UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO

FACULTAD CIENCIAS DE LA INGENIERÍA

CARRERA DE INGENIERÍA EN SISTEMAS

Proyecto de Investigación previo a la obtención del título de Ingeniero en Sistemas.

Título del Proyecto de Investigación:

"SISTEMA DE ANÁLISIS ESTADÍSTICO PARA UN BANCO DE DATOS ENTOMOLÓGICO"

Autor:

Kevin Manuel Onofre Wong

Director del Proyecto de Investigación:

Ing. Washington Alberto Chiriboga Casanova, MSc

Quevedo - Los Ríos – Ecuador.

2019

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

Yo, **Kevin Manuel Onofre Wong**, declaro que la investigación aquí descrita es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que he consultado las referencias bibliográficas que se incluyen en este documento.

La Universidad Técnica Estatal de Quevedo, puede hacer uso de los derechos correspondientes a este documento, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Kevin Manuel Onofre Wong

C.C. # 0941629388

CERTIFICACIÓN DE CULMINACIÓN DEL PROYECTO DE INVESTIGACIÓN

El suscrito, Ing. Washington Alberto Chiriboga Casanova, Docente de la Universidad Técnica Estatal de Quevedo, certifica que el estudiante Kevin Manuel Onofre Wong, realizó el Proyecto de Investigación de grado titulado **"SISTEMA DE ANÁLISIS ESTADÍSTICO PARA UN BANCO DE DATOS ENTOMOLÓGICO"**, previo a la obtención del título de Ingeniero en Sistemas, bajo mi dirección, habiendo cumplido con las disposiciones reglamentarias establecidas para el efecto.

Ing. Washington Alberto Chiriboga Casanova, MSc
DIRECTOR DEL PROYECTO DE INVESTIGACIÓN

CERTIFICADO DEL REPORTE DE LA HERRAMIENTA DE PREVENCIÓN DE COINCIDENCIA Y/O PLAGIO ACADÉMICO

Ing. Washington Alberto Chiriboga Casanova, MSc en calidad de Director de Proyecto de Investigación titulado **"SISTEMA DE ANÁLISIS ESTADÍSTICO PARA UN BANCO DE DATOS ENTOMOLÓGICO",** me permite manifestar a usted y por intermedio del Consejo Académico lo siguiente:

Que, el señor **Kevin Manuel Onofre Wong,** egresado de la Facultad de Ciencias de la Ingeniería, carrera Ingeniería en Sistemas, ha cumplido con las correcciones pertinentes, e ingresado su Proyecto de Investigación al sistema URKUND, tengo bien certificar la siguiente información sobre el informe del sistema anti plagio con un porcentaje de 3%.

URKUND

Document	Tesis-Wong.docx (D44479108)
Submitted	2018-11-23 22:07 (-05:00)
Submitted by	onofrekevin@gmail.com
Receiver	wchiriboga.uteq@analysis.urkund.com
Message	Tesis Kevin Onofre Show full message
	3% of this approx. 44 pages long document consists of text present in 7 sources.

Ing. Washington Alberto Chiriboga Casanova, MSc

DIRECTOR DEL PROYECTO DE INVESTIGACIÓN

UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO

FACULTAD CIENCIAS DE LA INGENIERÍA

CARRERA DE INGENIERIA EN SISTEMAS

PROYECTO DE INVESTIGACIÓN

Título:

"Sistema de Análisis Estadístico para un Banco de Datos Entomológico"

Presentado al Consejo Académico como requisito previo a la obtención del título de Ingeniero en Sistemas.

Aprobado por:

PRESIDENTE DEL TRIBUNAL

Ing. Ariosto Vicuña Pino, MSc.

MIEMBRO DEL TRIBUNAL

Ing. Elías Portilla Olvera, MSc.

MIEMBRO DEL TRIBUNAL

Ing. Eduardo Samaniego Mena, MSc.

QUEVEDO – LOS RIOS – ECUADOR

2019

AGRADECIMIENTO

A mi madre Clavel Wong, que sin ella no lo hubiera logrado.

Al Ing. Washington Chiriboga, director tesis por la guía y constante paciencia.

A la gama de docentes de la universidad, en especial a:

- Ing. Gleiston Guerrero Ulloa
- Ing. Ariosto Vicuña Pino
- Ing. Iván Jaramillo Chuqui
- Ing. Elías Portilla Olvera
- Ing. Eduardo Samaniego Mena
- Dr. Raúl Hernández Palacios
- Ing. Byron Oviedo Bayas
- Ing. Andrea Zúñiga Paredes

A Joselyn Averos por el apoyo constante y los consejos que fueron de gran ayuda.

A Ricardo Villarroel por el apoyo, Christian Guanoquiza y José Mosquera.

DEDICATORIA

Dedico este proyecto a mi madre Clavel Wong.

RESUMEN EJECUTIVO Y PALABRAS CLAVE

En la Universidad Técnica Estatal de Quevedo (UTEQ) se han desarrollado varios estudios sobre los insectos de la zona, especialmente sobre el área de investigación de la silvicultura, estos procesos de estudio se llevan a cabo de manera individual por cada investigador, los mismos que utilizan diversas maneras de recolectar datos lo cual genera archivos con abundantes registros. Con el objetivo de contar con un banco de datos donde consten todos los estudios realizados, se ha desarrollado una aplicación web y móvil que ofrece la gestión de datos entomológicos para análisis estadístico. Se diseñó una base datos que soporta las variables requeridas en investigaciones entomológicas. El desarrollo se ha llevado a cabo dentro de la metodología ágil "Proceso Unificado Racional" (RUP), por su alto nivel de abstracción lo que quiere decir que motiva el uso de conceptos reutilizables tales como frameworks. Se utilizó Microsoft SQL Server Analysis Services 13.0.4 como herramienta de minería de datos, porque proporciona bibliotecas de cliente (AMO 10.0.0, ADOMD.NET 10.5.0) las cuales se utilizaron para integrar en la aplicación web ASP.NET la creación de base de datos de análisis y modelos de minería de datos. La construcción de la aplicación se llevó a cabo bajo el framework .Net 4.5.2 de Microsoft, como lenguaje de programación C# 5.0, en un ambiente web ASP.NET usando páginas web forms, acceso a datos mediante Entity Framework 6.1.3 y como motor de base de datos SQL Server 13.0.4. Por otra parte, se empleó Mantnet.numers 4.5.1 para cálculos de correlación, la geolocalización se implementó con Opencagedata 1.0.0 y la librería JavaScript leaflet 1.3.4 para mapas interactivos.

Palabras clave: Geolocalización, estadística, minería, entomológico, dato enriquecido.

ABSTRACT AND KEYWORDS

In the State Technical University of Quevedo (UTEQ) several studies have been developed on the insects of the area, especially on the area of forestry research, these study processes are carried out individually by each researcher, the same as They use different ways to collect data which generates files with abundant records. In order to have a data bank where all the studies carried out, a web and mobile application has been developed that offers entomological data management for statistical analysis. A database was designed that supports the variables required in entomological research. The development has been carried out within the agile methodology "Unified Rational Process" (RUP), for its high level of abstraction which means that it motivates the use of reusable concepts such as frameworks. Microsoft SQL Server Analysis Services 13.0.4 was used as a data mining tool, because it provides client libraries (AMO 10.0.0, ADOMD.NET 10.5.0) which were used to integrate in the ASP.NET web application the creation of analysis databases and models of data. mining. The construction of the application was carried out under Microsoft's .Net framework 4.5.2, as a C # 5.0 programming language, in an ASP.NET web environment using form web pages, data access through the Entity Framework 6.1.3 and as a SQL Server 13.0.4 database engine. On the other hand, Mantnet.numers 4.5.1 was used for correlation calculations, geolocation was implemented with Opencagedata 1.0.0 and leaflet 1.3.4 JavaScript library for interactive maps.

Keywords: Geolocation, statistics, mining, entomological, data enriched.

TABLA DE CONTENIDO

Portada i
Declaración de Autoría y Cesión de Derechos ii
Certificación de Culminación del Proyecto de Investigación iii
Certificado del Reporte de la Herramienta de Prevención de Coincidencia Y/O Plagio Académico iv
Presentado a la Comisión Académica como requisito previo a la obtención del Título de Ingeniería en Sistemas v
Agradecimiento vi
Dedicatoria vii
Resumen Ejecutivo y Palabras Clave viii
Abstract and Keywords ix
Código Dublin xxi

INTRODUCCIÓN 1
CAPÍTULO I 3
CONTEXTUALIZACIÓN DE LA INVESTIGACIÓN 3
1.1. Problematización de investigación 4
1.1.1. Planteamiento del problema 4
1.1.2. Formulación del problema 5
1.1.3. Sistematización 5
1.2. Objetivos 5
1.2.1. Objetivo general 5
1.2.2. Objetivos específicos 5
1.3. Justificación 6
CAPÍTULO II 7
FUNDAMENTACIÓN TEÓRICA DE LA INVESTIGACIÓN 7
2.1. Marco conceptual 8
2.1.1. Estadística 8
2.1.2. Minería de datos 8
2.1.3. Microsoft Analysis Services 9
2.1.4. Proceso de minería datos 9
2.1.5. Conceptos clave para formular e implementar un escenario de minería 10
2.1.5.1. Atributo. 10
2.1.5.2. Caso DMX 11

2.1.5.3. Clave…. 11
2.1.5.4. Usos de los atributos 11
2.1.5.5. Origen de datos 12
2.1.5.6. Vista fuente de datos 12
2.1.5.7. Estructura de minería 12
2.1.5.8. Columnas discretizadas 14
2.1.5.9. Modelo minería 15
2.1.5.10. Particionamiento en entrenamiento y conjunto de pruebas 16
2.1.6. Algoritmos Analysis Services 17
2.1.6.1. Regresión Lineal 19
2.1.6.2. Árbol de decisión 20
2.1.6.3. Naive Bayes 21
2.1.6.4. Agrupamiento 22
2.1.7. Visor de árbol de contenido genérico de Microsoft 23
2.1.8. Metodología de desarrollo de software Proceso Unificado Racional (RUP) 30
2.1.8.1. Fase de inicio 31
2.1.8.2. Fase de elaboración 32
2.1.8.3. Fase de construcción 33
2.1.8.4. Fase de transición 34
2.1.9. Aplicación Web 34
2.1.10. Arquitectura Web Cliente/Servidor 35
2.1.11. Arquitectura REST 35
2.1.12. Herramientas de desarrollo 36
2.1.12.1. IDE Visual Studio 36
2.1.12.2. Web ASP.NET 36
2.1.12.3. ASP.NET Master Pages 37
2.1.12.4. LINQ…. 37
2.1.12.5. Newtonsoft 38
2.1.12.6. Mantnet.Numerics 38
2.1.12.7. OpenCage Geocoder 39
2.1.12.8. Map thunderforest 39
2.1.12.9. SQL Server y Tipos (Datos) 40
2.1.12.10. Arquitectura tecnológica servidor SQL Analysis Services 43
2.2. Marco referencial 46
2.2.1. Clave taxonómica de identificación virtual de la colección de insectos… acuáticos del departamento de biología de la Universidad Pedagógica Nacional 46
2.2.2. Pl@ntNet 47

2.2.3. Una nueva herramienta de exploración y generación de informes de minería ..... de datos basada en web para los que toman decisiones..... 48
2.2.4. Darwin Core: Estándar para la gestión de datos biológicos primarios en la UTN 49
2.2.5. Diseño e implementación de una solución de almacenamiento de datos basada ..... en la web para el análisis de costos 49
2.2.6. GBIF Global Biodiversity Information Facility 50
CAPÍTULO III 52
METODOLOGÍA DE LA INVESTIGACIÓN..... 52
3.1. Localización 53
3.2. Tipos de Investigación 53
3.2.1. Bibliográfica..... 53
3.2.2. Descriptiva 53
3.3. Métodos de investigación..... 54
3.3.1. Bibliográfica-documental..... 54
3.3.2. Analítica 54
3.3.3. Método de desarrollo: Proceso unificado racional (RUP)..... 55
3.4. Fuentes de recopilación de información..... 56
3.4.1. Fuentes primarias 56
3.4.2. Fuentes secundarias..... 56
3.5. Recursos y Materiales 57
CAPÍTULO IV 60
RESULTADOS Y DISCUSIÓN..... 60
4.1. Resultados de la metodología RUP 61
4.1.1. Fase inicial 61
4.1.1.1. Modelo conceptual o dominio..... 62
4.1.1.2. Caso uso usuario (Investigador)..... 63
4.1.2. Fase elaboración..... 64
4.1.2.1. Modelo de datos 64
4.1.2.2. Clases del sistema..... 67
4.1.2.3. Casos de uso extendido 70
4.1.2.4. Diagrama de secuencias 95
4.1.2.5. Arquitectura del sistema..... 96
4.1.2.6. Patrón de diseño 96
4.1.2.7. Arquitectura prototipo del hardware sistema 99
4.1.3. Fase construcción 99
4.1.3.1. Diagrama de despliegue 100

