			Imagen de la publicación
id_facebook	varchar	FK	372	Referencia a tabla fb_persona

Fuente: Elaboración propia

Tabla 3. 12  
 Tabla *fb\_comentario*

Nombre de Tabla <b>fb_comentario</b>				
Descripción	Contiene todos los registros de los comentarios			
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
<b>id_comentario</b>	bigint	PK		Llave primaria de la tabla
<b>id_api_facebook</b>	varchar		1000	Identificador API Facebook
<b>tipo</b>	varchar		100	Tipo de publicación
<b>comentario</b>	longtext			Comentario
<b>id_publicacion</b>	varchar	FK		Referencia a tabla publicacion
<b>cantidad_like</b>	varchar		10	Cantidad de me gustas
<b>id_facebook</b>	varchar	FK	372	Referencia a tabla fb_persona

Fuente: Elaboración propia

Tabla 3. 13  
 Vista *fb\_view\_usuarios*

Nombre de Vista <b>fb_view_usuarios</b>				
Descripción	Vista, unión de 5 tablas			
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
<b>id_persona</b>	bigint			
<b>nombres</b>	varchar		500	
<b>apellidos</b>	varchar		500	
<b>genero</b>	varchar		12	
<b>id_correo_electronico</b>	int			
<b>id_facebook</b>	varchar		372	
<b>url_perfil_facebook</b>	longtext			
<b>url_imagen_facebook</b>	varchar		5000	
<b>nombre_usuario</b>	varchar		30	
<b>clave</b>	varchar		150	
<b>id_usuario</b>	bigint			

Fuente: Elaboración propia

Tabla 3. 14

*Tabla fb\_correo\_electronico*

Nombre de Tabla		fb_correo_electronico		
Descripción				
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
id_correo_electronico	bigint	PK		Llave primaria de la tabla
correo	varchar		100	Correo
id_persona	int	FK		Referencia a tabla fb_persona

Fuente: Elaboración propia

Tabla 3. 15

*Tabla fb\_media*

Nombre de Tabla		fb_media		
Descripción				
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
id_media	int	PK		Llave primaria de la tabla
nombre_media	varchar		150	Nombre del archivo
ext	varchar		7	Extensión del archivo
tamano	float			Tamaño del archivo
ruta_media	varchar		50	Ruta del archivo

Fuente: Elaboración propia

Tabla 3. 16

*Tabla fb\_reaccion\_publicacion*

Nombre de Tabla		fb_reaccion_publicacion		
Descripción				
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
id_reaccion_publicacion	int	PK		Llave primaria de la tabla
nombre_reaccion	varchar		40	Nombre de la reacción
descripcion	varchar		60	Descripción de la reacción
id_media	float	FK		Referencia a tabla fb_media

Fuente: Elaboración propia

Tabla 3. 17  
 Tabla *fb\_data\_politic*

Nombre de Tabla <i>fb_data_politic</i>				
Descripción				
Nombre de Campo	Tipo de datos	Integridad	Longitud	Descripción
<b>id_data_politic</b>	bigint	PK		Llave primaria de la tabla
<b>nombre_profile</b>	longtext			Nombre Perfil
<b>url_profile</b>	longtext			Url Perfil
<b>img_publication</b>	longtext			Imagen Perfil
<b>html_comment</b>	longtext			Comentarios en html
<b>description</b>	longtext			Descripción de publicación
<b>reactions</b>	longtext			Reacciones de publicación
<b>reproductions</b>	longtext			Reproducciones
<b>image_profile</b>	longtext			Imagen perfil

Fuente: Elaboración propia



# CAPITULO IV

## CALIDAD Y COSTOS DEL SOFTWARE

El objetivo de este capítulo es desarrollar un modelo de estimación de costo y cronogramar del proyecto web, el mismo se adapta tanto a la práctica como desarrollar, se construye una base de datos que permita la calibración continua del modelo, y así incrementar la precisión en la estimación.

### 4.1. MÉTRICAS DE CALIDAD DE SOFTWARE

#### 4.1.1. Usabilidad

Un conjunto de atributos relacionados con el esfuerzo necesario para su uso, y en la valoración individual de tal uso. Se realiza una evaluación del sistema en base a encuestas de un grupo de usuarios.

Tabla 4. 1  
*Usabilidad del sistema*

NRO.	PREGUNTA	RESPUESTAS		PORCENTAJE
		SI	NO	
1	¿El sistema es comprensible?	9	0	90%
2	¿El sistema es agradable a la vista?	8	2	80%

3	¿El sistema hace lo que dice que hace?	8	2	80%
4	¿Las respuestas del sistema son satisfactorias?	8	2	80%
5	¿Es fácil aprender a manejar el sistema?	10	0	100%
6	¿El sistema satisface las necesidades que usted requiere?	9	1	90%
7	¿El acceso al sistema es complicado?	0	10	100%
8	¿Son comprensibles las respuestas del sistema?	9	1	90%
9	¿Son complicadas los procesos que realiza el sistema?	0	10	100%
10	¿El sistema reduce su tiempo de trabajo?	9	1	90%
				0,90%
<b>PROMEDIO</b>				

Fuente: Elaboración propia

**Usabilidad** 90%

De acuerdo a los datos obtenidos en la tabla de usabilidad, se concluye que el sistema tiene una usabilidad del 0.90, es decir del 90%. Que de cada 100 personas que lleguen a usar el sistema 90 personas indican que el sistema es fácil de manejar y comprensible.

### 4.1.2. Funcionalidad

El punto función es una métrica orientada a la función del software y del proceso por el cual se desarrolla. Se centra en la funcionalidad o utilidad del programa, los puntos de función se calculan realizando una serie de actividades, comenzando por determinar los siguientes números:

- Número de entradas de usuario
- Numero de salidas de usuario
- Número de peticiones de Usuario
- Numero de archivos
- Numero de interfaces externas

De acuerdo a lo mencionado es que se tiene los resultados en la tabla 4.2

Tabla 4. 2  
*Entradas para el cálculo de funcionalidad*

<b>Entradas de usuario</b>	<b>5</b>
<b>Salidas de usuario</b>	<b>7</b>
<b>Consultas de usuario</b>	<b>3</b>
<b>Numero de archivos</b>	<b>3</b>
<b>Interfaces externas</b>	<b>1</b>

Fuente: Elaboración propia

Los puntos de función se calculan rellorando la tabla 4.3 con los datos obtenidos

Tabla 4. 3  
Cálculo de punto de función sin ajustar

Parámetros de Medición	Cuenta	Factor de ponderación medio	Totales
Número de entradas de usuario	5	* 3	= 15
Numero de salidas de usuario	7	* 4	= 28
Número de consultas de usuario	3	* 5	= 15
Numero de archivos	3	* 5	= 30
Numero de interfaces externas	1	* 5	= 7
<b><i>Cuenta_total</i></b>			<b>95</b>

Fuente: Elaboración propia

La relación que permite calcular los puntos de función es la siguiente

$$PF = Cuenta\_total * (Grado\_de\_confiabilidad + tasa\_de\_error * \sum Fi)$$

Donde:

- ***PF*** = Medida de funcionalidad.
- ***Cuenta\_total*** = Es la suma de los siguientes datos: Número de entradas, número de salidas, número de peticiones, número de archivos y número de interfaces externas.
- ***Grado\_de\_confabilidad*** = Es la confiabilidad estimada del sistema

- **Tasa\_de\_error** = Probabilidad subjetiva estimada del dominio de la información, este error estimado es de 1%.
- **Fi** = Son valores de ajuste de complejidad que toman los valores de la tabla 4.4 y dan respuesta a las preguntas de la tabla 4.5

Tabla 4. 4  
Valores de ajuste de complejidad

<b>Sin importancia</b>	<b>0</b>
<b>Incidental</b>	<b>1</b>
<b>Moderado</b>	<b>2</b>
<b>Medio</b>	<b>3</b>
<b>Significativo</b>	<b>4</b>
<b>Esencial</b>	<b>5</b>

Fuente: Elaboración propia

Tabla 4. 5  
Ajuste de complejidad de punto función

<b>ESCALA</b>	<b>Sin importancia</b>	<b>Incidental</b>	<b>Moderado</b>	<b>Medio</b>	<b>Significativo</b>	<b>Esencial</b>
<b>Factor</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>¿Requiere el sistema copias de seguridad y recuperación fiables?</b>						✓
<b>¿Se requiere comunicación de datos?</b>						✓
<b>¿Existen funciones de proceso distribuidos?</b>					✓	
<b>¿Es crítico el rendimiento?</b>			✓			

¿Sera ejecutado el sistema en S.O. existente?						✓
¿Requiere el sistema de entrada interactiva?						✓
¿Requiere el sistema de entrada de datos interactivo sobre múltiples ventanas?				✓		
¿Se actualizan los archivos maestros de manera alternativa?						✓
¿Son complejas las entradas, las salidas, los archivos o las peticiones?						✓
¿Es complejo el procesamiento interno?					✓	
¿Se ha diseñado el código para ser reutilizable?						✓
¿Están incluidas en el diseño la conversión y la instalación?						✓
¿Se ha diseñado el sistema para soportar múltiples instalaciones?						✓
¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?						✓
<b>Total (<math>\sum Fi</math>)</b>						<b>63</b>

Fuente: Elaboración propia

Con la obtención de los anteriores datos y considerando un grado de confiabilidad mínimo del 65% es que a continuación calculamos el valor de PF

$$PF = Cuenta\_total * (Grado\_de\_confiabilidad + tasa\_de\_error * \sum Fi)$$

$$PF = 95 * (0.65 + 0.01 * 63)$$

$$PF = 121.6$$

Si consideramos el máximo valor de ajuste de complejidad como  $\sum Fi = 70$ , se tiene:

$$PF = 95 * (0.65 + 0.01 * 70)$$

$$PF = 128.25$$

Entonces si  $\sum Fi$  es considerada como el 100% la relación obtenida entre los puntos será

$$\text{Funcionalidad} = \left( \frac{PF}{PF_{maximo}} \right) = \frac{121.6}{128.25} = 0.9481$$

Por lo tanto, la funcionalidad del sistema es de 94% tomando en cuenta el punto de función máximo.

#### 4.1.3. Portabilidad

La portabilidad es la capacidad que tiene el sistema para ser trasladado de un entorno a otro. Para poder medir la portabilidad del sistema usaremos la siguiente formula que indica el grado de portabilidad que tiene un software.

$$GP = 1 - \frac{ET}{ER}$$

Donde:

$ET$  = Es la medida de los recursos necesarios para llevar el sistema a otro entorno.

$ER$  = Es la medida de los recursos necesarios para crear el sistema en el entorno residente.

Si  $GP > 0$ , la portabilidad es más rentable que el re-desarrollo

Si  $GP = 1$ , la portabilidad es perfecta

Si  $GP < 0$ , el re-desarrollo es más rentable que la portabilidad.

Con esta información requerida por la fórmula, se procede a calcular el grado de portabilidad:

$$GP = 1 - \left( \frac{1}{12} \right) = (1 - 0.125) * 100\% = 91\%$$

Esto concluye que el sistema es apto para funcionar bajo distintas plataformas, es una aplicación desarrollada bajo el lenguaje de programación PHP con una base de datos MySQL y servidor apache.

Con respecto al tamaño de la aplicación y la base de datos, estos ocupan alrededor de 50 MB debido a que se utilizó framework Codeigniter es por eso que la medida para llevar de un sistema a otro no requiere de un gran esfuerzo.

#### 4.1.4. Confiabilidad

Para determinar la confiabilidad del sistema especificaremos el instante en el que este comienza a funcionar, determinando por  $t_0 = 0$ . A partir de este momento se observa el trabajo del sistema hasta que se produzca una falla en el instante  $T$  que se va aproximando a un variable aleatoria continua, que nos determina la confiabilidad en términos probabilísticos.

Entonces tenemos las siguientes probabilidades:

$$P(T \leq t) = F(t) \quad (1) \text{ Probabilidad de fallas}$$

$$P(T \leq t) = 1 - F(t) \quad (2) \text{ Probabilidad de trabajo sin fallas}$$

Donde a que se tienen tiempos de inicio como de fin, para el cálculo de estas probabilidades se utiliza la distribución exponencial. Entonces la función  $F(t)$  está dada por:

$$F(t) = 0.94 e^{-\lambda * t} \quad \text{Ec. De confiabilidad}$$

Para el cálculo de error se consideran 100 ejecuciones en un mes en un periodo de 12 meses para obtener una probabilidad de fallas. En ese caso, el margen de error  $\lambda$  será de  $1/100$ . Además, se utiliza el resultado obtenido del punto función igual a 94% para la distribución en un periodo  $t = 12$  meses.