4.1.3.2. Entorno tecnológico 100
4.1.3.3. Análisis Sumarización 103
4.1.3.4. Análisis Correlación 105
4.1.3.5. Análisis de minería SSAS 108
4.1.4. Fase transición 115
4.2. Discusión 116
CAPITULO V 118
CONCLUSIONES Y RECOMENDACIONES 118
5.1. Conclusiones 119
5.2. Recomendaciones 120
CAPITULO VI 121
BIBLIOGRAFÍA 121
6.1. Bibliografía 122
CAPITULO VII 125
ANEXOS 125
7.1. Anexo 1: Matriz del Problema 126
7.2. Anexo 2: Modelo de datos 127
7.2.1. Diccionario de datos 127
7.3. Anexo 3: Control de versiones 135
7.3.1. Control de versiones bitbucket 135
7.4. Anexo 4: Registros de investigaciones entomológicas 136
7.4.1. Diferentes registros de datos en investigaciones entomológicas realizadas en la UTEQ 136
7.5. Anexo 5: Códigos fuente PIentomología 139
7.6. Anexo 6: Documentación PIentomología 145
7.6.1. Manuales de Usuario 145

ÍNDICE DE TABLAS

Tabla 1. Tipos de datos y contenido admitido por algoritmos SSAS 13
Tabla 2. Tipos de contenido y su detalle 13
Tabla 3. Propiedades para crear conjunto de datos de prueba 16
Tabla 4. Detalle de los algoritmos permitidos por SSAS 17
Tabla 5. Identificador de nodo, nombre, título y descripción. 24
Tabla 6. Tipos de valores para Value Type 27
Tabla 7. Lista de tipos de nodos de contenido minería de datos. 28
Tabla 8. Categorías de tipos de datos de SQL SERVER 40
Tabla 9. Tipos de datos por categoría. 41
Tabla 10. Tipos numéricos aproximados. 41
Tabla 11. Tipos fecha y hora. 41
Tabla 12. Tipos cadena de caracteres. 41
Tabla 13. Otros tipos de datos. 42
Tabla 14. Detalle de funciones de agregado 43
Tabla 15. Recursos humanos. 57
Tabla 16. Recursos hardware. 57
Tabla 17. Recursos software. 58
Tabla 18. Presupuesto total. 59
Tabla 19. Caso de uso iniciar sesión. 70
Tabla 20. Caso de uso menú de usuario. 72
Tabla 21. Caso de uso crear proyecto. 73
Tabla 22. Caso de uso crear variable. 74
Tabla 23. Caso de uso descargar observaciones. 76
Tabla 24. Caso de uso registrar observación. 78
Tabla 25. Caso de uso crear análisis de correlación. 81
Tabla 26. Caso de uso ejecutar análisis correlación. 84
Tabla 27. Crear análisis de sumarización. 86
Tabla 28. Caso de uso ejecutar análisis sumarización. 88
Tabla 29. Caso de uso crear modelo de minería de datos. 90
Tabla 30. Caso de uso ejecutar análisis de minería. 93
Tabla 31. Descripción de variables observación de insecto. 116
Tabla 32. Lista de tablas de base de datos PIentomología. 127
Tabla 33. Diccionario de datos: tbrol 128
Tabla 34. Diccionario de datos: tbusuario 129
Tabla 35. Diccionario de datos: tbproyectoinv. 129
Tabla 36. Diccionario de datos: tbobservacion. 129
Tabla 37. Diccionario de datos: tbdatosvariable. 130
Tabla 38. Diccionario de datos: tbvariable. 130
Tabla 39. Diccionario de datos: tbtipovariable. 131
Tabla 40. Diccionario de datos: tbopcionvariable. 131
Tabla 41. Diccionario de datos: tbtipofuente. 131
Tabla 42. Diccionario de datos: tbsumarizacion. 131
Tabla 43. Diccionario de datos: tbconfiguracionsumar. 132
Tabla 44. Diccionario de datos: tbtiposumarizacion. 132
Tabla 45. Diccionario de datos: tbtipousovariable. 132

Tabla 46. Diccionario de datos: tbmodelomineria. 133
Tabla 47. Diccionario de datos: tbtecnicamineriadatos..... 133
Tabla 48. Diccionario de datos: tbcorrelacionbivariada..... 133
Tabla 49. Diccionario de datos: tbconfiguracioncorrelacion. 134
Tabla 50. Diccionario de datos: tbtipocorrelacion. 134
Tabla 51. Registro de datos de investigación fluctuación. 136
Tabla 52. Registro de datos de investigación diversidad. 137
Tabla 53. Registros de datos de investigación diversidad scolytinae..... 138
Tabla 54. Parámetros de variables..... 147
Tabla 55. Detalle de parámetros de configuración sumarización..... 155

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Proceso de minería de datos SSAS 10
Ilustración 2. Visor de árbol de contenido genérico SSAS. 23
Ilustración 3 . Pila de componentes de .Net Framework 37
Ilustración 4.API REST Para localizar punto por medio de coordenadas 39
Ilustración 5. Vista de paisaje localidad Quevedo por Map Thunderforest 40
Ilustración 6. Arquitectura de comunicación de Analysis Services. 44
Ilustración 7. Jerarquía de objetos de AMO 45
Ilustración 8. Principales objetos de minería de datos de ADOMD.NET 45
Ilustración 9. Sitio Web Pl@ntNet 47
Ilustración 10. Sitio web Pl@ntNet página ubicación de insecto Scolytinae. 48
Ilustración 11. Página Index Gbif.org 50
Ilustración 12. Página GBIF de búsqueda registros de presencia. 50
Ilustración 13. Detalle módulos y componentes de ASMINER. 51
Ilustración 14. Arquitectura Tecnológica de la plataforma web para la gestión datos bilógicos. 51
Ilustración 15. Ubicación de UTEQ por Api Google Map 53
Ilustración 16. Fases de RUP Modelo iterativo muestra cómo el proceso se estructura a lo largo de dos dimensiones 56
Ilustración 17. Modelo de dominio del problema. 62
Ilustración 18. Caso de uso usuario investigador. 63
Ilustración 19. Diagrama entidad relación. 64
Ilustración 20. Parte I. Base de datos Física de PIentomología 65
Ilustración 21. Parte II. Base de datos Física de PIentomología. 66
Ilustración 22. Clase usuario relacionada con rol 67
Ilustración 23. Clase proyecto relacionada con usuario y variable. 67
Ilustración 24. Clase variable relacionada con opción variable, tipo variable, tipo fuente. 67
Ilustración 25. Clase datos variable relacionada con observación y variable. 68
Ilustración 26. Clase sumarización relacionada con configuración. 68
Ilustración 27. Clase correlación. 68
Ilustración 28. Clase modelo de minería. 69
Ilustración 29. Clases para minería de datos, base de datos, fuente de datos, vista de fuentes datos. 69
Ilustración 30. Clases de minería de datos, estructura de minería, columna estructura de minería. 69
Ilustración 31. Clases de minería de datos, modelo de minería, columna modelo de minería. 70
Ilustración 32. Diagrama de secuencia inicio sesión. 95
Ilustración 33. Diagrama de secuencia crear análisis de correlación. 95
Ilustración 34. Diagrama de secuencia crear análisis de sumarización. 96
Ilustración 35. Patrón de diseño de la aplicación móvil PIentomología. 97
Ilustración 36. Patrón de diseño de la aplicación web PIentomología. 97
Ilustración 37. Arquitectura MVP con ThirtyInch. 98
Ilustración 38. Arquitectura prototipo del hardware sistema 99
Ilustración 39. Diagrama de despliegue PIentomología. 100
Ilustración 40. Solicitud Get login en aplicación Postman. 101
Ilustración 41. Creación de variables dinámicas y mostrar en pantalla. 103
Ilustración 42. Script para calcular la sumarización. 104
Ilustración 43. Código usado para generar el archivo xls de resultado de sumarización. 105
Ilustración 44. Código crear variables dinámicas y mostrar en pantalla. 105
Ilustración 45. Script obtener datos del proyecto de investigación. 106
Ilustración 46. Código fuente para calcular Correlaciones. 107

Ilustración 47. Código fuente para convertir tabla a etiquetas HTML. 107
Ilustración 48. Configuración del servidor SSAS. 108
Ilustración 49. Configuración en modo de Modelo de minería. 108
Ilustración 50. Servidores de base de datos de PIentomología. 108
Ilustración 51. Pantalla de Visual Studio de la solución WebControls. 109
Ilustración 52. Archivos de instalación del componente Web. 109
Ilustración 53. Archivos en el servidor. 109
Ilustración 54. Bibliotecas de PIentomología. 110
Ilustración 55. Código fuente para crear variables dinámicas y mostrar en pantalla. 110
Ilustración 56. Código fuente de la clase control de objetos SSSAS. 111
Ilustración 57. Script para crear tabla de datos del proyecto. 112
Ilustración 58. Código para crear objetos para la minería de datos. 113
Ilustración 59. Objetos en el servidor SSAS. 113
Ilustración 60. Código fuente para ejecutar modelo de minería. 114
Ilustración 61. Código fuente de los componentes web de minería. 114
Ilustración 62. Código fuente de Proveedor datos Adomd 114
Ilustración 63. Matriz de relación problema-objetivos 126
Ilustración 64.Repositorio bitbucket, para control de versiones de sistema PIentomología. 135
Ilustración 65. Código fuente creación de objetos de SSAS. 139
Ilustración 66. Script SQL Server organizar datos a tablas. 140
Ilustración 67. Código fuente de clase POCO. 141
Ilustración 68. Código fuente de Clase DAO. 142
Ilustración 69. Código fuente clase Controlador. 143
Ilustración 70. Código fuente página ASP.NET. 143
Ilustración 71. Código para ingresar datos de observaciones. 144
Ilustración 72. Página inicio de sesión. 146
Ilustración 73. Página principal. 146
Ilustración 74. Página lista de proyectos. 147
Ilustración 75. Página crear proyecto. 147
Ilustración 76. Página crear variable. 148
Ilustración 77. Página principal opción observaciones. 149
Ilustración 78. Página lista de proyectos. 149
Ilustración 79. Página lista de observaciones. 149
Ilustración 80. Página agregar observaciones. 150
Ilustración 81. Página agregar observación con foto. 150
Ilustración 82. Página agregar observación, cargar foto. 151
Ilustración 83. Página principal, opción analizar. 151
Ilustración 84. Página lista de proyecto, opción correlación. 152
Ilustración 85. Página crear lista de análisis de correlación. 152
Ilustración 86. Página crear análisis de correlación. 152
Ilustración 87. Página crear análisis de correlación. 153
Ilustración 88. Notificación al configurar nueva correlación. 153
Ilustración 89. Notificación de correlación creada. 153
Ilustración 90. Tabla de correlación. 153
Ilustración 91. Página principal, opción analizar de sumarización. 154
Ilustración 92. Página lista de proyectos, opción sumarización. 154
Ilustración 93. Página lista de análisis de correlación. 154
Ilustración 94. Página crear sumarización, parámetros. 156
Ilustración 95. Página crear sumarización, tipos de sumarización. 156
Ilustración 96. Página sumarización, archivo Excel. 156
Ilustración 97. Api PIentomología. Inicio de sesión. 157

Ilustración 98. Menú de API PIentomología. 157
Ilustración 99. Api PIentomología, lista de proyectos. 158
Ilustración 100. API PIentomología, lista de observaciones. 158
Ilustración 101. API PIentomología, formulario crear observación. 158

ÍNDICE DE SIGLAS Y ACRÓNIMOS

Sigla	Significado
	A
ADO.NET:	Componentes de software para acceder a datos y servicios de datos.
ADOMD:	Proveedor de datos de Microsoft .NET Framework diseñado para comunicarse con Microsoft SQL Server Analysis Services.
AJAX	Asincrónico JavaScript And XML
AMO	Administrador de objetos en Analysis Services
API:	Interfaz de programación de aplicaciones.
ASP	Páginas activas de servidor
ASMINER	Herramienta para la generación de informes y exploración de minería de datos basado en la web
	C
CLR:	Entorno en tiempo de ejecución de lenguaje común
CSS:	Hojas de estilo en cascada.
CIL	Lenguaje intermedio común
CRUD:	Crear, leer, actualizar, eliminar
CGI:	Interfaz de entrada común
CSV:	Valores separados por comas
CIRAD	Investigación agrícola para el desarrollo
	D
DAO:	Objetos de acceso a datos
DLL	Biblioteca de enlace dinámico
DMX:	Extensiones de minería de datos
DSS:	Sistemas de soporte a la decisión
	E
EIS:	Sistema de información ejecutiva
EXEC:	Archivo ejecutable para Windows
	G
GIT	Software de control de versiones
GPS	Sistema de posicionamiento global
	H
HTML:	Lenguaje de marcado de hipertexto
HTTP:	Protocolo de transferencia de hipertexto
	I
IRD	Instituto de Investigaciones para el Desarrollo
IP:	Protocolo de internet
IIS:	Servicios de información de internet
INRA	Instituto nacional de investigaciones agropecuarias
	J
JAVA:	Lenguaje de programación de propósito general, orientado a objetos.
JAVASCRIPTS:	Lenguaje de programación interpretado.
JSON:	Notación de objetos de JavaScript
	L
LAN	Red de área local
LINQ	Lenguaje integrado de consultas

	M
MIT X11:	Licencia de software del Massachusetts Institute of Technology
MVC:	Modelo vista controlador
MVP:	Modelo vista presentador
MSIL:	Lenguaje intermedio de Microsoft
	N
NET:	Framework de Microsoft
	O
ORM:	Mapeo objeto-relacional
OLAP:	Procesamiento analítico en línea
OLE DB:	Vinculación de objetos para bases de datos
	P
PHP:	Preprocesador de hipertexto
POCO:	Objeto CLR antiguo simple
POJO	Simple antiguo objeto java
	R
RUP:	Proceso racional unificado
RDBMS:	Sistema de gestión de bases de datos relacionales
REST:	Transferencia de estado representacional
	S
SCRIPT	Archivo de instrucciones, escrito en código de programación
SIGBIO:	Sistema web para la gestión de datos biológicos
SCRUM	Marcos de desarrollo ágiles
SO	Sistema operativo
SOAP:	Protocolo de acceso simple a objetos
SQL:	Lenguaje de consulta estructurado
SSAS:	Microsoft SQL Analysis Services
	T
TSQL	Lenguaje de consulta estructurada de transacciones
TCP:	Protocolo de control de transmisión
	U
UI	Interfaz de usuario
UTEQ	Universidad Técnica Estatal de Quevedo
UTN	Universidad Técnica del Norte
	W
WSDL:	Lenguaje de descripción de servicios web
	X
XLS:	Extensión de archivo Excel
XML:	Lenguaje de marcado extensible

CÓDIGO DUBLIN

Título:	Sistema de Análisis Estadístico para un Banco de Datos Entomológico.				
Autor:	Kevin Manuel Onofre Wong				
Palabras claves:	Geolocalización	Estadística	Minería	Entomológico	Dato enriquecido
Fecha de publicación:					
Editorial:	Quevedo: UTEQ, 2019				
Resumen:	En la Universidad Técnica Estatal de Quevedo (UTEQ) se han desarrollado varios estudios sobre los insectos de la zona, especialmente sobre el área de investigación de la silvicultura, estos procesos de estudio se llevan a cabo de manera individual por cada investigador, los mismos que utilizan diversas maneras de recolectar datos lo cual genera archivos con abundantes registros. Con el objetivo de contar con un banco de datos donde consten todos los estudios realizados, se ha desarrollado una aplicación web y móvil que ofrece la gestión de datos entomológicos para análisis estadístico. Se diseñó una base datos que soporta las variables requeridas en investigaciones entomológicas. El desarrollo se ha llevado a cabo dentro de la metodología ágil "Proceso Unificado Racional" (RUP), por su alto nivel de abstracción lo que quiere decir que motiva el uso de conceptos reutilizables tales como frameworks. (…)				
Abstract:	In the State Technical University of Quevedo (UTEQ) several studies have been developed on the insects of the area, especially on the area of forestry research, these study processes are carried out individually by each researcher, the same as They use different ways to collect data which generates files with abundant records. In order to have a data bank where all the studies carried out, a web and mobile application has been developed that offers entomological data management for statistical analysis. A database was designed that supports the variables required in entomological research. The development has been carried out within the agile methodology "Unified Rational Process" (RUP), for its high level of abstraction which means that it motivates the use of reusable concepts such as frameworks. (…).				
Descripción:					
URI:					

INTRODUCCIÓN

En la Universidad Técnica Estatal de Quevedo (UTEQ) se han desarrollado varios estudios de investigación sobre la diversidad y caracterización de especies biológicas de distintos grupos taxonómicos, entre ellos de insectos en diferentes ecosistemas forestales. Los datos obtenidos en estos estudios permanecen en poder de los investigadores que realizaron las investigaciones. Es decir, una vez culminada la investigación los datos no se comparten por diversos aspectos como por ejemplo por falta de un repositorio.

Para la gestión de los datos (almacenamiento, procesamiento y obtención de resultados) de las investigaciones realizadas se usaron libros de Excel. Bajo esta forma de trabajo los datos fueron almacenados y se procesó las fórmulas y gráficas necesarias para el estudio. Esta forma de trabajo limita el análisis estadístico a formas descriptivas de los resultados. Esto dificulta ampliar la posibilidad de aplicar otras técnicas estadísticas que proporcionen una visión más completa sobre los objetivos de las investigaciones que se realizan.

Para solucionar este problema se propone contar con un sistema distribuido (web y móvil) con una base de datos centralizada que facilite la recolección de datos como registrar la información de contexto básica de la muestra obtenida, ubicación geográfica, fotografías de la localidad, temperatura, clima, entre otros datos que se obtienen en las observaciones de campo. También se registrará el análisis taxonómico correspondiente a la muestra y mediante un conjunto de herramientas estadísticas de minería de datos se dará paso búsqueda de conocimientos ocultos en el conjunto de datos que permita conocer los resultados entomológicos parametrizados con estándares de publicación e intercambio de información biológica.

Según la investigación preliminar, el sistema distribuido sería conveniente desarrollarlo utilizando la arquitectura Cliente/Servidor (C/S) con un patrón de diseño modelo, vista, controlador (MVC). Esto es pertinente por su idea principal de realizar una división entre objetos de dominio que modelan la forma en que se perciben los objetos del mundo real y objetos de presentación que son los elementos interfaz de usuarios que vemos en la pantalla [1]. Para la aplicación móvil de igual modo usará MVP por su arquitectura basada en

componentes que se enfoca en la descomposición del diseño en componentes funcionales o lógicos que expongan interfaces de comunicación bien definidas [2].

CAPÍTULO I

CONTEXTUALIZACIÓN DE LA INVESTIGACIÓN

1.1. Problematización de investigación

1.1.1. Planteamiento del problema

En la Universidad Técnica Estatal de Quevedo se han desarrollado varios estudios sobre los insectos de la zona, especialmente sobre el área de investigación de la silvicultura, motivados por la necesidad de determinar su efecto en la producción forestal en el Ecuador y en especial en la zona de influencia de la Universidad. Sin embargo, los datos de dichos estudios se mantienen de carácter privado; es decir, una vez que se realizan y se publican los resultados del estudio, todos los datos (guardados en Excel) permanecen con el investigador que realizó el estudio y normalmente no se vuelven a utilizar perdiéndose buena parte del valor de las investigaciones realizadas.

Cada estudio de investigación sobre los insectos de la zona presenta problemas en la toma de datos debido a las dificultades de logística; normalmente las muestras son llevadas desde la zona de recolección al sitio de análisis y valoración de datos, sin que se tenga un registro especifico georreferenciado de donde fue recolectada una muestra entomológica, lo que impide hacer mapeos poblacionales más exactos y realizar estudios de migración o de afectación. Entre las variables relevantes, como se indicó anteriormente, consta la ubicación geográfica y la evidencia fotográfica del sitio donde se produjo la recolección de la muestra. Esto significa el uso de alguna tecnología que provea estos datos de manera directa y económica. Además, de otros datos que corresponden al momento de la recolección, como los climáticos y características del ambiente. Sería entonces conveniente que al momento de recolectarlos estos datos (directos y complementarios) sean almacenados en el sitio y en el momento donde ocurre, para luego completar la información en los laboratorios donde se analiza más detalladamente la muestra.

Además, el procesamiento de los datos depende de la posibilidad de acceder a herramientas estadísticas según las capacidades del investigador. Generalmente se utilizan hojas electrónicas en las que se elaboran las fórmulas y las gráficas, limitando los análisis a estadísticas descriptivas y en pocos casos a análisis inferencial básico. Sería importante proveer a los investigadores de diferentes herramientas estadísticas y de minería que les

permitan obtener resultados más amplios y diversos, los cuales pueden conducir a resultados más relevantes.

1.1.2. Formulación del problema

- ¿Es posible gestionar (recolectar, recuperar y procesar) los datos entomológicos recolectados por los investigadores de la UTEQ?

1.1.3. Sistematización

- ¿Qué diseño físico de base de datos se acopla de manera adecuada a la información requerida para estudios entomológicos?
- ¿Se puede disponer de datos de investigaciones entomológicas realizadas anteriormente por investigadores de la UTEQ?
- ¿La implementación de un servicio de análisis (modelos, formulas estadísticas) mejorará la obtención de resultado y conclusiones de los estudios?

1.2. Objetivos

1.2.1. Objetivo general

- Desarrollar una aplicación Web y Móvil que permita la gestión automatizada de datos entomológicos recolectados por la UTEQ para análisis estadístico.

1.2.2. Objetivos específicos

- Crear el modelo físico que soporte las variables requeridas en investigaciones entomológicas en estudios transversales y longitudinales.
- Disponer de datos de investigaciones entomológicas realizadas mediante la utilización del sistema web.
- Proveer de servicios de análisis estadísticos aplicables a bases de datos entomológicas.

1.3. Justificación

Los investigadores entomológicos, están realizando diversos estudios de investigación en el país sobre la diversidad, fluctuación poblacional e importancia económica de insectos. Este hecho ha generado que existan diversas fuentes de datos (hojas de cálculo electrónico) de carácter privado. Lo que causa dificultad si en un momento dado se quisiera compartir el estudio con diversos investigadores. Se ha considerado importante solucionar dicho problema mediante una aplicación que permita el almacenamiento y el intercambio de datos, para de esa manera realizar un registro adecuado para investigaciones futuras y entre otros aspectos, contribuir a incrementar las fuentes de datos para estudios de la UTEQ, las mismas que se podrían intercambiar con otros investigadores para ampliar la investigación científica.

Al disponer de una aplicación móvil que permita la recolección de datos en lugares de difícil acceso, los investigadores se verán beneficiados al no tener que preocuparse de los costos que podrían generar herramientas que provean esos datos.

Con este trabajo los investigadores de la UTEQ contaran con una herramienta que le facilitará el almacenamiento de datos, que puedan ser utilizados para experimentar con una o más variables, buscar relaciones o a su vez poder aplicar diversas técnicas de análisis estadístico que permitan descubrir conocimiento.

La investigación es conveniente porque la herramienta permitirá que se facilite la implementación de técnicas de análisis estadístico y de minería a base de datos entomológicas, lo cual abre la posibilidad de obtener resultados más diversos de los que se tiene actualmente.

CAPÍTULO II

FUNDAMENTACIÓN TEÓRICA DE LA INVESTIGACIÓN

2.1. Marco conceptual

2.1.1. Estadística

Se define como el arte de aprender a partir de los datos. Está relacionada con la recopilación de datos, su descripción subsiguiente y su análisis, lo que nos lleva a extraer conclusiones [3]. Se puede dividir en dos partes: la estadística descriptiva y la estadística analítica.

La **estadística descriptiva** se plantea como objetivo representar, sumarizar y resumir los resultados. Una vez terminado un trabajo de investigación se dispone de muchos datos, que no se utilizaran eficientemente si no se aplican técnicas que los resuman. Las técnicas descriptivas dependen del tipo de variables y de los objetivos del trabajo realizado [4].

La **estadística analítica**, también denominada inferencial, estudia los elementos de una muestra y a partir de ellos infiere propiedades a la población muestreada. Su objetivo es conocer el todo estudiando una parte, evidentemente con un margen de error [4].

La correlación estudia el grado de dependencia que existe entre dos variables. Para cuantificar este grado de dependencia, se definen los coeficientes de correlación. El coeficiente de correlación halla un valor que nos dé una medida del grado de ajuste de la curva a la nube de puntos. El coeficiente de **Pearson** es un índice que mide el grado de covariación entre distintas variables relacionadas linealmente. En cuanto al coeficiente de **Spearman** se usa cuando los datos proporcionados por los valores de las variables no vienen dados por sus frecuencias absolutas, sino por el orden que éstas ocupan en la observación, resulta más difícil medir la intensidad de la relación, puesto que se posee menos información sobre las variables [5].

2.1.2. Minería de datos

Se define como el proceso de descubrir patrones en los datos. En [6] se la delimita como "el proceso de utilizar una variedad de herramientas de análisis de datos para descubrir patrones y relaciones en los datos que se pueden utilizar para hacer predicciones válidas".

La minería de datos se lleva a cabo por etapas. Inicialmente la finalidad de los sistemas de información era recopilar información sobre una parcela determinada para ayudar en la toma de decisiones. Con la información de las organizaciones y la aparición de aplicaciones software operacionales sobre el sistema de información, la finalidad principal de los sistemas de información es dar soporte a los procesos básicos de la organización (ventas, producción, personal). Una vez satisfecha la necesidad de tener un soporte de información para los procesos básicos de la organización (sistemas de información para gestión), las organizaciones exigen nuevas prestaciones de los sistemas de información (sistemas de información para la toma de decisiones) [7]. De esta manera han aparecido diferentes herramientas de negocio para la toma de decisiones (DSS) que coexisten: EIS, OLAP, consultas e informes, y las propias herramientas de minería de datos.

2.1.3. Microsoft Analysis Services

Ha sido el componente de inteligencia empresarial del software Microsoft SQL Server desde el año 2000. Proporciona funciones OLAP, minería de datos para quienes toman decisiones. Sirve sus propios algoritmos árboles de decisión, agrupación, extracción de reglas de asociación, Naive Bayes, análisis de series de tiempo y algoritmos de redes neuronales [8].