Obteniendo la portabilidad de fallas se tiene:

$$P(T \leq t) = F(t)$$

$$F(t) = 0.94 * e^{-\frac{1}{100} * 12}$$

$$F(t) = 0.8 \text{ Probabilidad de fallo}$$

Para el valor de la probabilidad de que el sistema trabaje sin fallos a partir del anterior resultado tenemos que: El software tendrá una confiabilidad de 92% en una gestión, lo cual indica que se mantendrá estable y en funcionamiento.

#### 4.1.5. Mantenibilidad

La Mantenibilidad es la interfaz a modificar. Incluyendo correcciones o mejoras del software, a cambios en el entorno, y especificaciones de requerimientos funcionales, para poder medir la calidad del mantenimiento del sistema utilizaremos el índice de madurez del software (IMS), que indica la estabilidad de un producto de software. El índice de madurez del software se calcula con la siguiente formula

$$IMS = \frac{[Mt - (Fa + Fb + Fc)]}{Mt}$$

Donde

***Mt*** = Número de módulos en la versión actual

***Fa*** = Número de módulos en la versión actual que se han cambiado

***Fb*** = Número de módulos en la versión actual que se han añadido

***Fc*** = Número de módulos en la versión anterior que se han borrado en la versión actual.

A continuación, se presenta la información requerida por la fórmula que se obtuvo la información que se muestra en la siguiente tabla:

Tabla 4. 6

*Datos*

Información	Valor
<b><i>Mt</i></b>	4
<b><i>Fa</i></b>	0

<b><i>Fb</i></b>	0
<b><i>Fc</i></b>	0

Fuente: Elaboración propia

Ahora calculamos el IMS, usando los valores obtenidos

$$IMS = \frac{[4 - (0 + 0 + 0)]}{4}$$

$$IMS = \frac{4}{4} = 100\%$$

#### 4.1.6. Resultados

Tabla 4. 7

*Resultado de métricas de calidad*

<b>Características</b>	<b>Resultados %</b>
<b>Usabilidad</b>	90%
<b>Funcionalidad</b>	94%
<b>Portabilidad</b>	91%
<b>Confiabilidad</b>	92%
<b>Mantenibilidad</b>	100%
<b>Total</b>	93.4%

Fuente: Elaboración propia

La calidad del sistema corresponde al 93%, lo que se interpreta como la satisfacción que tiene un usuario al interactuar con el sistema.

## 4.2. Cocomo II

Es útil para estimar el costo total del sistema se tomarán en cuenta los siguientes costos: Costo de la elaboración del proyecto, costos del software desarrollado, costos de la implementación del sistema

### 4.2.1. Punto de función

Primero se debe hallar el punto de función para poder obtener el costo del proyecto para ello vamos analizar cada una de las pantallas de nuestro sistema.

#### 4.2.1.1. Número de entradas de usuario

Tabla 4. 8  
*Entradas del prototipo*

Nro.	Entradas de Usuario
1	Ingreso de usuario
2	Búsqueda textual
3	Buscador de perfil

Fuente: Elaboración propia

**4.2.1.2. Número de salidas de usuario**

En la siguiente tabla proporciona información elaborada por el sistema que son gestionadas al usuario.

Tabla 4. 9  
*Sistemas gestionados al usuario*

Nro.	Salidas de usuario
1	Resultado de la búsqueda textual
2	Resultados de reporte o informe
3	Confirmación de datos de Facebook
4	Cerrar sesión

Fuente: Elaboración propia

**4.2.1.3. Número de peticiones de usuario**

En esta tabla veremos las peticiones que hace el usuario al prototipo.

Tabla 4. 10  
*Peticiones que hace el usuario al prototipo*

Nro.	Peticiones de Usuario
1	Autenticación al usuario
2	Autenticación al usuario por Facebook
4	Visualizar los resultados de la búsqueda
5	Visualizar reporte

Fuente: Elaboración propia

#### 4.2.1.4. Numero de Archivos

En esta tabla mostramos las salidas lógicas del prototipo

Tabla 4. 11  
*Salidas lógicas del prototipo*

Nro.	Archivos
1	Información de la búsqueda realizada.
2	Información de los perfiles de Facebook

Fuente: Elaboración propia

Una vez contando todos los datos requeridos de nuestra aplicación, lo siguiente es agruparlos para tener el parámetro de medición.

Tabla 4. 12  
*Interfaz parámetros de medición*

Parámetros de medición	de Cuentas	Factor de Ponderación			Totales
		Simple	Medio	Complejo	
Nro. De Entradas	3	6	10	24	30
Nro. De Salidas	4	4	5	7	20
Nro. De Peticiones	5	5	4	9	45
Nro. De Archivos	2	7	10	15	20
Totales de Cuentas					115

Fuente: Elaboración propia

#### 4.2.1.5. Cálculo de factores de ajustes de la complejidad

Tabla 4. 13

*Factor de complejidad*

Factor de complejidad	Valor
<b>Requiere copias de seguridad y recuperación</b>	4
<b>Necesita comunicación de datos</b>	4
<b>Se requiere entrada de datos en línea (on-line)</b>	4
<b>Transacción de salida en múltiples pantallas</b>	0
<b>Código diseñado para la reutilización</b>	4
<b>Conversión instalación en diseño</b>	2
<b>Aplicación diseñada para el cambio</b>	3
<b>Interface con el usuario</b>	4
<b>Actualización (on-line)</b>	0
<b>Procesamiento complejo</b>	3
<b>Facilidad de cambios</b>	2
<b>Volumen de transacciones</b>	2
<b>Total</b>	32

Fuente: Elaboración propia

Se calcula el factor de ajuste

$$Factor\ de\ Ajuste = 0.65 + 0.01 * \sum fi()$$

$$Factor\ de\ Ajuste = 0.65 + 0.01 * 32$$

$$Factor\ de\ Ajuste = 0.97$$

Calculando el punto de función con los datos obtenidos

$$\text{Punto Función} = \text{Cuenta Total} * \text{Factor de Ajustes}$$

$$\text{Punto Función} = 115 * 0.97$$

$$\text{Punto Función} = 111.55$$

#### 4.2.2. Costo del software desarrollado

Se utiliza como punto (PF)  $PF = 111.55$ , realizaremos la conversión de punto función a miles de líneas de código mediante la siguiente tabla

Tabla 4. 14  
Factor LCD/PF de lenguaje de programación

LENGUAJE	FACTOR
	LDC/PF
Java	53
JavaScript	47
Visual Basic	46
ASP	36
Visual C++	34
PHP	29
Ensamblador	320
C	150

Fuente QSM,2017

Aplicaremos las fórmulas básicas de esfuerzo, tiempo calendario y personal requerido, Las fórmulas de COCOMO II que utilizaremos serán las siguiente:

$$E_D = 2,4(KLDC)^{1.05}$$

$$T_D = 2,5(E_D)^{0.38}$$

Donde:

$E_D$  = Esfuerzo aplicado en personas por mes

$T_D$  = Tiempo de desarrollo en mes

KLDC = número estimado de líneas de código distribuidas.

$$LCD = PF * Factor = LCD$$

$$LCD = 111.55 * 29 = 3234.95$$

$$KLDC = 3.23$$

Calculando el esfuerzo en personas mes:

$$E_D = 2.4 * (3.23)^{1.05} = 8.22 \text{ personas/mes}$$

Calculando el tiempo de desarrollo mes

$$T_D = 2.5 * (8.22)^{0.38} = 5.56 \text{ meses}$$

Calculando el personal requerido para el desarrollo del proyecto se obtiene con la siguiente formula:

$$\text{Numero de programadores} = \frac{E_D}{T_D} = \frac{8.22}{5.56} = 1.5$$

Por lo tanto, se necesita 2 programadores para el desarrollo del prototipo. El salario de un programador es de 213 \$us/mes. Por lo tanto, con este dato la estimación del costo del software se calculará con la siguiente formula.

$$\text{Costo Software} = \# \text{ de programadores} * \text{salario de programador} * \# \text{ de mes}$$

$$\text{Costo Software} = 2 * 213 * 5 = 2130$$

Por lo tanto, el costo de desarrollo del software es de 2130 \$us.

### 4.2.3. Costo de la elaboración del proyecto

Los costos de elaboración del proyecto se refieren a los costos de estudio del sistema en la etapa de análisis.

Tabla 4. 15

*Costo de elaboración del proyecto*

Descripción	Costo (\$us)	Costo (Bs)
<b>Análisis y diseñado del proyecto</b>	21	150
<b>Material de escritorio</b>	30	208
<b>Computadora</b>	800	5568
<b>Desarrollo del Software</b>	2130	14824
<b>Internet</b>	22	154
<b>Total</b>	3003	20904

Fuente: Elaboración propia

Por lo tanto, el costo total para la elaboración del proyecto es de 3003 \$us y en Bs equivale a 20904 Bs.



# CAPITULO V

## PRUEBAS Y RESULTADOS

En el siguiente capítulo se describe las pruebas para comprobar la hipótesis planteada y también mostrar los resultados que sustenten este trabajo de investigación

### 5.1. INTRODUCCIÓN

En este capítulo el objetivo central es el de demostrar que la hipótesis planteada en el primer capítulo se cumplió, para esto usaremos herramientas de tipo estadístico.

Al usar el método estadístico se podrán realizar pruebas unitarias al sistema como también pruebas globales de todo su funcionamiento

### 5.2. PRUEBA DE HIPÓTESIS

#### 5.2.1. Pasos de la prueba de hipótesis

Etapas para una prueba de hipótesis.

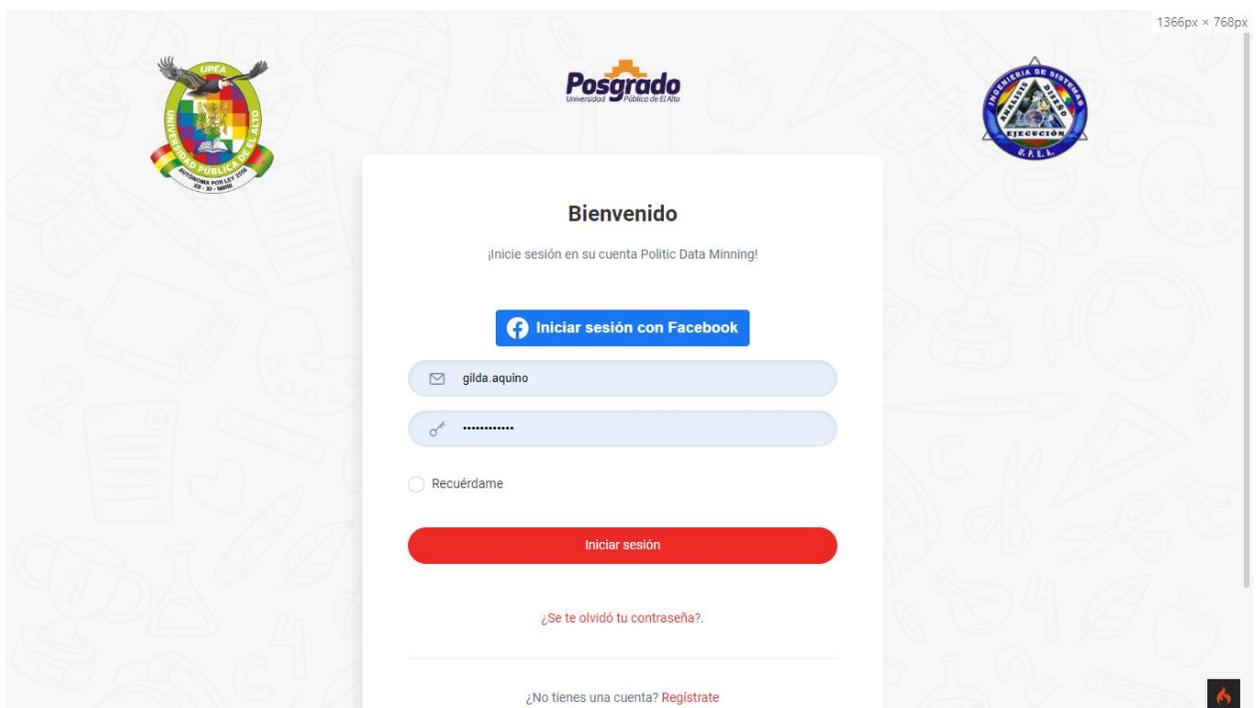
- Primero se define la hipótesis nula, en este caso la hipótesis del investigador.
- Se formula la hipótesis alternativa o también llamada contra hipótesis.
- Se elige un nivel de significancia y a partir de este nivel se construye la zona de aceptación.
- Con la zona de aceptación surge la zona de rechazo la cual se llama región crítica y su área es el nivel de significación o aceptación.
- Se verifica la hipótesis extrayendo una muestra.