2.1.4. Proceso de minería datos

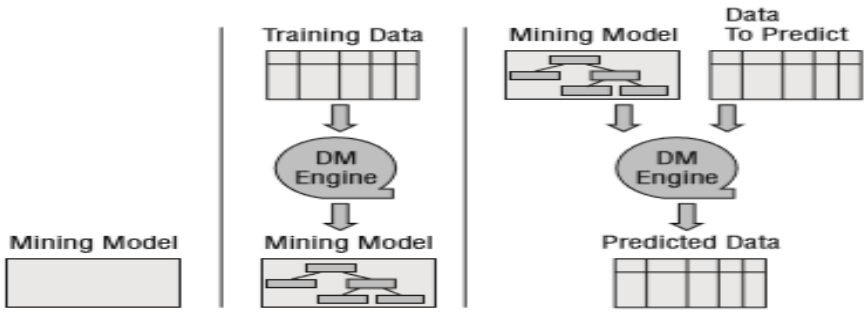
La minería de datos se lleva a cabo mediante la selección del algoritmo más adecuado para el trabajo quiera realizar, a partir de ello el algoritmo puede extraer patrones que luego se puedan usar para la inspección o para deducir información sobre nuevos datos.

Proceso resumido:

1. Definir el problema y formular un objetivo, denominado modelo de minería de datos.
2. Proporcionar los ejemplos de datos para el algoritmo. El mismo utiliza la definición de su problema (modelo de minería) como una especie de lente a través del cual examinar los datos y extraer los patrones (entrenar el modelo). Cuando el proceso termine, dependiendo de lo que esté tratando de realizar, podría detenerse y analizar los patrones encontrados. Finalmente puede proporcionar nuevos datos, formulados

de la misma manera que los ejemplos de capacitación, para realizar predicciones o deducciones de información sobre los ejemplos que utilizan los patrones descubiertos por el algoritmo [9]. En la (Ilustración 1) se puede observar los procesos llevados a cabo por SSAS para realizar la minería de datos.

Ilustración 1. Proceso de minería de datos SSAS



FUENTE: [9]

2.1.5. Conceptos clave para formular e implementar un escenario de minería

2.1.5.1. Atributo

Es la parte básica definida como información única de un ejemplo, podría ser el sexo, ingresos anuales o si un cliente compra un producto en particular. Los atributos podrían ser:

- Categóricos (o discretos) tienen un número determinado de valores (masculino o femenino)
- Continuos por definición números. Permiten operarse matemáticamente y usar tipo de distribución (como gaussiana).
- Adicionales que se basan en categóricos o continuos principales por ejemplo tipos ordenados o cíclicos son categóricos, pero con orden preestablecido (tamaño de una camisa)
- Discretizados son de tipo categórico, pero son continuos que se dividen en rangos discretos para el análisis. Esto sucede por los ciertos algoritmos funcionan solo con este tipo como reglas de asociación [9].

Un algoritmo de minería de datos tratará cada **estado** con un sesgo igual y no entenderá ninguna relación implícita entre ellos. Por ejemplo, donde se usen atributos divorciados y viudo, el algoritmo no entenderá que de hecho es soltero. En SQL Server Data Mining:

- Todos los atributos tienen el estado faltante implícito. Este estado indica que el atributo puede no estar presente en absoluto para una fila particular en los datos.
- Otro estado puede ser existente implícito, lo que significa que la presencia del atributo es la única información disponible, como la existencia de un producto en una cesta de la compra
- El estado de falta también se usa con frecuencia cuando se encuentra un valor no visto previamente durante la fase de predicción [9].

2.1.5.2. Caso DMX

Es un solo ejemplo que proporciona al algoritmo de extracción de datos. Consta de un conjunto de atributos con sus valores asociados. Aunque esto suena simple, malinterpretar su caso es la causa más común de falla en una implementación de extracción de datos. Un caso representa la entidad que está minando, sobre lo que está preguntando [9].

2.1.5.3. Clave

Hace referencia a dos tipos diferentes de columnas como claves. Estas columnas tienen significados e implicaciones muy diferentes. Los primeros resultados se refieren a la identidad de la entidad representada por el caso. En muchas implementaciones, la clave principal de la fuente de datos se puede usar como caso clave [9].

2.1.5.4. Usos de los atributos

En la minería de datos de SQL Server los atributos pueden ser entrada, salida, ambos. En términos generales, los algoritmos de minería de datos usan entradas para aprender sobre salidas. En el caso de que un atributo sea tanto de entrada como de salida, la convención adoptada por el Microsoft le proporciona valores de rendimiento que se utilizan para predecirse [9].

2.1.5.5. Origen de datos

Es una cadena de conexión que indica la ubicación de los datos. Que contiene información adicional que le indica a Analysis Services exactamente cuál es la propiedad de credencial de seguridad (o la personificación) cuando se conecta a la base de datos.

Se puede crear un objeto de origen de datos con una de las siguientes cuatro opciones de suplantación:

- Suplantar cuenta (Impersonate Account) le permite especificar las credenciales de la cuenta que se utilizarán para acceder a la fuente de datos. Las credenciales (que consisten en un nombre de usuario y una contraseña) se almacenan con Analysis Services, y todo acceso a esos datos usa esas credenciales. Este método es el más seguro si la delegación no es una opción [9].

2.1.5.6. Vista fuente de datos

Es una capa de abstracción que le permite modificar la forma en que se ven las fuentes de datos o incluso definir el esquema y cambiar la fuente real en un momento posterior. En la fuente de datos se selecciona, organiza, explora y se manipulan los datos de la fuente. La vista de fuente de datos le dice a Analysis Services cómo desea ver los datos en la fuente.

2.1.5.7. Estructura de minería

Describe la forma del problema y representa las columnas de datos disponibles para el problema de minería planteado, así como la información sobre esas columnas, y puede contener un cache de los datos de origen utilizados para capacitar y probar sus modelos.

La Estructura de minería contiene una lista de columnas, con los tipos de datos (ver Tabla 1) y la información que describen cómo deben manejarse, esencialmente, si son categóricos o continuos (ver Tabla 2). Las columnas en una estructura minera no representan atributos; son simplemente las columnas de datos que estarán disponibles al crear modelos [9].

Cuando se procesa una estructura de minería de datos, contiene un caché comprimido, o copia, de los datos de origen. Este caché se utiliza para entrenar cualquier modelo que se agregue posteriormente a la estructura.

Tabla 1. Tipos de datos y contenido admitido por algoritmos SSAS

Tipo de dato	Tipos de contenido admitido
Long	Continuous, Cyclical, Discrete, Discretized, Key, Key Sequence, Key Time, Ordered, Sequence, Time, Classified
Double	Continuous, Cyclical, Discrete, Discretized, Key, Key Sequence, Key Time, Ordered, Sequence, Time, Classified
Text	Cyclical, Discrete, Discretized, Key Sequence, Ordered, Sequence
Date	Continuous, Cyclical, Discrete, Discretized, Key, Key Sequence, Key Time, Ordered
Boolean	Cyclical, Discrete, Ordered

FUENTE: [9]

Tabla 2. Tipos de contenido y su detalle

Tipo contenido	Detalle
Key	Indica que la columna identifica una fila de forma inequívoca. Utilizado para indicar q no se utilice en análisis, solo de seguimiento de registro.
Discrete	Indica que la columna contiene un número finito de valores no continuos. Los valores no pueden implicar la ordenación, aun sean numéricos. Como género que específica de categorías. Compatible con todos los tipos de datos de minería [10].
Continuos	Indica que la columna contiene valores que representan datos numéricos en una escala que permite valores intermedios. Compatible con tipos de datos: date, double, long.
Discretized	Es un proceso mediante el cual los valores de un conjunto de datos continuo se incluyen en depósitos para que haya un número limitado de valores posibles. Solo se puede discretizar los datos

	numéricos. Compatible con tipos de datos: date, double, long y text [10].
Key time	Solamente se puede utilizar en modelos de serie temporal. Indica que los valores están ordenados y que representan una escala de tiempo. Compatible con tipos de datos: double, long y date.
Key sequence	Solamente se puede utilizar en modelos de agrupación en clústeres de secuencia. Indica que la columna contiene valores que representan una secuencia de eventos. Compatible con tipos de datos: double, long, text y date [10].
Cyclical	Indica que la columna contiene valores que representan un conjunto ordenado cíclico. Como los días numerados de la semana. Columnas se consideran ordenados y discretos en términos de contenido. Compatible con todos los tipos de datos de minería de datos en Analysis Services.
Ordered	Indica también que la columna contiene valores que definen una secuencia u orden. Pero esto no implican ninguna relación de distancia o magnitud entre los valores del conjunto. No se considera discreta en tipo de contenido. Compatible con todos los tipos de datos de minería de datos en Analysis Services.
Table	Indica que la columna contiene otra tabla de datos, con una o más columnas y una o más filas. Tipo de dato siempre es table.

FUENTE: [9]

2.1.5.8. Columnas discretizadas

Se define como la operación de romper valores en rangos. La razón de realizarlo se debe a:

- Que los rangos a menudo tienen más sentido que los valores y variaciones cuando observamos los patrones un modelo.
- Que algunos algoritmos simplemente no aceptan valores continuos [9].

Estos procesos, se realizan manualmente por los científicos de datos, ellos definen los rangos para usarlos en sus modelos. La minería de SQL Server resuelve este problema mediante la creación automática de grupos basados en la distribución de los datos en la columna y luego

utiliza el número de grupo cuando expone los datos al algoritmo. Esto tiene la ventaja obvia de no requerir que el usuario predetermine los rangos, pero también ayuda más adelante durante la predicción, lo que le permite proporcionar el algoritmo con valores en su forma nativa y continua, y permitir que SQL Server Data Mining determine qué grupo contiene el valor [9].

Discretizado es el único tipo de contenido que está parametrizado. El comportamiento predeterminado del sistema es crear cinco grupos al intentar primero un enfoque de áreas iguales y, si eso falla, usar un enfoque de agrupamiento. Si no hay datos suficientes para admitir cinco grupos, el sistema intentará automáticamente menos grupos [9].

2.1.5.9. Modelo minería

Mientras que una estructura de minería de datos es un contenedor y un descriptor, un modelo de minería de datos es el objeto que transforma filas de datos en casos y realiza el aprendizaje automático utilizando un algoritmo de minería de datos específico.

Se describe como un subconjunto de columnas de la estructura, cómo deben usarse esas columnas como atributos (de entrada, salida o ambo), junto con los parámetros de algoritmo y algoritmo que se usarán para realizar el aprendizaje de máquina en los datos de la estructura [11].

En una columna de minería de datos puede definir dos elementos adicionales de información uso y marcas de modelado:

- **Uso** define cómo el modelo va a usar la columna. Se pueden usar como entrada, clave o ambos.
- **Marca de modelado** proporciona información adicional al algoritmo sobre los datos que se definen en la tabla de casos, ayuda a generar un modelo más preciso. Marcas de modelado que se puede establecer:
    - MODEL_EXISTENCE_ONLY: Indica que la presencia del atributo es más importante que los valores que están en la columna de atributos.
    - REGRESSOR: Indica que el algoritmo puede usar la columna especificada en la fórmula de regresión de algoritmos de regresión. Esta marca se admite

en los algoritmos de árboles de decisión de Microsoft y de serie temporal de Microsoft [11].

En detalle el modelo usa las entradas para aprender sobre las salidas. La idea detrás de la minería de datos es mostrar un modelo de datos (ejemplos de datos) que contengan tanto entradas como salidas, a partir de las cuales puede extraer patrones. Esto se llama la **fase de entrenamiento**. Los patrones pueden estudiarse por sí mismos o aplicarse a nuevos ejemplos de datos. Para probar el rendimiento predictivo, al modelo entrenado solo se le asignan los atributos de entrada de los nuevos casos y, a partir de estos, intenta predecir el estado del atributo de salida. La predicción se compara con el estado conocido del atributo de salida de ese caso. De esta manera, se puede evaluar el rendimiento del modelo para predecir los resultados. Esto se llama la **fase de prueba** [9].

2.1.5.10. Particionamiento en entrenamiento y conjunto de pruebas

La estructura de minería le permite dividir automáticamente sus datos de origen en conjuntos de pruebas y capacitación. Puede especificar un porcentaje de los datos que se utilizarán para la prueba (ver Tabla 3), un recuento de casos o ambos (en cuyo caso, se utiliza la menor de las dos cantidades). Los datos que no se guardan para la prueba se utilizan para la capacitación. El conjunto de pruebas se utiliza para validar modelos después del entrenamiento.

Tabla 3. Propiedades para crear conjunto de datos de prueba

Propiedades	Descripción
HoldoutMaxCases	Especifica el número máximo de casos que se van a incluir en el conjunto de pruebas.
HoldoutMaxPercent	Especifica el número de casos que se van a incluir en el conjunto de pruebas como porcentaje del conjunto de datos completo. Para no tener ningún conjunto de datos, especificaría 0.
HoldoutSeed	Especifica un valor entero para usarlo como valor de inicialización al seleccionar los datos para las particiones de forma aleatoria. Este valor no afecta al número de casos del

	conjunto de entrenamiento; sino que sirve para asegurarse de que la partición se puede repetir.

FUENTE: [9]

Analysis Service permite la selección de conjunto aleatorios (HOLDOUT) de casos para la prueba. Puede garantizar que se seleccionen las mismas filas cada vez agregando la cláusula REPEATABLE (<semilla>) inmediatamente después de la especificación HOLDOUT [9].

2.1.6. Algoritmos Analysis Services

Tipos de algoritmos disponibles por SQL Server:

- Clasificación, que predicen una o más variables discretas, basándose en los demás atributos del conjunto de datos.
- Segmentación, que dividen los datos en grupos, o clústeres, de elementos que tienen propiedades similares.
- Asociación, que buscan correlaciones entre diferentes atributos de un conjunto de datos.
- Análisis de secuencias resumen las secuencias frecuentes o episodios en los datos.

Tabla 4. Detalle de los algoritmos permitidos por SSAS

Algoritmo	Detalle	Requisito de uso	Tipos de contenido soportado
Regresión Lineal	Es una variación del algoritmo de árboles de decisión de Microsoft que ayuda a calcular una relación lineal entre una variable independiente y otra dependiente y, a continuación, utilizar esa relación para la predicción.	**Columna de una sola clave** (identifica cada registro de manera única). **Al menos una columna de predicción** o más, tipo numérico continuo.	Continuo Ordenado Cíclico

	La relación toma la forma de una ecuación para la línea que mejor represente una serie de datos. No permite predecir datos fecha. Ni datos de entrada fecha.	**Columnas de entrada** numérica continua con tipo de dato adecuado.	
Árbol de decisión	La idea principal de un árbol de decisión es dividir sus datos recursivamente en subconjuntos. Cada atributo de entrada se evalúa para determinar con qué claridad divide los datos entre las clases (o estados) de su variable objetivo (atributo predecible). El proceso de evaluación de todas las salidas se repite en cada subgrupo. Cuando este proceso recursivo se completa, se forma un árbol de decisión [9].	**Columna de una sola clave** (identifica cada registro de manera única) **Al menos una columna de predicción** o más (tipos numérico o discreto). **Columnas de entradas** discretas o continuos	Continuo Cíclico Discreto Discretizado Ordenado
Naive Bayes	El algoritmo Bayes naive de Microsoft es un algoritmo de clasificación basado en los teoremas de Bayes y se puede usar para el modelado de predicción y de exploración. El algoritmo Bayes naive de Microsoft calcula la probabilidad de cada estado de cada columna de entrada, dado cada posible estado de la columna de predicción. Las variables deben ser independientes.	**Una columna de una sola clave** (identifica cada registro de manera única) **Columna de entrada** tipos discreta todas. **Al menos una columna de predicción** el cual debe contener valores discretos o discretizados.	Discretizado Cíclico Discreto Ordenado

	Atributos de entrada sean independientes unos de otros. Dado que un modelo Bayes naive no admite tipos de datos continuos, todos los valores de las columnas de entrada se tratan como discretos o discretizados [12].		
Agrupamiento	Es un algoritmo de segmentación o clústeres que itera en los casos de un conjunto de datos para agruparlos en clústeres que contengan características similares. Estas agrupaciones son útiles para la exploración de datos, la identificación de anomalías en los datos y la creación de predicciones. Los modelos de agrupación en clústeres identifican las relaciones en un conjunto de datos que no se podrían derivar lógicamente a través de la observación casual [13].	**Una columna de una sola clave** (identifica cada registro de manera única) **Al menos una columna de entrada** q contenga valores q se utiliza para generar los clusteres. **No necesita columna de predicción**. Pero se puede agregar	Continuo Cíclico Discreto Discretizado Ordenado

FUENTE: [9], [12], [13]

ELABORADO POR: AUTOR

2.1.6.1. Regresión Lineal

El algoritmo de regresión lineal de Microsoft es una versión especial del algoritmo de árboles de decisión de Microsoft que está optimizado para modelar pares de atributos continuos. El método de puntuación y selección de características es la puntación de grado de interés, porque el modelo únicamente admite las columnas continuas. La ecuación que representa la recta de regresión toma la forma general de **y = ax + b** y se conoce como ecuación de

regresión. La variable Y representa la variable de salida, X representa la variable de entrada, y a y b son coeficientes ajustables [14].

2.1.6.2. Árbol de decisión

Se caracteriza por ser un algoritmo de clasificación y regresión para el modelo de predicción de atributos discretos y continuos. Cuando los atributos son de tipo discreto el algoritmo hace predicciones basándose en las relaciones entre las columnas de entrada de un conjunto de datos. Utiliza los valores, conocidos como estados, de estas columnas para predecir los estados de una columna que se designa como elemento de predicción. Específicamente, el algoritmo identifica las columnas de entrada que se correlacionan con la columna de predicción. Cuando la entrada es contínua, el algoritmo usa la regresión lineal para determinar donde se divide un árbol de decisión [15].

El algoritmo es híbrido que incorpora distintos métodos para crear un árbol y admite varias tareas de análisis, incluye regresión, la clasificación y la asociación. El algoritmo en si aplica el enfoque bayesiano en el aprendizaje de los modelos de interacción causales al obtener las distribuciones posteriores aproximadas de los modelos. La metodología para evaluar el valor de la información de las prioridades necesarias para el aprendizaje se basa en el supuesto de equivalencia de probabilidad. Este supuesto establece que los datos no deberían ayudar a discriminar estructuras de red que, de otro modo, representarían las mismas aserciones de independencia condicional. Se supone que cada caso tiene una única red bayesiana anterior y una única medida de confianza para dicha red [15].

Mediante estas redes anteriores, el algoritmo calcula las probabilidades posteriores relativas de las estructuras de red dados los datos de entrenamiento actuales, e identifica las estructuras de red con las probabilidades posteriores más altas. La forma y profundidad del árbol integrado en un modelo determinado depende del método de puntuación y del resto de parámetros usados. Los cambios en los parámetros también pueden afectar al lugar donde se dividen los nodos [15].

Cuando el algoritmo de árboles de decisión de Microsoft crea el conjunto de valores de entrada posibles, realiza la selección de características para identificar los atributos y valores que proporcionan la mayor información y elimina los valores que son muy raros. El

algoritmo también agrupa los valores en contenedores, para crear agrupaciones de valores que pueden procesarse como una unidad para optimizar el rendimiento [15].

Un árbol se construye determinando las correlaciones entre una entrada y el resultado objetivo. Una vez que todos los atributos se han correlacionado, el algoritmo identifica el atributo único que separa de manera más clara los resultados. Este punto de la mejor separación se mide utilizando una ecuación que calcula la ganancia de información. El atributo que tiene la mejor puntuación para la ganancia de información se usa para dividir los casos en subconjuntos, que luego se analizan recursivamente por el mismo proceso, hasta que el árbol ya no se pueda dividir. La ecuación exacta utilizada para evaluar la ganancia de información depende de los parámetros establecidos cuando creó el algoritmo, el tipo de datos de la columna predecible y el tipo de datos de la entrada [15].

2.1.6.3. Naive Bayes

Se caracteriza por ser un algoritmo de clasificación, basado en teoremas de Bayes. Y generalmente es utilizado para predicción y exploración. El algoritmo está basado en los teoremas de Bayes, que expresa la probabilidad condicional de un evento aleatorio A dado B en términos de la distribución de probabilidad condicional del evento B dado A y la distribución de probabilidad marginal de sólo A. Por ejemplo, que sabiendo la probabilidad de tener un dolor de cabeza dado que se tiene gripe, se podría saber (si se tiene algún dato más), la probabilidad de tener gripe si se tiene un dolor de cabeza.

Fórmula de Bayes con base a la definición de Probabilidad condicionada se obtiene la fórmula de Bayes, también conocida como la regla de Bayes [16].

Ecuación 1. Teorema de Bayes.

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{\sum_{k-1}^{n} P(B|A_k)P(A_k)}$$

Esta fórmula nos permite calcular la probabilidad condicional $P(A_i|B)$ de cualquiera de los eventos B, dado.

2.1.6.4. Agrupamiento

Se caracteriza por ser un algoritmo de segmentación o clústeres. Se utiliza para explorar datos, identificar anomalías en los datos y la creación de predicciones. Es capaz de identificar las relaciones en conjuntos de datos que no se podrían derivar lógicamente a través de la observación casual. El algoritmo funciona mediante la identificación de las relaciones de un conjunto de datos y genera una serie de clústeres basándose en ellas. Después de definir los clústeres, el algoritmo calcula el grado de perfección con que los clústeres representan las agrupaciones de puntos y, a continuación, intenta volver a definir las agrupaciones para crear clústeres que representen mejor los datos. El algoritmo establece una iteración en este proceso hasta que ya no es posible mejorar los resultados mediante la redefinición de los clústeres. El método que se utiliza para crear los clústeres y asignar puntos de datos a dichos clústeres es Expectation Maximization (EM), es un método de agrupación en clústeres blando. Esto significa que un punto de datos siempre pertenece a varios clústeres, y que se calcula una probabilidad para cada combinación de punto de datos y clúster [13].

El funcionamiento del algoritmo es el siguiente, refina de forma iterativa un modelo de clústeres inicial para ajustar los datos y determina la probabilidad de que un punto de datos exista en un clúster. El algoritmo finaliza el proceso cuando el modelo probabilístico ajusta los datos. La función usada para determinar el ajuste es el logaritmo de la probabilidad de los datos dado el modelo. Si durante el proceso se generan clústeres vacíos, o si la pertenencia de uno o varios de los clústeres cae por debajo del umbral especificado, los clústeres con poblaciones bajas se reinicializan en los nuevos puntos y vuelve a ejecutarse el algoritmo EM [13].

El algoritmo EM se usa de forma predeterminada en los modelos de agrupación de Microsoft por que proporciona ventajas en relación con el algoritmo mediana-k:

- Requiere examinar la base de datos como máximo una vez.
- Funciona incluso si la cantidad de memoria (RAM) es limitada.
- Tiene la capacidad de usar un cursor de solo avance.
- Sus resultados superan los obtenidos por los métodos de muestreo.

2.1.7. Visor de árbol de contenido genérico de Microsoft

Puede explorar el modelo para buscar tendencias interesantes. Este visor muestra el contenido del modelo de minería de datos según un formato de tabla HTML normalizado (ver Ilustración 2). Sin embargo, la organización de los nodos y el contenido de cada nodo variarán considerablemente dependiendo del algoritmo utilizado para generar los resultados. Es más útil cuando ya se entiende el modelo y se desea extraer estadísticas o reglas a partir de un nodo concreto. Por ejemplo, utilice el visor genérico cuando desee ver información detallada sobre los patrones y estadísticas que Analysis Services capturó durante el análisis, como la probabilidad de un nodo, o una fórmula de regresión [17].

Ilustración 2. Visor de árbol de contenido genérico SSAS.

Leyenda de nodo (id. único)

- (0)
  - All (000000001)

Detalles de nodo

MODEL_CATALOG	ProyectoMultidimensional1
MODEL_SCHEMA	
MODEL_NAME	mmTbregresionlineal
ATTRIBUTE_NAME	Atributo Y
NODE_NAME	000000001
NODE_UNIQUE_NAME	000000001
NODE_TYPE	25 (Raíz del árbol de regresión)
NODE_GUID	
NODE_CAPTION	All
CHILDREN_CARDINALITY	0
PARENT_UNIQUE_NAME	0
NODE_DESCRIPTION	All
NODE_RULE	<compound-predicate op="and" />
MARGINAL_RULE	
NODE_PROBABILITY	1
MARGINAL_PROBABILITY	1
NODE_DISTRIBUTION	
NODE_SUPPORT	7000
MSOLAP_MODEL_COLUMN	Atributo Y
MSOLAP_NODE_SCORE	0
MSOLAP_NODE_SHORT_CAPTION	All

ATTRIBUTE_NAME	ATTRIBUTE_VALUE	SUPPORT	PROBABILITY	VARIANCE	VALUETYPE
Atributo Y	Missing	0	0	0	1 (Ausente)
Atributo Y	828.761627142859	7000	1	133470.771372679	3 (Continuo)
Atributo X	0.499828515981056	0	0	520981.495634204	7 (Coeficiente)
Atributo X	12934.4780838845	0	0	0	8 (Ganancia de puntuación)
Atributo X	1257.84028571429	0	0	520981.495634204	9 (Estadísticas)
	200.0571837931	0	0	3314.2534149562	11 (Interceptar)

ELABORADO POR: AUTOR

El visor de contenido genérico de Microsoft muestra las columnas, las reglas, las propiedades, los atributos, los nodos, relaciones de elementos primarios y secundarios entre los nodos y otro tipo de contenido del conjunto de filas de esquema de contenido del modelo de minería de datos. El conjunto de filas de esquema de contenido es un marco genérico para presentar información detallada (ver Tabla 5)sobre el contenido de un modelo de minería de datos [17].

Tabla 5. Identificador de nodo, nombre, título y descripción.

MODEL_CATALOG	Metadato de donde esta almacenado el modelo.
MODEL_SCHEMA	Nombre del esquema.
MODEL_NAME	Nombre del modelo.
ATTRIBUTE_NAME	Nombre del atributo puede ser el nombre de una columna.
NODE_NAME	Nombre usado como identificador único, cada nodo tiene un nombre.
NODE_UNIQUE_NAME	El nodo raíz de cualquier modelo siempre tiene el identificador único 0.
NODE_TYPE	Observar la tabla 6 para una mejor compresión.
NODE_GUID	
NODE_CAPTION	Sirven como etiquetas para ayudar a comprender el contenido del nodo.
CHILDREN_CARDINALITY	La cardinalidad hace referencia al número de elementos en un conjunto. En el contexto de un modelo de minería de datos procesado, la cardinalidad indica el número de elementos secundarios de un nodo determinado. únicamente los nodos secundarios inmediatos se cuentan al calcular la cardinalidad de un nodo.
PARENT_UNIQUE_NAME	La relación entre los nodos primarios y los nodos secundarios en una estructura de árbol. • Si es NULL significa que el nodo es el nodo superior del modelo. • Si es 0, el nodo debe ser un descendiente directo del nodo superior en el modelo. Esto se debe a que el identificador del nodo raíz es 0 siempre.