- Se evalúa con una prueba estadística.
- Mediante el proceso se evalúa el cumplimiento de la hipótesis.

### 5.3. PRUEBAS

Una vez explicados los elementos que componen al sistema, su funcionalidad, su desarrollo y demás características es importante realizar una fase de pruebas las cuales se verán a continuación.

El prototipo cuenta con un inicio de sesión activa (Véase en la figura 5.1).



*Figura 5. 1 Pruebas del prototipo*

*Fuente: Elaboración propia*

El prototipo cuenta con un visor de perfiles que más comentan en la página de Posgrado (Véase en la figura 5.2).

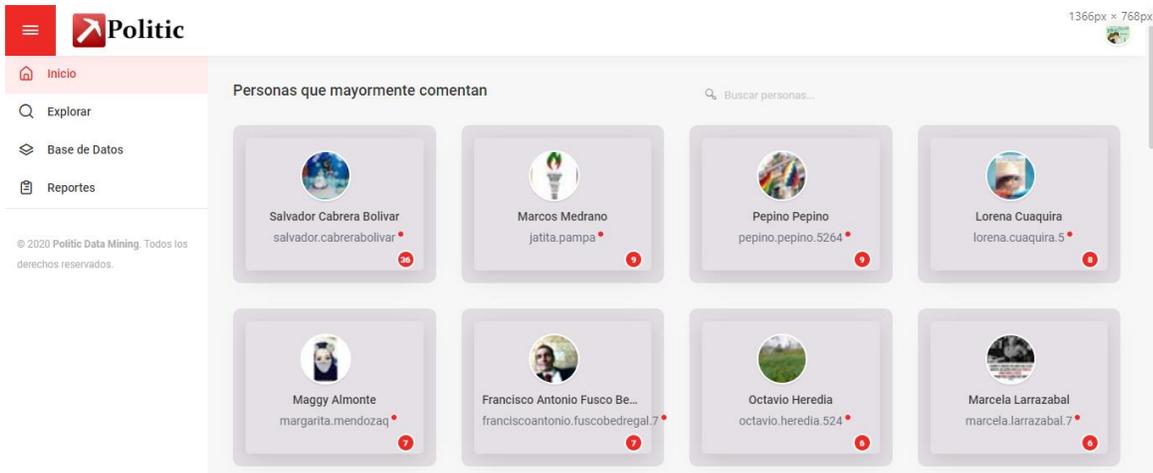


Figura 5. 2 Pruebas del prototipo

Fuente: Elaboración propia

Comenzamos con las pruebas de búsqueda de patrones con algoritmos de minería de texto (Véase en la figura 5.3).

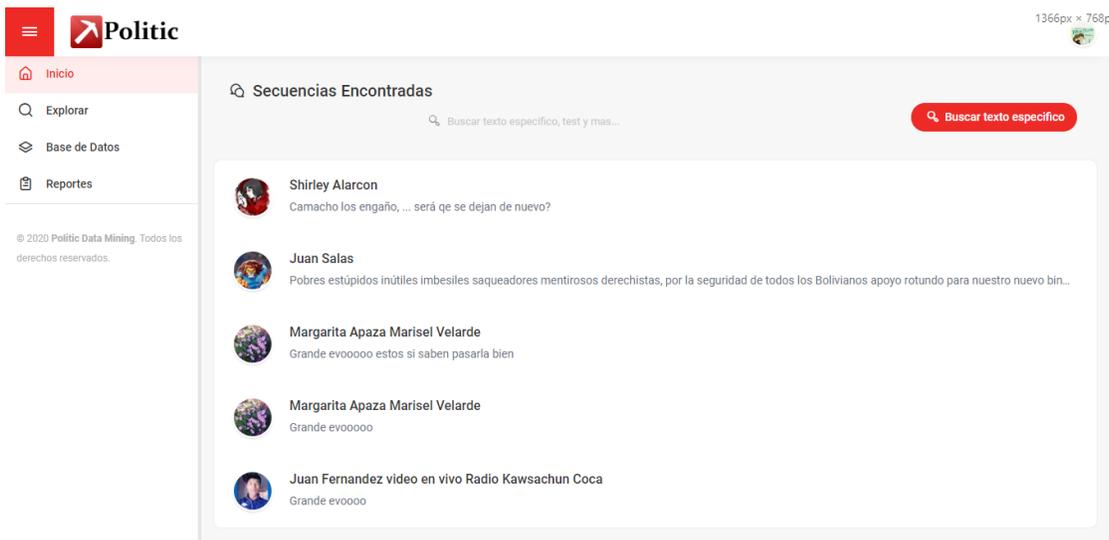


Figura 5. 3 Pruebas del prototipo

Fuente: Elaboración propia

Una vez terminado la búsqueda podemos apreciar cómo se generan las estadísticas y los reportes (Véase en la figura 5.4).

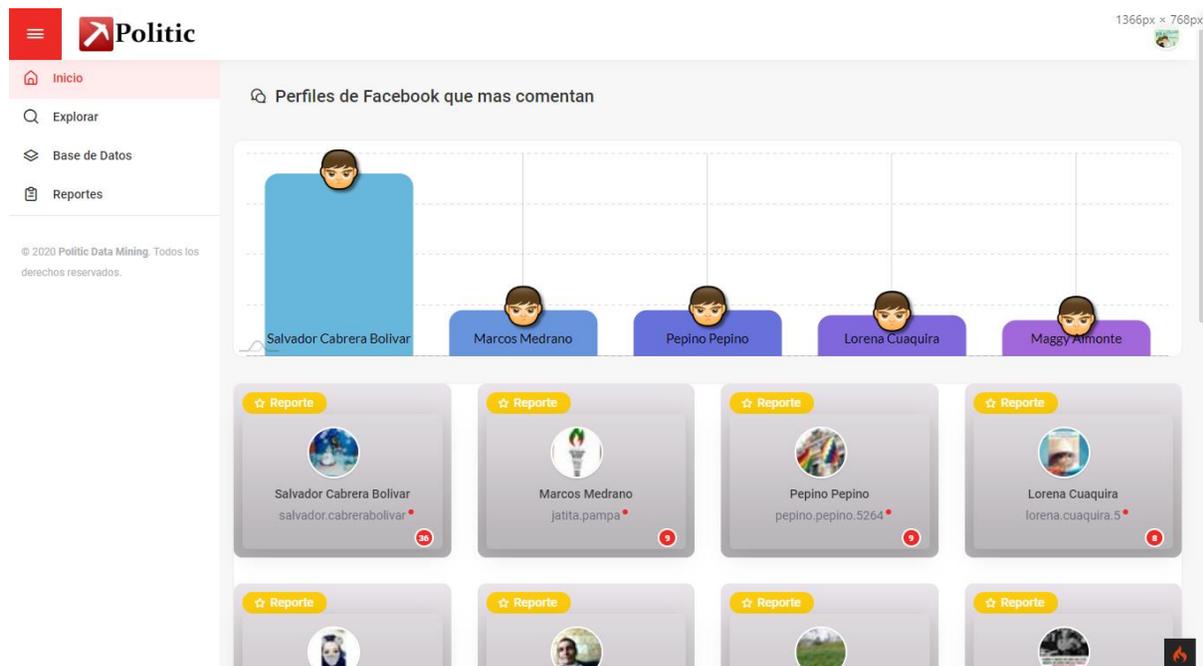


Figura 5. 4 Pruebas del prototipo

Fuente: Elaboración propia

### 5.3.1. Resultados

En la Primera prueba se realizará la ejecución del prototipo haciendo búsquedas secuenciales

Tabla 5. 1

*Tabla de pruebas realizadas*

Descripción	Nro. de Pruebas	Porcentaje
<b>Pruebas aceptadas</b>	28	93%
<b>Pruebas rechazadas</b>	2	7%
<b>Total, de pruebas</b>	30	100%

Fuente: Elaboración propia

En la Segunda prueba en lo que hacemos énfasis es la muestra de perfiles de usuario.

Tabla 5. 2

*Pruebas realizadas en el usuario*

Descripción	Nro. De Pruebas	Porcentaje
<b>Pruebas aceptadas</b>	96	96%
<b>Pruebas rechazadas</b>	4	4%
<b>Total, de pruebas</b>	100	100%

Fuente: Elaboración propia

En la Tercera prueba en lo que hacemos énfasis en la emisión de reportes

Tabla 5. 3  
*Pruebas realizadas emisión de reportes*

Descripción	Nro. De Pruebas	Porcentaje
<b>Pruebas aceptadas</b>	39	98%
<b>Pruebas rechazadas</b>	1	2%
<b>Total, de pruebas</b>	40	100%

Fuente: Elaboración propia

### 5.3.2. Evaluación de resultados

En este punto ahora aplicaremos el método estadístico que se usara para el presente trabajo es el de proporciones.

<b>Pruebas totales:</b>	170
<b>Pruebas aceptadas:</b>	163
<b>Pruebas reprobadas:</b>	7
<b>Porcentaje de éxitos:</b>	96%
<b>Porcentaje de fracasos:</b>	4%

Como se observa en los resultados, el porcentaje de éxitos es mayor al 95% esperado al momento de plantearse la hipótesis, pero para comprobar de manera cuantificable si este valor se asemeja al valor esperado, se realiza una prueba de hipótesis estadística.

Las variables usadas en dicha prueba serán las mismas mencionadas en la evaluación de casos de prueba:

$$\begin{aligned}
 P_0 &= 95 \\
 \alpha &= 5 \\
 p &= 96 \\
 n &= 170
 \end{aligned}$$

N es el número total de sujetos en el espacio muestral y no se conoce. Además, que se tomará un nivel de significancia  $\alpha$  del 5%.

Para el caso de la hipótesis se tiene los siguientes datos:

En este caso se espera que el porcentaje de éxitos sea igual o mayor a 95%:

$$H_0: P \geq P_0$$

$$H_1: P < P_0$$

Reemplazando:

$$H_0: P \geq 0.95$$

$$H_1: P < 0.95$$

#### 5.4. DETERMINACIÓN DE LA REGIÓN CRÍTICA

La región crítica para la hipótesis planteada es la siguiente:

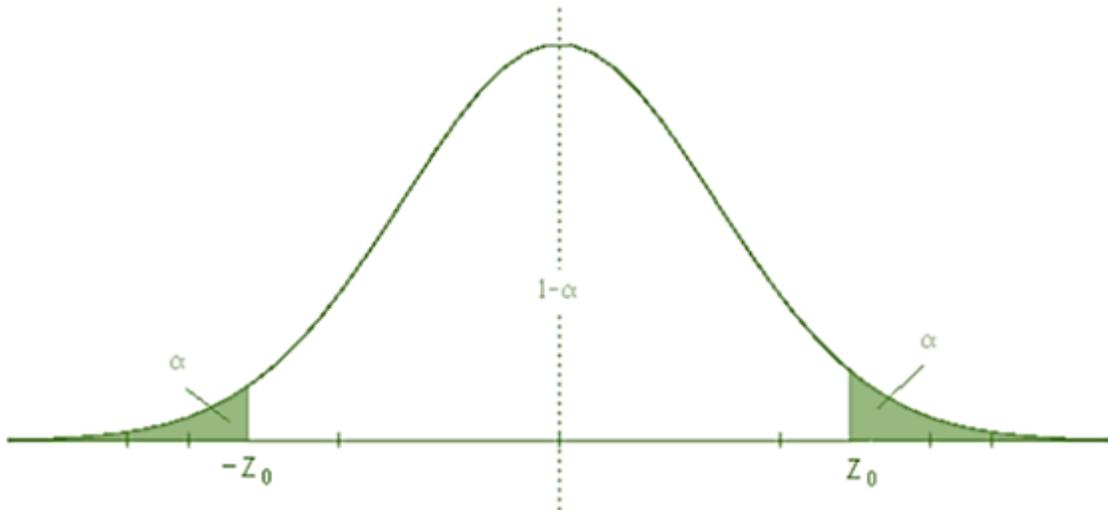


Figura 5. 5 Región crítica para la hipotenusa

Fuente: Elaboración propia

Como n se refiere en este caso al número de pruebas, en este caso 170, el punto crítico a usar es  $-Z_o$  y se determina mediante:

$$-Z_o = -Z_{1-\alpha} = -Z_{1-0.05}$$

$$-Z_o = -Z_{0.05}$$

Este valor se halla de la tabla de la función de distribución normal, la cual se encuentra en la sección de anexos. Para obtener el valor de z se elige de la tabla mencionada el valor más cercano a 0,95; el cual está ubicado en la fila 1,6 y columna 0,04.

$z_0$	0,00	0,01	0,02	0,03	0,04	0,05	0,06	0,07	0,08	0,09	$z_0$
0,0	0,5000	0,5040	0,5080	0,5120	0,5160	0,5199	0,5239	0,5279	0,5319	0,5359	0,0
0,1	0,5398	0,5438	0,5478	0,5517	0,5557	0,5596	0,5636	0,5675	0,5714	0,5753	0,1
0,2	0,5793	0,5832	0,5871	0,5910	0,5948	0,5987	0,6026	0,6064	0,6103	0,6141	0,2
0,3	0,6179	0,6217	0,6255	0,6293	0,6331	0,6368	0,6406	0,6443	0,6480	0,6517	0,3
0,4	0,6554	0,6591	0,6628	0,6664	0,6700	0,6736	0,6772	0,6808	0,6844	0,6879	0,4
0,5	0,6915	0,6950	0,6985	0,7019	0,7054	0,7088	0,7123	0,7157	0,7190	0,7224	0,5
0,6	0,7257	0,7291	0,7324	0,7357	0,7389	0,7422	0,7454	0,7486	0,7517	0,7549	0,6
0,7	0,7580	0,7611	0,7642	0,7673	0,7704	0,7734	0,7764	0,7794	0,7823	0,7852	0,7
0,8	0,7881	0,7910	0,7939	0,7967	0,7995	0,8023	0,8051	0,8078	0,8106	0,8133	0,8
0,9	0,8159	0,8186	0,8212	0,8238	0,8264	0,8289	0,8315	0,8340	0,8365	0,8389	0,9
1,0	0,8413	0,8438	0,8461	0,8485	0,8508	0,8531	0,8554	0,8577	0,8599	0,8621	1,0
1,1	0,8643	0,8665	0,8686	0,8708	0,8729	0,8749	0,8770	0,8790	0,8810	0,8830	1,1
1,2	0,8849	0,8869	0,8888	0,8907	0,8925	0,8944	0,8962	0,8980	0,8997	0,9015	1,2
1,3	0,9032	0,9049	0,9066	0,9082	0,9099	0,9115	0,9131	0,9147	0,9162	0,9177	1,3
1,4	0,9192	0,9207	0,9222	0,9236	0,9251	0,9265	0,9279	0,9292	0,9306	0,9319	1,4
1,5	0,9332	0,9345	0,9357	0,9370	0,9382	0,9394	0,9406	0,9418	0,9429	0,9441	1,5
1,6	0,9452	0,9463	0,9474	0,9484	0,9495	0,9505	0,9515	0,9525	0,9535	0,9545	1,6
1,7	0,9554	0,9564	0,9573	0,9582	0,9591	0,9599	0,9608	0,9616	0,9625	0,9633	1,7
1,8	0,9641	0,9649	0,9656	0,9664	0,9671	0,9678	0,9686	0,9693	0,9699	0,9706	1,8
1,9	0,9713	0,9719	0,9726	0,9732	0,9738	0,9744	0,9750	0,9756	0,9761	0,9767	1,9

Figura 5. 6 Función de distribución

Fuente: Elaboración propia

El valor de Z se obtiene sumando ambos valores:

$$Z_{0,95} = (1.6 + 0.04) = 1.64$$

Se relocizará el cálculo estadístico con los valores que tenemos:

Como se conoce el número total de individuos en el espacio muestral, el valor del estadístico de la prueba se obtiene mediante la fórmula:

$$Z_c = \frac{p - p_o}{\sqrt{\frac{p_o q_o}{n}}} = \frac{0.96 - 0.95}{\sqrt{\frac{0.95 * 0.05}{170}}} = 0.5982$$

Al comparar el valor de los estadísticos  $Z_0$  y  $Z_c$ , se observa que el valor del estadístico de la prueba no se encuentra dentro de la región crítica, por lo tanto, se acepta la hipótesis de investigación  $H_i$ .



Figura 5. 7 Distribución de  $Z_c$  para la toma de decisiones

Fuente: Elaboración propia

El promedio de éxito del prototipo al momento de reconocer las muestras se acerca al 95%. Por tanto, como no se rechaza  $H_i$  se podría concluir y afirmar la hipótesis  $H_i$ :

**“La minería de texto nos permite extraer comentarios con fines políticos para analizar las opiniones de la página de posgrado con una eficiencia del 95%”**





# CAPITULO VI

## CONCLUSIONES Y RECOMENDACIONES

En el presente capítulo se muestra las conclusiones de la investigación realizada además de las recomendaciones y estado de los objetivos e hipótesis.

### 6.1. ESTADO DE HIPÓTESIS

De acuerdo a la muestra, existe suficiente evidencia para demostrar que la minería de texto nos permite extraer comentarios con fines políticos para analizar las opiniones de la página de posgrado con una eficiencia del 95%

### 6.2. ESTADO DE LOS OBJETIVOS

La implementación del proceso de minería de texto permite afirmar los objetivos proyectados, sustentados en:

- Se mejoró la búsqueda que antes fue manual de los comentarios de la página oficial de Posgrado en Facebook.
- Se llegó a implementar una de una base de datos estandarizada para guardar el historial de comentarios de cada perfil, ahora podremos saber que perfil tiene más tendencias políticas.

- Se desarrolló un algoritmo que llego a analizar la totalidad de los comentarios realizados en la página Unidad de Posgrado.
- Se utilizó diferentes librerías de PHP para poder realizar la representación e interpretación de la información obtenida.

### **6.3. RECOMENDACIONES**

Con respecto al trabajo de investigación algunas recomendaciones que puedo dejar es que al momento que se realice el procesamiento de datos se trate de reducir casi por completo el ruido en los datos ya que si se deja algún comentario con palabras vacías o algún comentario sin haber realizado el proceso de stemming producirá un mayor error.

Para trabajos futuros con respecto al trabajo de investigación presentado como base se podría analizar los comentarios de Facebook y conjuntamente con el API de Google para predecir acciones se puede predecir el mejor comentario de la publicación con base a los datos de salida de este trabajo de investigación.

## BIBLIOGRAFÍA

Vallejos, S. (2006). Obtenido de

[http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/Mineria\\_Datos\\_Vallejos.pdf](http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/Mineria_Datos_Vallejos.pdf)

Fuentes Alarcon, P. C. (2015). *Influencia de procesamiento de datos dentro del desempeño de modelos de perfilamientos de cliente elaborados con herramienta de minería de datos*. Facultad de Ingeniería.

Aguilar Ruiz, J. (2007). *Discovering alpha expression patterns from gene expression data*. Diciembre: LNCS.

Alava, N. (18 de Abril de 2015). Obtenido de

<https://ingenieriaensoftwarerathalyalava.wordpress.com/2015/04/18/procesos-del-software-y-modelo-cascada/>

Balderrama Vasquez, C. R. (2017). *DocPlayer*. Obtenido de <https://moodle2.unid.edu.mx>

Beltran, A. (06 de 28 de 2020). Monitorean cuentas oficiales de partidos en redes sociales en silencio electoral. *ATBdigital*, págs. 1-2.

Boehm, B. (2011). *Slideshare*. Obtenido de Ingeniería de sistemas utea.

Carabali, M. (septiembre de 2013). *Prezi*. Obtenido de <https://prezi.com/sjwfwmix7slk/pruebas-de-caja-negra-y-caja-blanca/>

Cervantes, H. (2006). *Software Guru*. Obtenido de <https://sg.com.mx/revista/27/arquitectura-software>

Chapman, P., Clinton, J., & Kerber, R. (2000). *Data Mining Guide*. Obtenido de <https://www.the-modeling-agency.com/crisp-dm.pdf>

Chiavenato, I. (2006). *Introducción a la teoría general 7ma Edición*. Delegación Cuajimalpa, México, D. F: McGraw-Hill.

Christopher, A. (2005). *Pattern Language* .

Cortes Morales, R. (2006). *Introducción al análisis de sistemas de ingeniería de software*. EUNED.

D'Ambrosio, S. (2020). *monografias.com*. Obtenido de <https://www.monografias.com/trabajos14/datos/datos.shtml>

Daza Vergaray, A. (2004). *DATA MINIG (Minería de datos)*. Lima, Peru Av. Paseo de la Republica N.° 5613, Miraflores: Macro EIRL.

Dyche, J. (2000). *Convertir datos en información con el almacenamiento de datos*. Wesley Professional.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). *The kdd Process for Extracting Useful*. Communications of the acm.

Fernandez, D. (2007). Obtenido de <https://www.redalyc.org/pdf/3783/378343637004.pdf>

Galiano, L. (2012). *blogSpot*. Obtenido de <http://elproyectodeluisgaliano.blogspot.com/2012/11/metodologia-uwe-aplicada-mi-solucion.html>

Gonzales Marcos, A. (2006). *Dialnet*. Obtenido de <https://dialnet.unirioja.es/servlet/tesis?codigo=1166>

Gonzales Reyes, D. (2017). *Monografias.com*. Obtenido de <https://sg.com.mx/revista/27/arquitectura-software>

Gorunescu. (2011). *Scielo*. Obtenido de [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S2218-36202016000400005](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2218-36202016000400005)

- Guitierrez Aleman, J. M. (2017). *Repositorio.Umsa*. Obtenido de <https://repositorio.umsa.bo/bitstream/handle/123456789/12553/T.3286.pdf?sequence=1&isAllowed=y>
- Han, J., & Kamber, M. (2001). *Data Mining Concepts and Techniques*. San Francisco: Morgan: Kaufmann Publishers.
- Hearst. (1999). *SlideShare*. Obtenido de <https://es.slideshare.net/janettejf/mineria-de-datos-3582262>
- Hernandez, J. (2004). *ntroducción a la Minería de Datos*.
- Hernández, J., Ramírez, M., & Ferri, C. (2005). *Introducción a la Minería de Datos*. Madrid: Pearson Educación sa.
- Justicia Torre, M. C. (2017). *Nuevas Tecnicas de Minería de Datos*. Obtenido de <https://digibug.ugr.es/handle/10481/46975?locale-attribute=es>
- Lara Tarralbo, J. A. (21 de julio de 2016). *Udimia*. Obtenido de <https://www.udima.es/es/conocimiento-bases-datos>
- Larose, D., & Larose, C. (2014). *Discovering Knowledge in Data: An Introduction to Data Mining (2da Edicion)*. New Jersey.
- Leiva, A. (2016). *devexperto*. Obtenido de <https://devexperto.com/patrones-de-diseno-software/>
- LMU – Ludwig-Maximilians, U. M. (10 de Agosto de 2016). *UWE – UML-based Web Engineering*.
- Mata, R. (13 de Enero de 2017). *ICDM*. Obtenido de <https://www.icemd.com/digital-knowledge/articulos/mineria-datos-proceso-areas-se-puede-aplica/>
- Michalski. (1986). Obtenido de [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S2218-36202016000400005](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2218-36202016000400005)

Molina Felix, L. C. (2002). *businessintelligence*. Obtenido de <https://www.businessintelligence.info/resources/assets/dss/molina-torturando-datos.pdf>

Molina, J. (2001). *Analisis de Redes Sociales*. Barcelona: Bellaterra .

Molina, L. C. (2002). *datos*.