NODE_DESCRIPTION	Sirven como etiquetas para ayudar a comprender el contenido del nodo.
NODE_RULE	Contiene fragmentos XML que se pueden utilizar para serializar un modelo o para representar alguna parte de la estructura del modelo. Describe la ruta de acceso al nodo actual desde la raíz del modelo.
MARGINAL_RULE	Define el atributo y el valor para el nodo actual.
NODE_PROBABILITY	Estas columnas solo contienen valores en los nodos en los que un valor de probabilidad es significativo. Es la probabilidad de alcanzar el nodo desde la raíz. La probabilidad de nodo siempre es mayor o igual que la probabilidad marginal.
MARGINAL_PROBABILITY	Es la probabilidad de alcanzar el nodo a partir de su elemento primario.
NODE_DISTRIBUTION	Contiene una tabla anidada que en muchos nodos proporciona información importante y detallada sobre los modelos que detecta el algoritmo. Las estadísticas exactas que se proporcionan en esta tabla cambian según el tipo de modelo, la posición del nodo en el árbol y si el atributo de predicción es un valor numérico continuo o discreto; sin embargo, pueden incluir los valores mínimo y máximo de un atributo, los pesos asignados a los valores, el número de casos de un nodo, los coeficientes que se usan en una fórmula de regresión y medidas estadísticas como la desviación estándar y la varianza. **ATTRIBUTE_NAME**: Puede ser el nombre de una columna, como un atributo de

predicción, una regla, un conjunto de elementos o un fragmento de información interno al algoritmo, como parte de una fórmula. Tambien puede contener atributo valor.

ATTRIBUTE_VALUE: Si el nombre de atributo es una columna, en el caso más sencillo, contiene uno de los valores discretos para esa columna.

En función de cómo el algoritmo procese los valores, también puede contener una marca que indique si existe un valor para el atributo (Existing) o si el valor es nulo (Missing).

SUPPORT: Recuento de los casos que tienen este par atributo-valor o que contienen este conjunto de elementos o regla.

VARIANCE: Indica la varianza de los valores dentro del nodo. Por definición, la varianza siempre es 0 para los valores discretos. Si el modelo admite valores continuos, la varianza se calcula como σ (sigma), usando el denominador n o el número de casos en el nodo.

PROBABILITY: Indica la probabilidad para este nodo concreto dentro del modelo completo. Generalmente, la probabilidad representa la compatibilidad para este valor determinado, dividido por el recuento total de casos dentro del nodo (NODE_SUPPORT).

VALUETYPE: Indica el tipo de datos del valor o un atributo, y el uso del valor. Ver tabla 7.

NODE_SUPPORT	Recuento de los casos que tienen este par atributo-valor o que contienen este conjunto de elementos o regla.
MSOLAP_MODEL_COLUMN	Nombre del atributo de predicción.
MSOLAP_NODE_SCORE	Igual que NODE_PROBABILITY.
MSOLAP_NODE_SHORT_CAPTION	Etiqueta que se utiliza para la visualización.

FUENTE: [17]

Tabla 6. Tipos de valores para Value Type

VALUE_ TYPE ID	ETIQUETA DEL VALOR	NOMBRE DEL TIPO DE VALOR
1	Missing	Indica que los datos del caso no contenían un valor para este atributo. El estado Missing se calcula de forma independiente de los atributos que tienen valores.
2	Existing	Indica que los datos del caso contienen un valor para este atributo.
3	Continuous	Indica que el valor del atributo es un valor numérico continuo y, por consiguiente, puede ser representado por una media, junto con la varianza y la desviación estándar.
4	Discrete	Indica un valor, numérico o de texto, que se trata como discreto.
5	Discretized	Indica que el atributo contiene valores numéricos que se han convertido en datos discretos. El valor será una cadena con formato que describe los depósitos de discretización.
6	Existing	Indica que el atributo tiene valores numéricos continuos y que los valores se han proporcionado en los datos, frente a los valores que están ausentes o se han deducido.
7	Coefficient	Indica un valor numérico que representa un coeficiente.
8	Score gain	Indica un valor numérico que representa la ganancia de puntuación para un atributo.
9	Etatistics	Indica un valor numérico que representa una estadística para un regresor.

10	Node unique name	Indica que el valor no se debería tratar como numérico o cadena, sino como el identificador único de otro nodo de contenido en un modelo.
11	Intercept	Indica un valor numérico que representa la intersección en una fórmula de regresión.
12	Periodicity	Indica que el valor denota una estructura periódica en un modelo.
13	Autoregressive order	Indica que el valor representa el número de serie de regresión automática.
14	Moving average order	Representa el número de medias móviles en una serie.
15	Difference order	Indica que el valor muestra cuántas veces se diferencia la serie.
16	Boolean	Representa un tipo booleano.
17	Other	Representa un valor personalizado definido por el algoritmo.
18	Prerendered string	Representa un valor personalizado que el algoritmo representa como una cadena. El modelo de objetos no aplicó ningún formato.

FUENTE: [17]

Tabla 7. Lista de tipos de nodos de contenido minería de datos.

NODE_TYPE ID	ETIQUETA DE NODO	CONTENIDO DE NODO
1	Modelo	Los metadatos y el nodo de contenido raíz. Se aplica a todos los tipos de modelos.
2	Tree	Nodo raíz de un árbol de clasificación. Se aplica a los modelos del árbol de decisión.
3	Interior	Nodo de división interior en un árbol. Se aplica a los modelos del árbol de decisión.
4	Distribution	Nodo terminal de un árbol. Se aplica a los modelos del árbol de decisión.

5	Cluster	Clúster detectado por el algoritmo. Se aplica a los modelos de agrupación en clústeres y a los modelos de clústeres de secuencia.
6	Unknown	Tipo de nodo desconocido.
7	ItemSet	Conjunto de elementos detectado por el algoritmo. Se aplica a modelos de asociación o modelos de agrupación en clústeres de secuencia.
8	AssociationRule	Regla de asociación detectada por el algoritmo. Se aplica a modelos de asociación o modelos de agrupación en clústeres de secuencia.
9	PredictableAttribute	Atributo de predicción. Se aplica a todos los tipos de modelos.
10	InputAttribute	Atributo de entrada. Se aplica a los árboles de decisión y los modelos Bayes Naïve.
11	InputAttributeState	Estadísticas sobre los estados de un atributo de entrada. Se aplica a los árboles de decisión y los modelos Bayes Naïve.
13	Sequence	Nodo superior para un componente del modelo Markov de un clúster de secuencia. Se aplica a los modelos de agrupación en clústeres.
14	Transition	Matriz de transición de Markov. Se aplica a los modelos de agrupación en clústeres.
15	TimeSeries	Nodo no raíz de un árbol de serie temporal. Solo se aplica a los modelos de serie temporal.
16	TsTree	Nodo raíz de un árbol de serie temporal que corresponde a una serie temporal de predicción. Se aplica a los modelos de serie temporal y solo si el modelo se creó utilizando el parámetro MIXED.
17	NNetSubnetwork	Una subred. Se aplica a los modelos de red neuronal.
18	NNetInputLayer	Grupo que contiene los nodos del nivel de entrada. Se aplica a los modelos de red neuronal.

19	NNetHiddenLayer	Grupos que contiene los nodos que describen el nivel oculto. Se aplica a los modelos de red neuronal.
21	NNetOutputLayer	Grupos que contiene los nodos del nivel de salida. Se aplica a los modelos de red neuronal.
21	NNetInputNode	Nodo en el nivel de entrada que coincide un atributo de entrada con los estados correspondientes. Se aplica a los modelos de red neuronal.
22	NNetHiddenNode	Nodo en el nivel oculto. Se aplica a los modelos de red neuronal.
23	NNetOutputNode	Nodo en el nivel de salida. Este nodo normalmente coincidirá con un atributo de salida y los estados correspondientes. Se aplica a los modelos de red neuronal.
24	NNetMarginalNode	Estadísticas marginales sobre el conjunto de entrenamiento. Se aplica a los modelos de red neuronal.
25	RegressionTreeRoot	Raíz de un árbol de regresión. Se aplica a los modelos de regresión lineal y a los modelos de árboles de decisión que contiene atributos de entrada continuos.
26	NaiveBayesMarginalStatNode	Estadísticas marginales sobre el conjunto de entrenamiento. Se aplica a los modelos Bayes Naïve.

FUENTE: [17]

2.1.8. Metodología de desarrollo de software Proceso Unificado Racional (RUP)

En [18] definen tres características distintivas. Estas son:

- Dirigido por casos de uso:
  - El proceso utiliza casos de usos para manejar el desarrollo desde la incepción hasta el despliegue [18].
- Centrado en arquitectura:

- El proceso busca entender los aspectos estáticos y dinámicos más significativos en términos de arquitectura de software. La arquitectura se define en función de las necesidades de los usuarios y se determina a partir de los Casos de Uso base del negocio [18].

- Iterativo e incremental:
  - El proceso reconoce que es práctico dividir grandes proyectos en proyectos más pequeños o mini-proyecto. Las interacciones son planificadas con base en los Casos de uso [18].

El proceso unificado racional divide su ciclo de vida en cuatro fases inicio, elaboración, construcción, transición. Además de las fases involucra nueve disciplinas, las cuales son repetibles en cada fase (modelado del negocio, requisitos, análisis y diseño, implementación, pruebas, despliegue, gestión del cambio y configuraciones, gestión del proyecto, entorno). A las seis primeras disciplinas se les conoce como flujos de trabajo del proceso y a las tres últimas como flujos de trabajo de soporte. La reciprocidad que tiene RUP con UML es muy estrecha, ya que el primero establece las actividades y los criterios para construir un sistema desde el máximo nivel de abstracción (ideas), hasta el nivel más concreto (software); mientras que el segundo ofrece una perspectiva gráfica para representar y documentar los modelos en cada iteración [19]. A continuación, se describen las cuatro fases mencionadas:

2.1.8.1. Fase de inicio

Se establece el caso de negocios para el sistema y delimita el alcance del proyecto. Para lograr esto, debe identificar todas las entidades externas con las que el sistema interactuará (actores) y definir la naturaleza de esta interacción a un alto nivel [20].

Los resultados de la fase inicial son:

- Un modelo de caso de uso inicial (10% -20%) completo.
- Un glosario inicial del proyecto (opcionalmente puede expresarse parcialmente como un modelo de dominio es un modelo conceptual de todos los temas relacionados con un problema específico. En él se describen las distintas entidades, sus atributos, papeles y relaciones, además de las restricciones que rigen el dominio del problema.)

- Un modelo de negocio, si es necesario.
- Uno o varios prototipos.

Al final de la fase inicial se encuentra el primer hito importante del proyecto: el hito Objetivos del ciclo de vida. Los criterios de evaluación para la fase de inicio son:

- Concordancia de los interesados en la definición del alcance y las estimaciones de costo / programa.
- Comprensión de los requisitos como lo demuestra la fidelidad de los casos de uso primarios.
- Profundidad y amplitud de cualquier prototipo arquitectónico que se haya desarrollado [20].

2.1.8.2. Fase de elaboración

El propósito de la fase de elaboración es analizar el dominio del problema, establecer una base arquitectónica sólida y ejecutable, desarrollar el plan del proyecto y eliminar los elementos de mayor riesgo del proyecto. El esfuerzo en esta fase debe al menos abordar los casos de uso crítico identificados en la fase inicial, que generalmente exponen los principales riesgos técnicos del proyecto. Los resultados de la fase de elaboración son:

- Un modelo de caso de uso (al menos 80% completo)
- Requisitos complementarios que capturan los requisitos no funcionales y cualquier requisito que no esté asociado con un caso de uso específico.
- Descripción de una arquitectura de software.
- Un prototipo arquitectónico ejecutable.
- Un manual de usuario preliminar (opcional).

Al final de la fase de elaboración se encuentra el segundo hito importante del proyecto, el hito de la arquitectura del ciclo de vida. En este punto, examina los objetivos y el alcance detallados del sistema, la elección de la arquitectura y la resolución de los principales riesgos [20].

Los principales criterios de evaluación para la fase de elaboración incluyen las respuestas a estas preguntas:

- ¿Es estable la visión del producto?
- ¿Es la arquitectura estable?
- ¿La demostración ejecutable muestra que los principales elementos de riesgo se han abordado y resuelto de manera creíble?
- ¿Es el plan para la fase de construcción suficientemente detallado y preciso? ¿Está respaldado con una base creíble de estimaciones?
- ¿Todas las partes interesadas están de acuerdo en que la visión actual se puede lograr si el plan actual se ejecuta para desarrollar el sistema completo, en el contexto de la arquitectura actual?
- ¿Es aceptable el gasto real de recursos versus el gasto planificado?