Monarchi , & Puhr. (1992). *Grial*. Obtenido de [https://repositorio.grial.eu/bitstream/grial/1146/1/IS\\_I%20Tema%207%20-%20Analisis%20Orientado%20a%20Objetos.pdf](https://repositorio.grial.eu/bitstream/grial/1146/1/IS_I%20Tema%207%20-%20Analisis%20Orientado%20a%20Objetos.pdf)

Monterol, Y. H. (2006). Obtenido de <https://www.redalyc.org/pdf/3783/378343637004.pdf>

Muñoz, E. (5 de Mayo de 2015). *Sumilla*. Obtenido de <https://sites.google.com/site/mineriadedatosunalm>

Nasukawa. (2001). *FUOC*. Obtenido de <https://www.businessintelligence.info/resources/assets/dss/molina-torturando-datos.pdf>

Nuez, A. (2001). Obtenido de <https://www.redalyc.org/pdf/3783/378343637004.pdf>

Pablo, J. (19 de Agosto de 2013). *elConspirador*. Obtenido de <https://www.elconspirador.com>

Pavez. (2000). *Scielo*. Obtenido de [https://scielo.conicyt.cl/scielo.php?script=sci\\_arttext&pid=S0718-07642006000200011](https://scielo.conicyt.cl/scielo.php?script=sci_arttext&pid=S0718-07642006000200011)

Perez Marquez, M. (2014). *Mineria de Datos*. Calle Mar Mediterráneo, 2. Nave 6 España: Copyright RC Libros.

Pessman, R. S. (2006). *Ingenieria de software (7ma Edicion)*. Mexico: The McGraw-Hil.

Piatetsky, S., & Frawley. (2000). *Data Mining Minería de Datos*. ACM,.

Posted. (12 de 2016). *PowerData*. Obtenido de <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/calidad-de-datos-en-mineria-de-datos-a-traves-del-preprocesamiento>

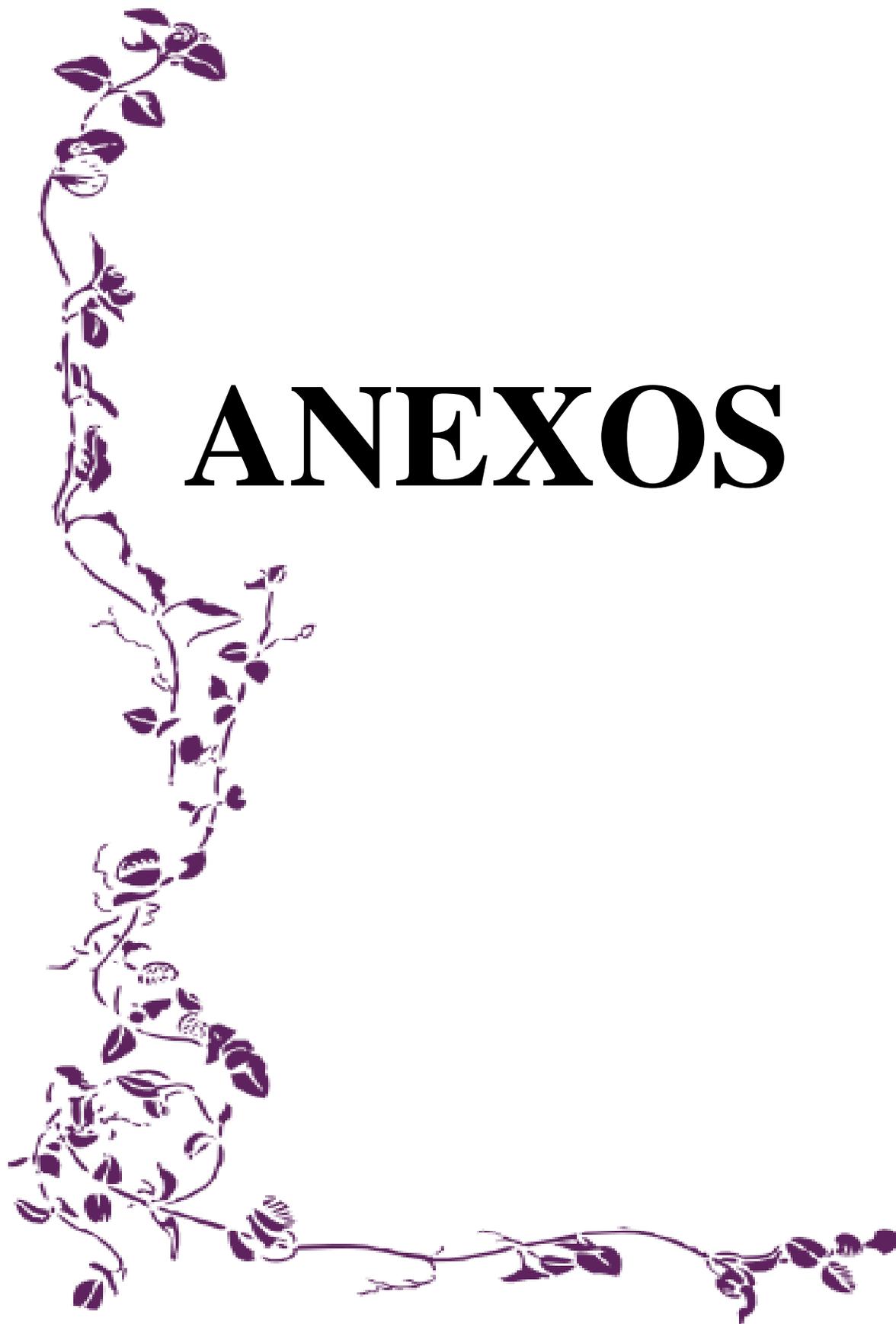
Pressman, R. S. (2010). *Ingenieria de software*. Mexico: The McGraw-Hil.

Rodriguez, O. (2013). *Mineria de Datos*.

- Rufenba. (14 de julio de 2014). *Genbeta*. Obtenido de <https://www.genbeta.com/desarrollo/patrones-de-diseno-que-son-y-por-que-debes-usarlos>
- Salazar Martinez, E. (Enero de 2015). *Informatica Juridica*. Obtenido de <http://www.informatica-juridica.com/trabajos/procedimiento-realizar-pruebas-caja-blanca/>
- Sanchez, M. A. (22 de Noviembre de 2017). *Medium*. Obtenido de <https://medium.com/all-you-need-is-clean-code/patrones-de-dise%C3%B1o-b7a99b8525e>
- Schwabe, O. D., Olsina, L., & Rossi, G. P. (2008). *Web Engineering - Modelling and Implementing WebApplications*. Londres: Springe.
- Servente, M. (Febrero de 2002). *BuenosAires*. Obtenido de <http://laboratorios.fi.uba.ar/lsi/servente-tesisingenieriainformatica.pdf>
- Sommerville, I. (2005). *Ingenieria de Sotware (7ma Edicion)*. Madrid: Pearson Educacion.
- Srikant, R., & Agrawal, R. (1994). *Fast Algorithms for Mining Association Rules*. Chile: VLDB.
- Thompson, I. (Octubre de 2008). *Promonegocios.net*. Obtenido de <https://www.promonegocios.net/mercadotecnia/que-es-informacion.html>
- Thuraisingham,, B. (1999). *Database Technologies: Concepts, Methodologies, Tools, and Applications*. EGI.
- Vallejos, S. (2006). *exa.unne.edu.ar*. Obtenido de [http://exa.unne.edu.ar/informatica/SO/Mineria\\_Datos\\_Vallejos.pdf](http://exa.unne.edu.ar/informatica/SO/Mineria_Datos_Vallejos.pdf)
- Vallejos, S. (s.f.). *unne*. Obtenido de [http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/Mineria\\_Datos\\_Vallejos.pdf](http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/Mineria_Datos_Vallejos.pdf)
- Vladimirou, T. (2005). *Revista de Economia y Empresa*.
- Wang, X. (1999). *Data Mining and Knowledge Discovery for Process*. London: Springer-Verlag.
- Weitzenfeid, A. (2002). *Ingenieria de Software*. Mexico D.F.: Thomson.

Zavala Ruiz, J. (2003). *Semrush*. Obtenido de

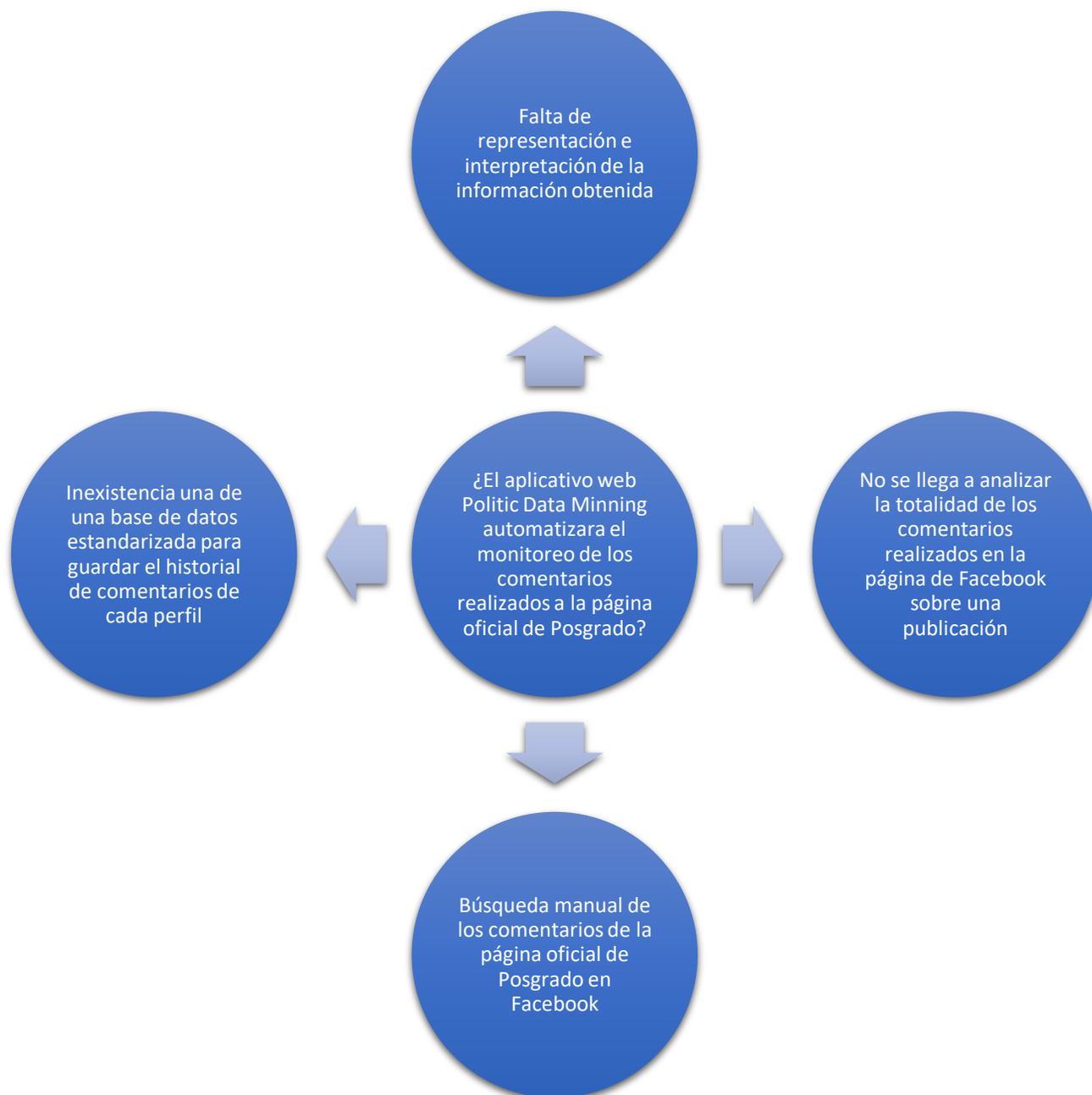
[https://www.researchgate.net/publication/283546859\\_Por\\_Que\\_Fracasan\\_los\\_Proyectos\\_de\\_Software\\_Un\\_Enfoque\\_Organizacional](https://www.researchgate.net/publication/283546859_Por_Que_Fracasan_los_Proyectos_de_Software_Un_Enfoque_Organizacional)



# ANEXOS

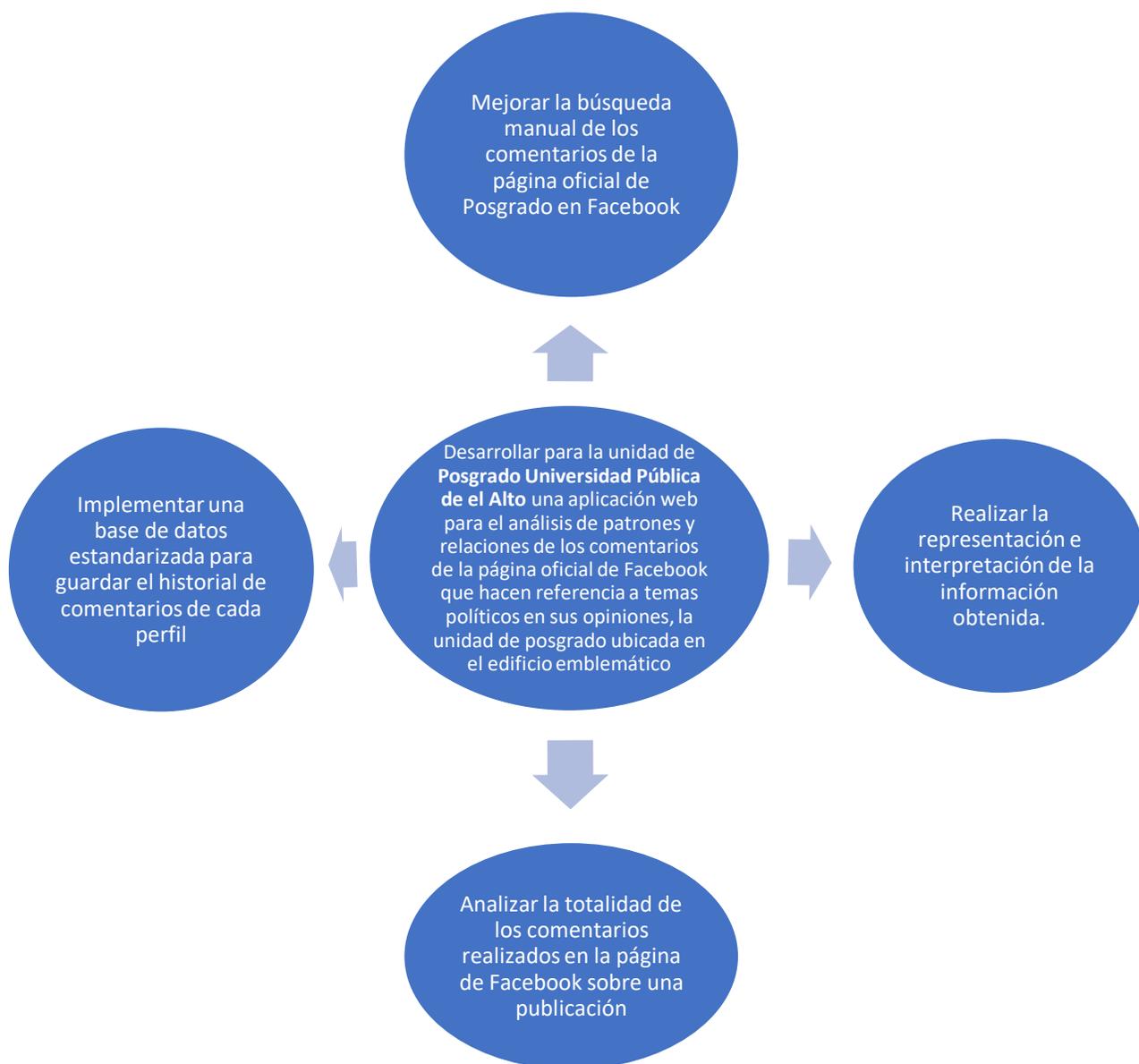
## ANEXO A

### ÁRBOL DE PROBLEMAS



## ANEXO B

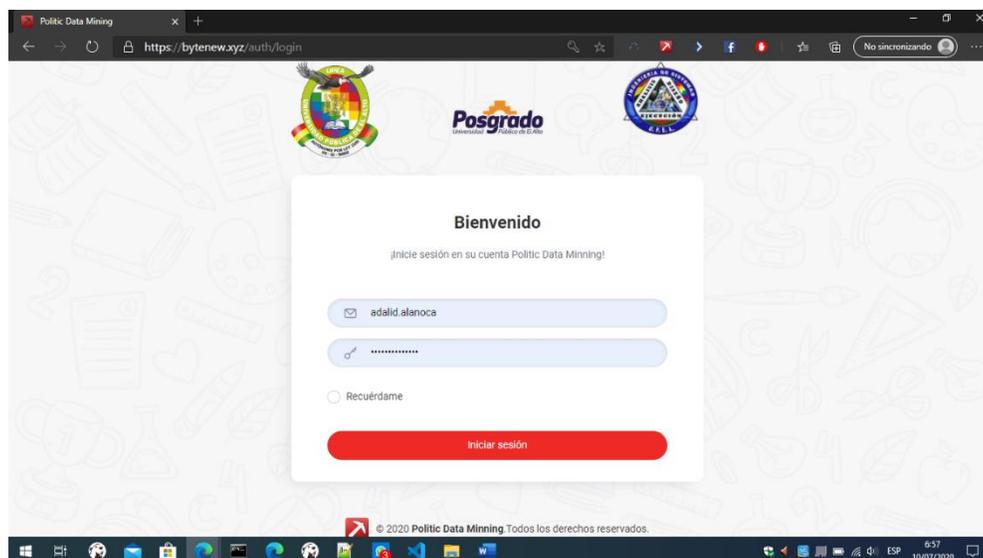
### ÁRBOL DE OBJETIVOS



## ANEXO C

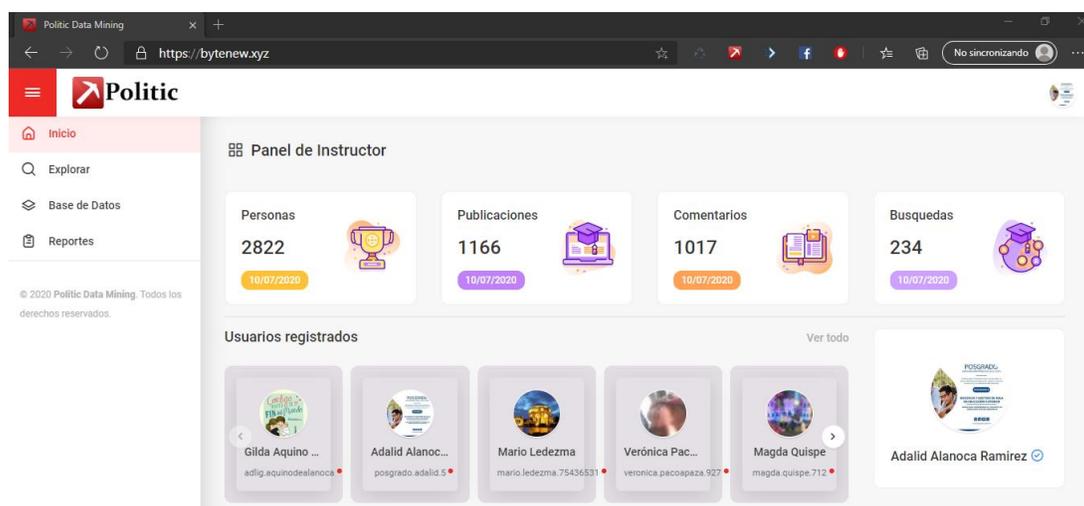
### MANUAL DE USUARIO (ADMINISTRADOR)

#### 1. En primera instancia se presenta el inicio de sesión al usuario



- Los campos necesarios son el nombre de usuario y la contraseña que serán entregados personalmente.
- El campo **Recuérdeme** se refiere si desea guardar una sesión más larga, la sesión continuara activa, aunque cierre el navegador o apague la máquina.
- Hagamos click en Iniciar Sesión

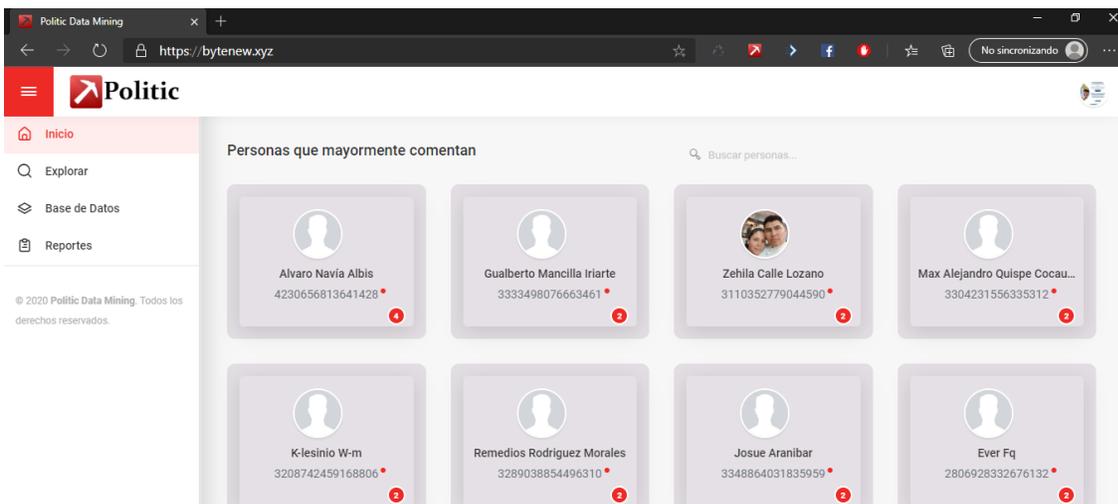
#### 2. Se carga el panel del Administrador



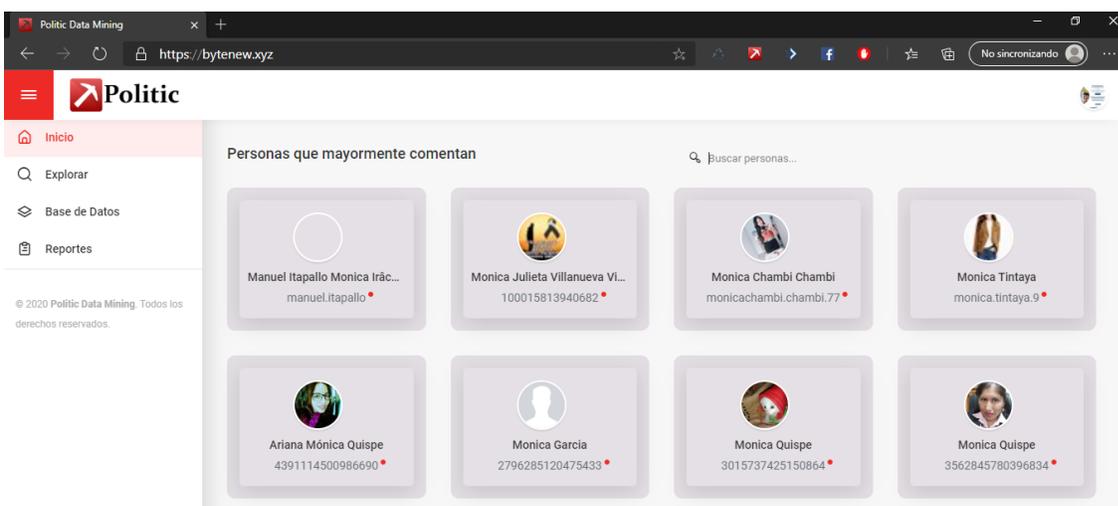
- En la primera columna detallamos la cantidad de personas registradas, publicaciones, comentarios y las búsquedas realizadas, estos datos ya se encuentran en nuestra base de datos.
- En la parte inferior está el listado de las personas que al presionar en la foto de perfil nos redirecciona hacia su página oficial de Facebook.