2.1.8.3. Fase de construcción

En la fase de construcción todos los componentes y características de la aplicación son desarrollados e integrados en el producto. La fase de construcción es básicamente el proceso de fabricación de software, el cual se enfoca en manejar recursos y controlar operaciones para optimizar costos, tiempos y calidad. El resultado de la fase de construcción es un producto listo para poner en manos de sus usuarios finales. Como mínimo, consiste en:

- El producto software integrado en las plataformas adecuadas.
- Los manuales de usuario.
- Una descripción de la versión actual.

Al final de la fase de construcción se encuentra el tercer hito principal del proyecto (hito inicial de capacidad operativa). En este punto, usted decide si el software, los sitios y los usuarios están listos para operar, sin exponer al proyecto a altos riesgos. Esta versión a menudo se llama una versión beta [20].

Los criterios de evaluación para la fase de construcción implican responder a estas preguntas:

- ¿Es esta versión del producto estable y lo suficientemente madura como para ser implementada en la comunidad de usuarios?
- ¿Están todos los interesados listos para la transición a la comunidad de usuarios?
- ¿Los gastos de recursos reales en comparación con los gastos planificados siguen siendo aceptables?

2.1.8.4. Fase de transición

El propósito de la fase de transición es la transición del producto de software a la comunidad de usuarios. Una vez que el producto final se ha entregado al usuario final, generalmente surgen problemas que requieren que desarrolle nuevas versiones, corrija algunos problemas o finalice las funciones que se pospusieron [20]. Los resultados de la fase transición son:

- Pruebas beta para validar el nuevo sistema contra las expectativas del usuario.
- Formación de usuarios y mantenedores.
- Implementar el producto a los equipos de marketing, distribución y ventas.

Los objetivos principales de la fase de transición incluyen:

- Lograr la autocompatibilidad del usuario.
- Lograr la concurrencia de las partes interesadas de que las líneas de base de implementación son completas y consistentes con los criterios de evaluación de la visión.
- Lograr la línea de base del producto final de la manera más rápida y rentable posible.

2.1.9. Aplicación Web

Definida como software que es accedido vía web por una red como internet o una intranet. Generalmente se ejecutan en el entorno de software de navegación web, por ejemplo, un applet de Java. Se codifica en lenguajes soportados por el navegador como JavaScript, combinado con HTML; confiándose en el navegador web para que reproduzca (renderice) la aplicación [21]. Las aplicaciones web pueden tener una arquitectura con múltiples capas y componentes, habitualmente son de 3 capas.

En su forma más común, el navegador Web ofrece la primera capa y un motor capaz de usar alguna tecnología Web dinámica ejemplo: PHP, Java Servlets o ASP, ASP.Net, CGI, Python (Lenguaje de programación) o Ruby en Rails, constituyen la capa intermedia. Por último, una base de datos constituye la tercera capa [22].

2.1.10. Arquitectura Web Cliente/Servidor

En términos básico la arquitectura se define como un cliente solicitando servicios al servidor el cual responde con una respuesta http con el contenido solicitado [23]. La arquitectura permite la separación en tres niveles:

- Lógica de presentación: Encargada de la entrada y salida de la aplicación con el usuario. Entre sus tareas tiene obtener información del usuario, enviar información al nivel de lógica de negocio para su procesado, recibir los resultados del procesamiento de lógica de negocio [23].
- Lógica de negocio: Encargada de gestionar datos a nivel de procesamiento. Es el puente entre usuario y datos. Interactúa con la lógica de datos para ejecutar las reglas del negocio [23].
- Lógica de datos: Encargada de gestionar los datos a nivel de almacenamiento.

2.1.11. Arquitectura REST

Es un estilo de arquitectura para el diseño de aplicaciones orientadas a la web. Entre los estándares y protocolos que se recomiendan con el uso de REST se encuentran algunos como HTTP o XML, utilizado en las peticiones y respuestas REST cuando la seguridad es muy importante gracias a la facilidad para codificar y enviar mensajes (ficheros .XML) con este lenguaje. Existen muchos desarrolladores que prefieren usar REST junto a JSON para tener una alternativa más ligera y eficiente con los recursos de la web debido al uso que hace de HTTP [24].

En [25] definen a REST como un conjunto de principios arquitectónicos mediante los cuales se puede diseñar servicios web que se centren en los recursos de un sistema, incluyendo

cómo los estados de los recursos son tratados y transferidos a través de HTTP por una amplia gama de clientes escritos en diferentes idiomas. Ventajas de REST sobre SOAP son:

- REST comparte principios con HTTP, por lo cual la integración no resulta complicada.
- No se limita a un solo formato de datos.
- Los servicios REST sufren un menor acoplamiento que los SOAP.
- Con un simple framework se puede realizar varios trabajos.
- REST es auto descubrible. No necesita publicar en WSDL.

2.1.12. Herramientas de desarrollo

Net Framework (ver Ilustración 3) es una biblioteca de clases que ofrece muchas capacidades para la creación de aplicación en C#. Contiene miles de valiosas clases precompiladas que han sido probadas y ajustadas para maximizar el rendimiento [26]. Las aplicaciones de software creadas con .NET se ejecutan en equipos con el sistema operativo Windows. En .NET Framework, el código se compila en un lenguaje intermedio llamado CIL (anteriormente MSIL) y reside en ensamblajes que pueden ser bibliotecas (DLL) o procesos (EXE). Los ensamblados se cargan y ejecutan dentro de la máquina virtual de (CLR) y se conocen como código administrado [27]. Algunas de las clases de .NET son: Collections.Generic, Linq, Web, Web.UI, Web.UI.WebControls.

2.1.12.1. IDE Visual Studio

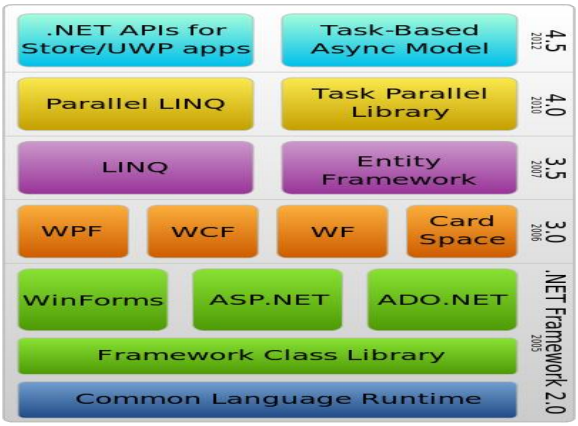
Permite crear, ejecutar y depurar aplicaciones escritas en C# y en otros lenguajes de programación .NET. Provee un conjunto de herramientas de desarrollo para la construcción de software: Web ASP, servicios Web RESTFUL [28].

2.1.12.2. Web ASP.NET

Dentro de .net existe dos marcos para el trabajo en ambiente web: ASP.NET web forms y ASP.NET MVC. Web forms comparte muchos conceptos de Windows forms. Con Web Forms, Microsoft introdujo una serie de mecanismos que tenían como objetivo convertir las

páginas web en componentes con estado. Entre estos mecanismos se encuentran los controles de servidor, que abstraen la representación del HTML para el desarrollador y proporcionan un enfoque basado en eventos para responder a la interacción del usuario, así como funciones útiles para leer y actualizar el estado de una página web [27]. Algunas de las ventajas de ASP.NET son su fácil uso, parecido a Windows forms y admite componentes de terceros, como controles de servidor para crear gráficos.

Ilustración 3 . Pila de componentes de .Net Framework



FUENTE: [29]

2.1.12.3. ASP.NET Master Pages

Son páginas que representan una especie de plantilla. Tienen la misma estructura que las páginas típicas. Sin embargo, viven en archivos que usan la extensión .master. Una página maestra sirve como una plantilla que presenta una apariencia común a todas las páginas basadas en ella. Las páginas maestras utilizan etiquetas de documento XHTML (como <html>, <head> y <body>), que se aplican solo a la página maestra [29].

2.1.12.4. LINQ

Permite manipular los datos almacenados en una base de datos relacional, como SQL Server. Mientras que **ADO.NET Entity Framework (EF)** permite que las aplicaciones interactúen con datos en varias formas, incluidos los datos almacenados en bases de datos relacionales. Funciona con los sistemas de administración de bases de datos más populares. Detrás de

escena, el Entity Framework de ADO.NET genera sentencias de SQL que interactúan con una base de datos [26]. Métodos útiles:

- Enumerable.OrderByDescending: Ordena los elementos de una secuencia en orden descendente.
- Enumerable.FirstOrDefault: Devuelve el primer elemento de una secuencia o un valor predeterminado si no se encuentra ningún elemento.
- Enumerable.Where: Filtra una secuencia de valores basada en un predicado.
- Enumerable.ToList(IEnumerable<TSource>): Crea una List<T> a partir de un IEnumerable<TSource>.

2.1.12.5. Newtonsoft

Es una biblioteca para serialización y deserialización de JSON. Se utiliza para la lectura de JSON, divide en las tres clases principales: JsonReader, JsonWriter y JsonSerializer [30].

Beneficios:

- Flexible serializador JSON para convertir entre objetos .NET y JSON.
- Alto rendimiento y código libre.
- Compatible con .NET 2, .NET 3.5, .NET 4, .NET 4.5, Silverlight, Windows Phone y Windows 8 Store.

Métodos útiles:

- JObject: Inicializa una nueva instancia de la clase JObject.
- JProperty: Inicializa una nueva instancia de la clase JProperty.
- JArray: Inicializa una nueva instancia de la clase JArray desde otro objeto JArray.

2.1.12.6. Mantnet.Numerics

Tiene como objetivo proporcionar métodos y algoritmos para cálculos numéricos en ciencias, ingeniería y uso diario. Los temas cubiertos incluyen funciones especiales, álgebra

lineal, modelos de probabilidad, números aleatorios, interpolación, integración, regresión, problemas de optimización. Disponible de forma gratuita bajo la licencia MIT / X11 [31].

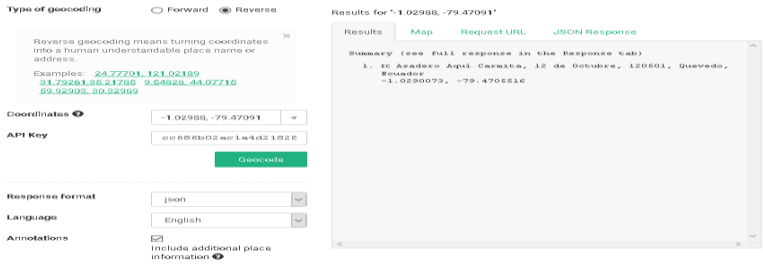
Métodos útiles:

- Spearman(IEnumerable<double> dataA, IEnumerable<double> dataB): Calcula el coeficiente de correlación de Spearman.
- SpearmanMatrix(Double[][] vectors): Calcula la matriz de correlación clasificada de Spearman.
- Pearson(IEnumerable<double> dataA, IEnumerable<double> dataB): Calcula el coeficiente de correlación del momento del producto de Pearson.
- Matrix<T> PearsonMatrix(Double[][] vectors): Calcula la matriz de correlación de momentos de productos de Pearson.

2.1.12.7. OpenCage Geocoder

Proporciona geo codificación inversa (ver Ilustración 4). Entre sus funciones constan las de enviar coordenadas de latitud y longitud, y devuelve el texto (long) con la ubicación, esta implementado a través de una API REST [32].

Ilustración 4.API REST Para localizar punto por medio de coordenadas



FUENTE: https://opencagedata.com/demo

2.1.12.8. Map thunderforest

Proporciona estilos de mapas para aplicaciones y sitios web (ver Ilustración 5). Usa tecnología JavaScript. El estilo de mapa paisaje muestra características de naturaleza, es una exhibición perfecta para aquellos interesados en la naturaleza, el campo [33].

Ilustración 5. Vista de paisaje localidad Quevedo por Map Thunderforest

FUENTE: https://www.thunderforest.com/maps/neighbourhood/

2.1.12.9. SQL Server y Tipos (Datos)

Se define como un RDBMS que mantiene una arquitectura (C/S) que usa Transact-SQL para mandar peticiones entre un cliente y el SQL Server. Cada columna, variable local, expresión y parámetro tiene un tipo de dato relacionado (ver Tabla 8). Este tipo es un atributo que especifica el contenido que puede contener el objeto, los tipos pueden ser datos enteros, datos de caracteres, datos de moneda, datos de fecha y hora, cadenas binarias [34]. Cuando dos expresiones que tienen tipos de datos, intercalaciones, precisión, escala o longitud diferentes son combinadas por un operador, las características del resultado vienen determinadas por lo siguiente:

- El tipo de datos del resultado viene determinado por la aplicación de las reglas de precedencia de tipos de datos a los tipos de datos de las expresiones de entrada.
- La intercalación del resultado viene determinada por las reglas de precedencia de intercalación cuando el tipo de datos del resultado es char, varchar, text, nchar, nvarchar o ntext.
- La precisión, escala y longitud del resultado dependen de la precisión, escala y longitud de las expresiones de entrada [34].