### 3. Visualizaremos las personas que más comentaron

a. Para ello haremos click en ver mas



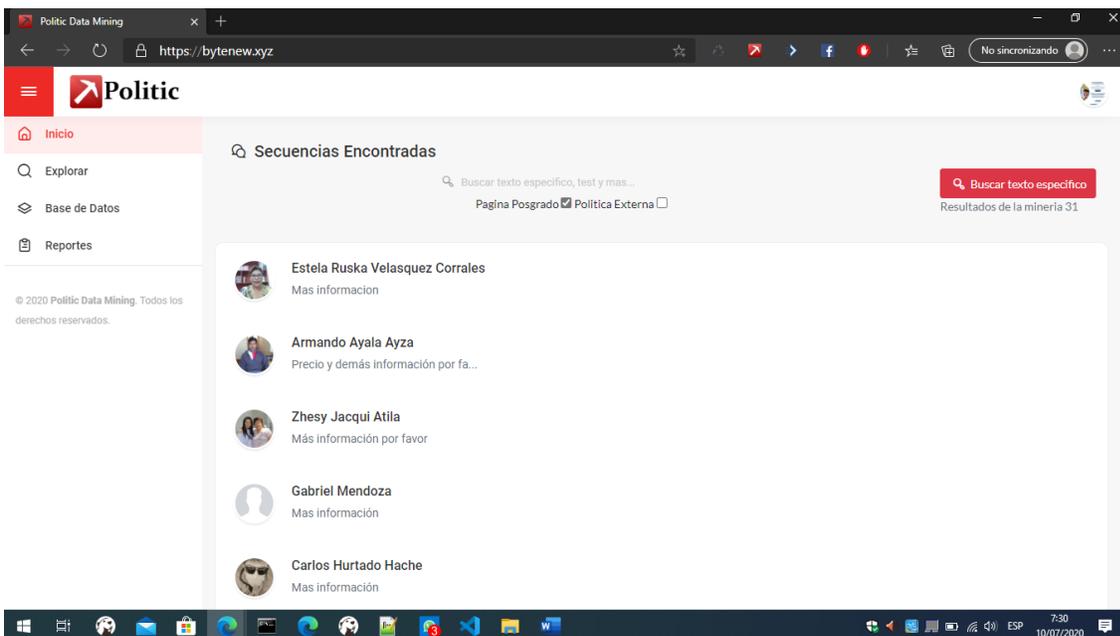
- Nos muestra un listado con los perfiles que más comentaron más en la página de Posgrado.
- También podemos hacer una búsqueda de perfiles



- De la misma forma al presionar en la foto de perfil nos redirecciona hacia su página oficial de Facebook.

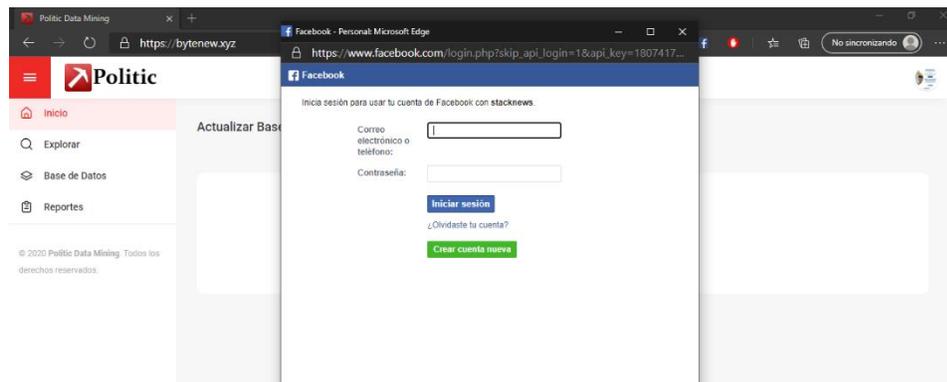
#### 4. Presionamos en Explorar

- a. Este apartado es para hacer la minería de texto.
- b. Puede escribir un texto en concreto y presionar Buscar texto específico.
- c. Puede seleccionar si desea que busqué solo en la página de Posgrado o buscar Política Externa

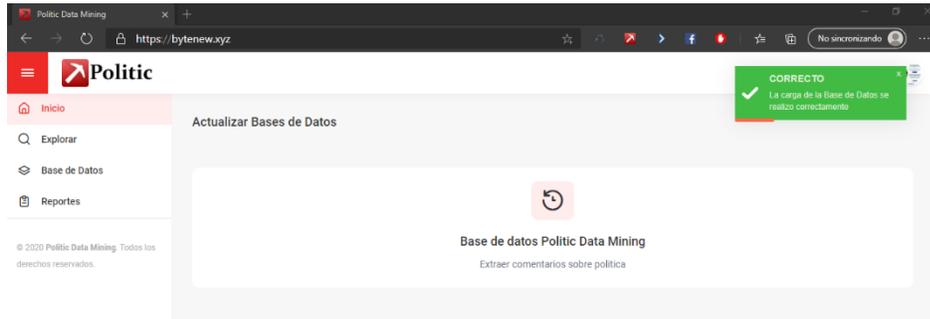


#### 5. Presionamos en Base de Datos

- a. Para recargar los comentarios de la página de Posgrado es necesario ser administrador de la página.



b. Al iniciar sesión automáticamente se recargarán los comentarios



## 6. Presionamos en Reportes

a. Nos muestra un ranking de los perfiles con más interacción en la página.



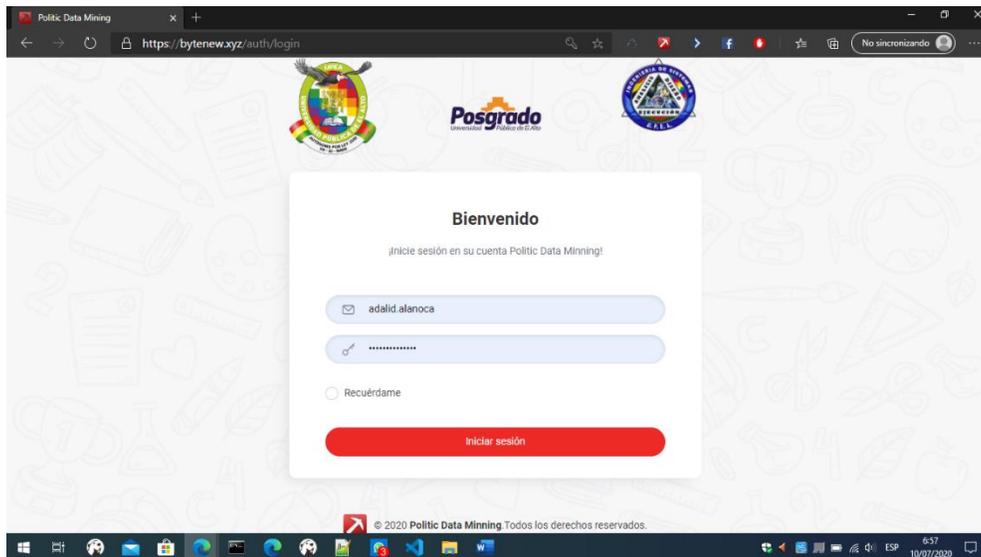
b. Presionemos en reporte



i. Se extrae un reporte directamente para impresión de un perfil.

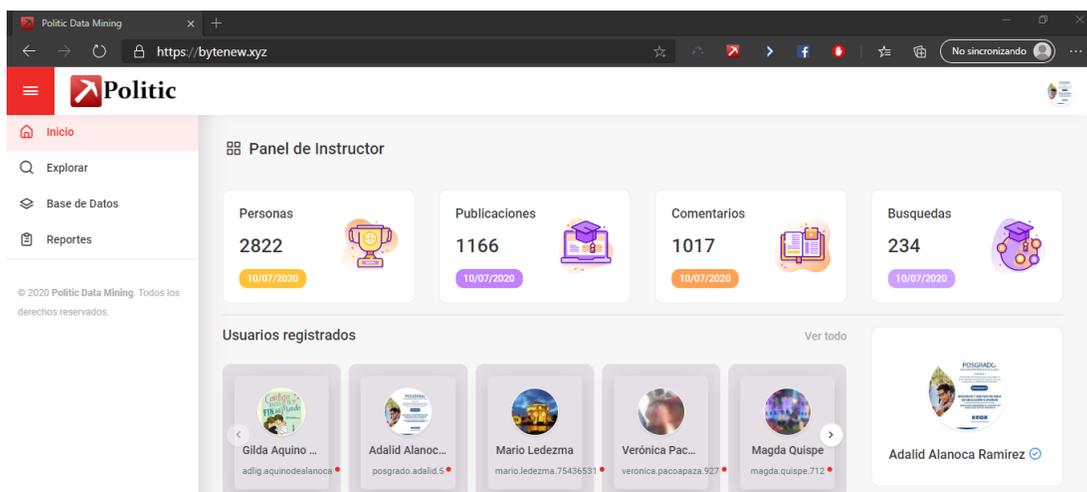
# MANUAL DE USUARIO (VISOR)

## 1. En primera instancia se presenta el inicio de sesión al usuario



- Los campos necesarios son el nombre de usuario y la contraseña que serán entregados personalmente.
- El campo **Recuérdame** se refiere si desea guardar una sesión más larga, la sesión continuara activa, aunque cierre el navegador o apague la máquina.
- Hagamos click en Iniciar Sesión

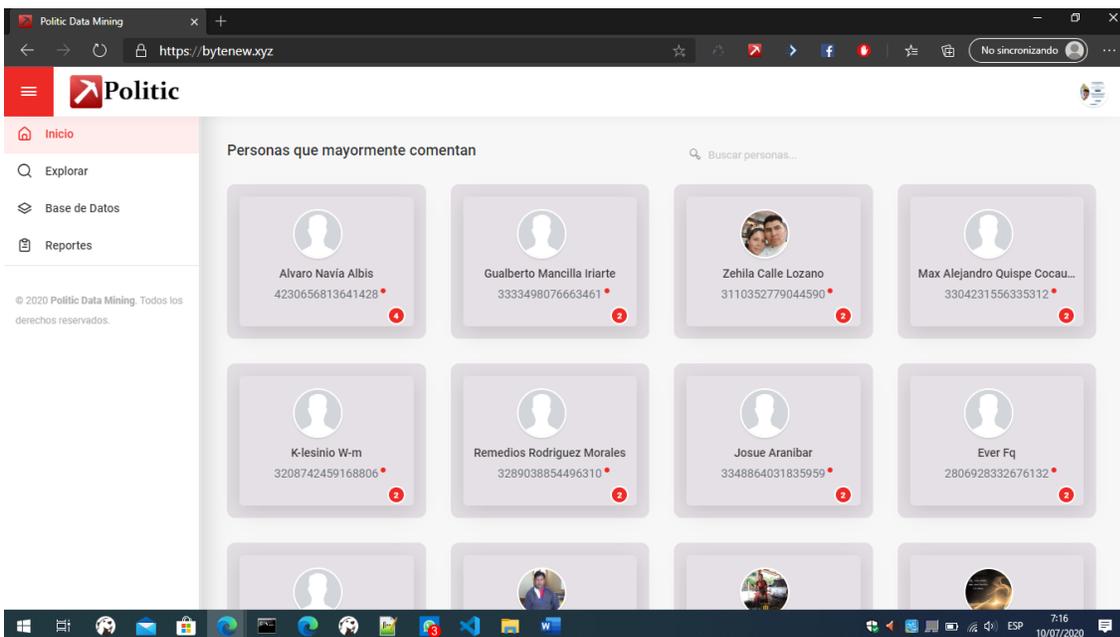
## 2. Se carga el panel del Administrador



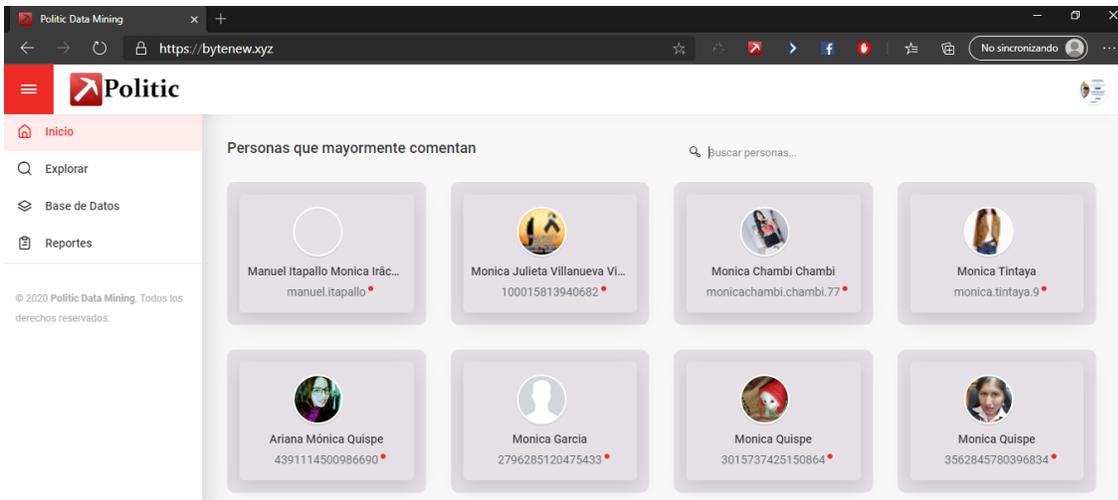
- En la primera columna detallamos la cantidad de personas registradas, publicaciones, comentarios y las búsquedas realizadas, estos datos ya se encuentran en nuestra base de datos.
- En la parte inferior está el listado de las personas que al presionar en la foto de perfil nos redirecciona hacia su página oficial de Facebook.

### 3. Visualizaremos las personas que más comentaron

a. Para ello haremos click en ver mas



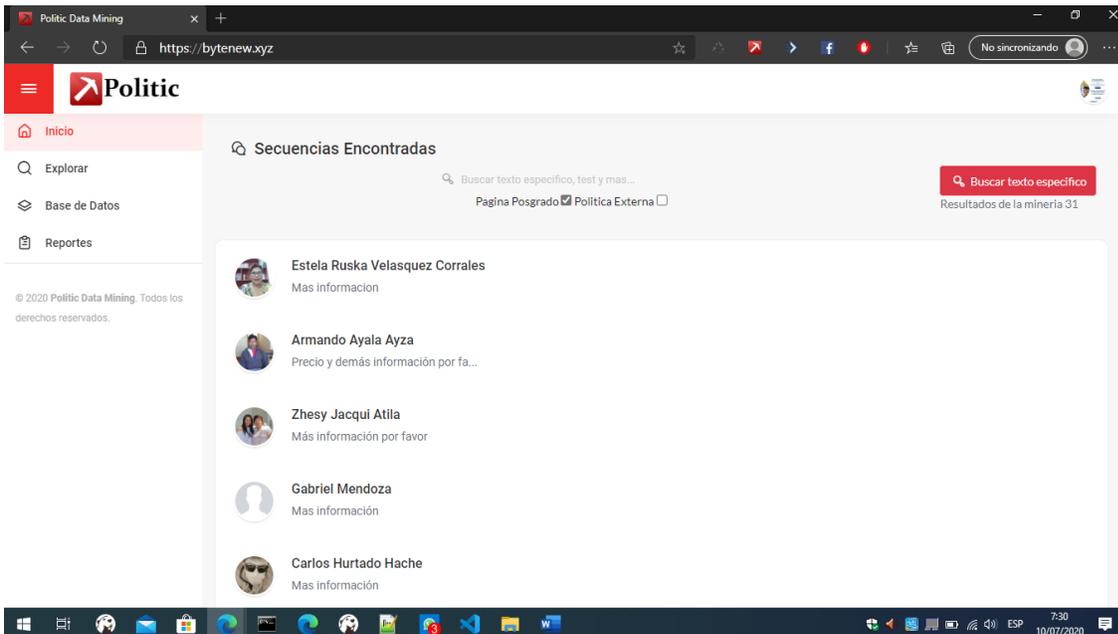
- Nos muestra un listado con los perfiles que más comentaron más en la página de Posgrado.
- También podemos hacer una búsqueda de perfiles



- De la misma forma al presionar en la foto de perfil nos redirecciona hacia su página oficial de Facebook.

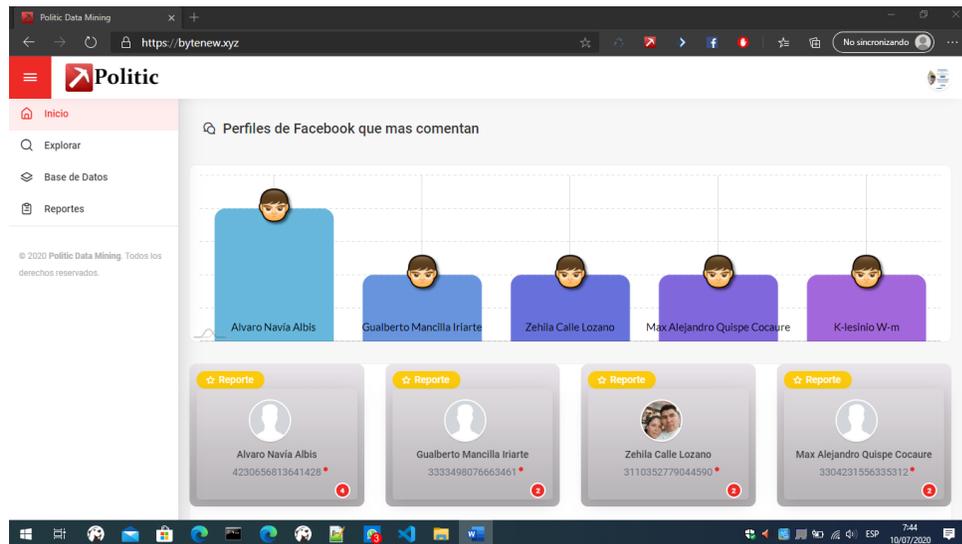
#### 4. Presionamos en Explorar

- Este apartado es para hacer la minería de texto.
- Puede escribir un texto en concreto y presionar **Buscar texto específico**.
- Puede seleccionar si desea que busqué solo en la página de Posgrado o buscar Política Externa



## 5. Presionamos en Reportes

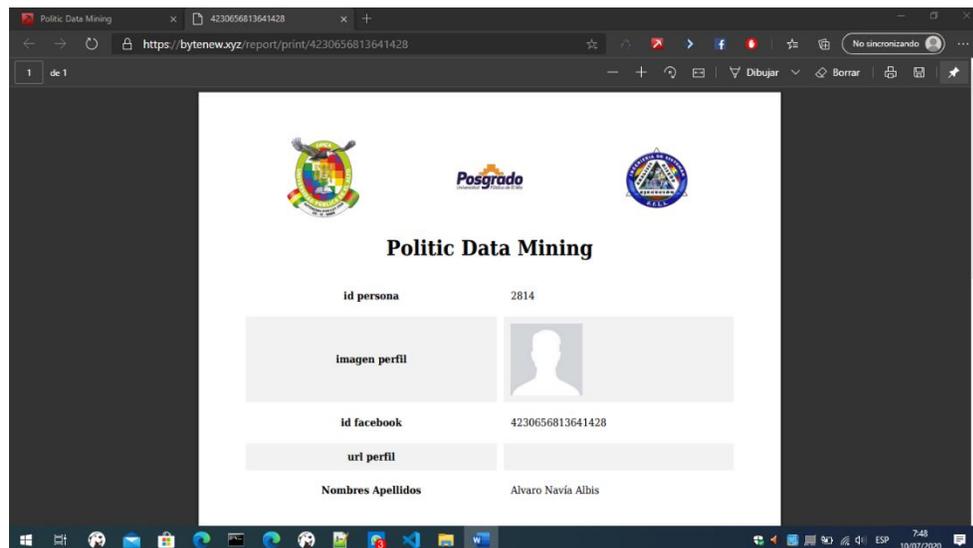
a. Nos muestra un ranking de los perfiles con más interacción en la página.



The screenshot shows a web browser window with the URL <https://bytenew.xyz>. The page title is "Perfiles de Facebook que mas comentan". It displays a horizontal bar chart with five bars of increasing height from left to right. Below each bar is a profile card with a "Reporte" button. The profiles are:

Nombre	ID de Facebook
Alvaro Navía Albis	4230656813641428
Gualberto Mancilla Iriarte	3333498076653461
Zehila Calle Lozano	3110352779044590
Max Alejandro Quispe Cocaure	3304231556335312
K-lesIno W-m	

b. Presionemos en reporte



The screenshot shows a web browser window with the URL <https://bytenew.xyz/report/print/4230656813641428>. The page displays a detailed report for the profile of Alvaro Navía Albis. The report includes the following information:

id persona	2814
imagen perfil	
id facebook	4230656813641428
uri perfil	
Nombres Apellidos	Alvaro Navía Albis

i. Se extrae un reporte directamente para impresión de un perfil.

# **DOCUMENTACIÓN**

El Alto, julio 2020

Señor:

Ing. David Carlos Mamani Quispe

**DIRECTOR DE CARRERA INGENIERIA DE SISTEMAS**

Presente:

**REF.: AVAL DE CONFORMIDAD**

Distinguido Ingeniero:

Mediante la presente tengo a bien comunicarle mi conformidad a la tesis de grado **"MINERIA DE TEXTO DE LA WEB APLICADO EN REDES SOCIALES INFLUENCIADO EN TEMAS POLITICOS"** Caso: **"Facebook"**, que proponen los postulantes Gilda Aquino Castro con CI.: 8378170 LP y Adalid Alanoca Ramírez con CI.: 9874182 LP para su defensa, evaluación correspondiente a la materia Taller de Licenciatura II, de acuerdo a reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de el Alto.

Sin otro particular, saludos a usted muy atentamente



.....  
Ing. Enrique Flores Baltazar

**TUTOR METODOLOGICO**



# Universidad Pública de El Alto

Creada por Ley 2115 del 5 de Septiembre de 2000 y Autónoma por Ley 2556 del 12 de Noviembre de 2003

El Alto, julio de 2020

Señor:

Ing. David Carlos Mamani Quispe  
DIRECTOR DE CARRERA INGENIERIA DE SISTEMAS  
UNIVERSIDAD PÚBLICA DE EL ALTO

Presente:

Ref.: CONFORMIDAD Y AVAL PARA LA TESIS DE GRADO.

De mi mayor consideración:

Mediante la presente reciba un cordial saludo, en calidad de Responsable de Sistemas de Información de la Dirección de Posgrado, de la Universidad Pública de El Alto, luego de haber realizado el seguimiento de la Tesis de Grado "MINERÍA DE TEXTO DE LA WEB APLICADO EN REDES SOCIALES INFLUENCIADO EN TEMAS POLÍTICOS" CASO FACEBOOK, presentado por los estudiantes, Univ. Gilda Aquino Castro con C.I. 8378170 L.P., Univ. Adalid Alanoca Ramirez con C.I. 9874182 L.P., para optar al título de Licenciatura en Ingeniería de Sistemas, le informo mi conformidad y aval respectivo para la defensa pública de la tesis de grado sujeto al reglamento vigente de la Universidad Pública de El Alto.

Sin otro particular me suscribo con las atenciones más distinguidas.

Atentamente:

Ing. Walter Emilio Paco Siles  
RESPONSABLE DE SISTEMAS DE INFORMACIÓN  
DE LA DIRECCIÓN DE POSGRADO



El Alto, julio 2020

Señor:

M.Sc. Ing. Enrique Flores Baltazar

**TUTOR METODOLÓGICO TALLER DE LICENCIATURA II**

Presente:

**REF.: AVAL DE CONFORMIDAD**

Distinguido Ingeniero:

Mediante la presente tengo a bien comunicarle mi **conformidad** a la tesis de grado **“MINERIA DE TEXTO DE LA WEB APLICADO EN REDES SOCIALES INFLUENCIADO EN TEMAS POLITICOS” Caso: “Facebook”**, que proponen los postulantes Univ. Gilda Aquino Castro con CI.: 8378170 LP y Univ. Adalid Alanoca Ramirez con CI.: 9874182 LP para su defensa, evaluación correspondiente a la materia Taller de Licenciatura II de la Carrera Ingeniería de Sistemas U.P.E.A.

Sin otro particular, saludos a usted muy atentamente



.....  
Lic. Fredy Alanoca Coareti  
**TUTOR ESPECIALISTA**

El Alto 9 de julio del 2020

Señor:

M.Sc. Ing. Enrique Flores Baltazar  
**TUTOR METODOLÓGICO TALLER DE LICENCIATURA II**  
**CARRERA INGENIERÍA DE SISTEMAS – U.P.E.A.**

Presente:

**REF.: AVAL DE CONFORMIDAD**

Distinguido Ingeniero:

Mediante la presente tengo a bien comunicarle mi conformidad a la tesis de grado **“MINERIA DE TEXTO DE LA WEB APLICADO EN REDES SOCIALES INFLUENCIADO EN TEMAS POLITICOS”** Caso: **“Facebook”**, que proponen los postulantes Gilda Aquino Castro con CI.: 8378170 LP y Adalid Alanoca Ramirez con CI.: 9874182 LP para su defensa, evaluación correspondiente a la materia Taller de Licenciatura II, de acuerdo a reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de el Alto.

Sin otro particular, saludos a usted muy atentamente

  
.....  
Lic. Katya Mariela Perez Martinez  
**TUTOR REVISOR**