Tabla 8. Categorías de tipos de datos de SQL SERVER

Categorías de los tipos de datos	
numéricos exactos	cadenas de caracteres unicode
numéricos aproximados	cadenas binarias
fecha y hora	otros tipos de datos

FUENTE: [34]

Según las características de almacenamiento, algunos tipos de datos están designados como pertenecientes a los siguientes grupos:

- Tipos de datos de valores grandes: varchar(max) y nvarchar(max)
- Tipos de datos de objetos grandes: text, ntext, image, varbinary(max) y xml

Tabla 9. Tipos de datos por categoría.

Numéricos exactos	
bigint	numeric
bit	smallint
decimal	smallmoney
int	tinyint
money	

FUENTE: [34]

Tabla 10. Tipos numéricos aproximados.

Numéricos aproximados	
float	real

FUENTE: [34]

Tabla 11. Tipos fecha y hora.

Fecha y hora	
date	datetimeoffset
datetime2	smalldatetime
datetime	time

FUENTE: [34]

Tabla 12. Tipos cadena de caracteres.

Cadenas de caracteres	
char	varchar
varchar (max)	

FUENTE: [34]

Tabla 13. Otros tipos de datos.

Otros tipos de datos	
cursor	rowversion
hierarchyid	uniqueidentifier
sql_variant	xml
tipos de geometría espacial	tipos de geografía espacial
table	

FUENTE: [34]

Tipo de dato cursor permiten recorrer el resultado de una consulta SQL columna por columna, donde se podría realizar operaciones en cada interacción. En [35] lo definen como tipos de datos para las variables o para los parámetros de resultado de los procedimientos almacenados que contiene una referencia a un cursor. Las operaciones a las que pueden hacer referencia las variables y los parámetros que tienen un tipo de datos cursor son:

- Las instrucciones DECLARE y SET.
- Las instrucciones del cursor OPEN, FETCH, CLOSE y DEALLOCATE.
- Los parámetros de resultado de procedimientos almacenados.
- La función CURSOR_STATUS.

Tipo de dato tabla se puede utilizar para almacenar un conjunto de resultados para procesar en otro momento, generalmente se utiliza para almacenar temporalmente un conjunto de filas devueltas como el conjunto de resultados de una función con valores de tabla. La variable tabla se implementan en funciones, procedimientos almacenados y lotes [36].

Funciones de agregado (ver Tabla 14) realizan cálculos sobre un conjunto de valores y devuelve un solo valor. Pero existen algunas excepciones como COUNT. Ignoran el valor NULL, usualmente se usan con GROUP BY de la instrucción SELECT [37].

Tabla 14. Detalle de funciones de agregado

Transact-SQL funciones de agregado	Devuelve
approx_count_distinct	El número aproximado de valores no nulos únicos de un grupo.
avg	El promedio de los valores de un grupo. Omite valores Null.
checksum_agg	La suma de comprobación de los valores de un grupo. CHECKSUM_AGG omite los valores NULL.
count	El número de elementos encontrados en un grupo. Tipo de dato int.
max	El valor máximo de la expresión.
min	El valor mínimo de la expresión.
stdev	La desviación típica estadística de todos los valores de la expresión especificada.
stdevp	La desviación estadística estándar para la población de todos los valores de la expresión especificada.
sum	La suma de todos los valores o solo de los valores DISTINCT de la expresión. Solo puede utilizarse con columnas numéricas. Omite valores Null.
var	La varianza estadística de todos los valores de la expresión especificada.
varp	La varianza estadística de la población para todos los valores de la expresión especificada.

FUENTE: [37]

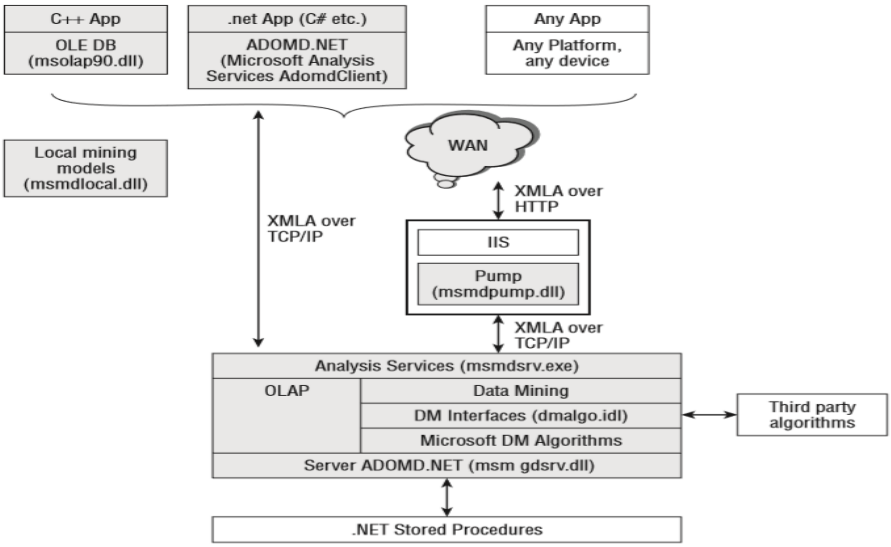
2.1.12.10. Arquitectura tecnológica servidor SQL Analysis Services

Es una herramienta de procesamiento analítico en línea (OLAP) y de minería de datos en Microsoft SQL Server. Su arquitectura corresponde a cliente/servidor, permitiendo a los clientes conectarse a través de una conexión LAN TCP/IP,o de IIS, utilizando HTTP a través de internet [9].

Cuando el servidor recibe una solicitud, Analysis Services determina si la solicitud es una solicitud de procesamiento analítico en línea (OLAP) o una solicitud de minería de datos, y enruta la solicitud de manera adecuada. Al final, una solicitud de minería de datos es

manejada por la infraestructura común de minería de datos o se transmite a un algoritmo de minería de datos, que puede ser un algoritmo creado por Microsoft [9].

Ilustración 6. Arquitectura de comunicación de Analysis Services.



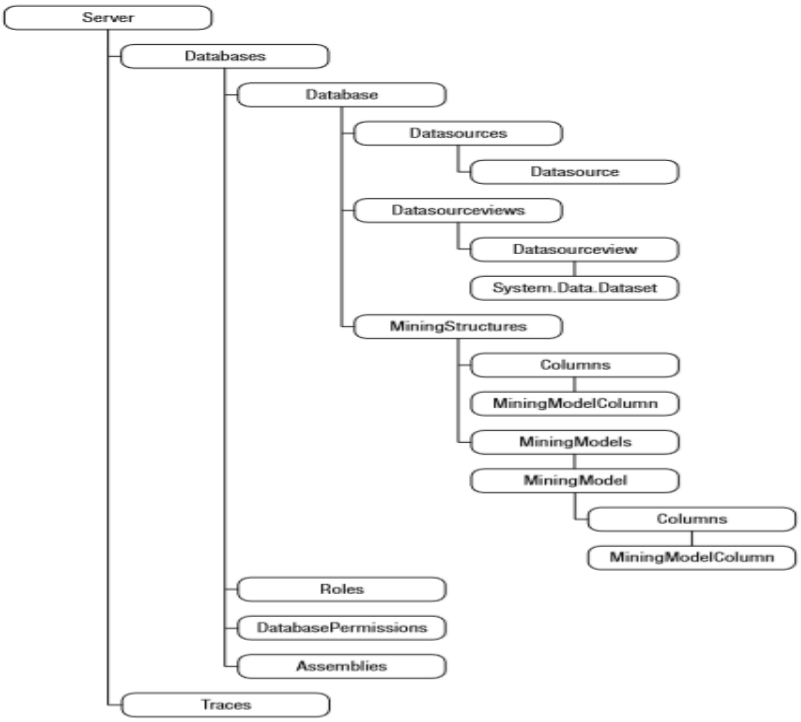
FUENTE: [9]

SSAS admite varias API para la comunicación, la utilización depende del nivel de operación y el entorno de programación [27]. A continuación, se detalla características de los marcos disponibles:

- **XMLA**: es un protocolo XML simple para comunicarse con servidores de análisis, independientemente de la plataforma de origen o destino.
- **MDX**: se define como un lenguaje de consulta OLAP y de objetos de minería comparable con SQL para bases de datos relacionales.
- **AMO**: es una interfaz de administración para Analysis Services que proporciona objetos para realizar operaciones como la creación, el procesamiento (ver Ilustración 7).
- **ADOMD.NET**: proveedor dedicado para Analysis Services. Funciona solo con Analysis Services y no usa OLE DB. Tiene la misma facilidad de acceso que ADO.NET, pero está optimizado para las operaciones de Analysis Services y ofrece varias clases e interfaces específicas. Proporciona acceso a los objetos de datos de Analysis Services desde lenguajes administrados como Visual Basic .NET, C # y J #. Proporciona un rico modelo de objetos (ver Ilustración 8) para explorar el

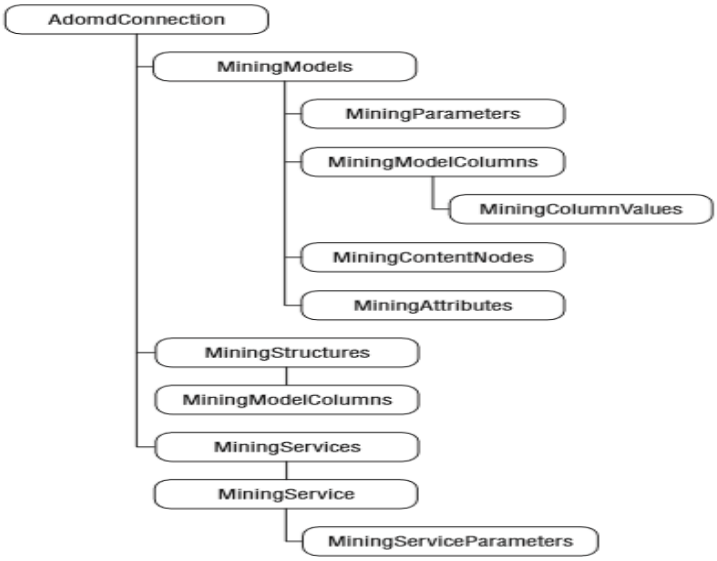
contenido y los metadatos de los objetos de minería de datos en un servidor que, de otro modo, son accesibles solo a través de conjuntos de filas de esquema [9].

Ilustración 7. Jerarquía de objetos de AMO



FUENTE: [9]

Ilustración 8. Principales objetos de minería de datos de ADOMD.NET



FUENTE: [9]

2.2. Marco referencial

Se ha investigado varios trabajos de tesis, aplicaciones web, investigaciones de diferentes partes del mundo, que presentan aspectos que aportan a esta investigación y que sirvieron de guía para la elaboración de la presente investigación. Los puntos siguientes detallan los trabajos investigados.

2.2.1. Clave taxonómica de identificación virtual de la colección de insectos acuáticos del departamento de biología de la Universidad Pedagógica Nacional

Realizado en Bogotá D.C (2013), trabajo de grado para la obtención del título de licenciado en electrónica, que tuvo como objetivo desarrollar una aplicación cliente-servidor que permita la búsqueda, clasificación e identificación de los insectos presentes en la colección de insectos acuáticos de la línea de investigación, biodiversidad y conservación de los ecosistemas acuáticos de la región andina (S.A.R.A), del departamento de biología de la Universidad Pedagógica Nacional [38].

El sistema realizado se basó en una arquitectura Cliente/Servidor, con un patrón de diseño MVC, desarrollado en PHP usando un framework llamado Yii el mismo que posee una herramienta llamada Gii que genera modelos de las tablas con sus respectivas llaves primarias y relaciones. Uso el método de desarrollo de software ágil llamada Scrum, el cual permitió una relación constante entre los dos departamentos, mostrando los avances de la aplicación y realizando las correcciones correspondientes a medida que se avanzaba en el proyecto [38]. En el trabajo se logró crear una clave taxonómica con todos los requerimientos biológicos pertinentes para la identificación de los insectos presentes en la colección de dicho departamento. A satisfacción de los dos departamentos participantes y en conformidad con las pruebas realizadas, el proyecto en su condición de software permite una identificación de cualquier insecto perteneciente a los órdenes Tricoptera y Coleóptera [38].

2.2.2. Pl@ntNet

Es un sistema web (ver Ilustración 9) y móvil con más de 135000 especies, 662744 imágenes y 13 proyectos. Esta aplicación está dedicada a la recopilación, anotación y recuperación de imágenes para ayudar en la identificación de plantas. Hace posible tomar fotos de una planta y compararlas con las de una base de datos de imágenes actualizada dinámicamente, para facilitar la identificación de esta planta. El resultado propuesto por la aplicación es, por lo tanto, evolutivo, ya que se enriquece de días a días de nuevas imágenes y nuevas especies gracias a las contribuciones de los miembros de la red [39]. Fue desarrollada por un consorcio formado por científicos de CIRAD, INRA, INRIA, IRD, y la red Tela Botánica en virtud de un proyecto financiado por la Fundación Agropolis.

Características:

- Comparte tus observaciones y colabora en la creación de listas de especies.
- Participa en la identificación de tus observaciones y las de otros usuarios en compañía de naturalistas expertos y aficionados.
- Organiza un evento en el que muchas personas se reúnan para encontrar la mayor cantidad de especies posible.
- Tus observaciones ayudarán a los científicos a conocer más sobre la biodiversidad y desarrollar investigaciones que promuevan su conservación.

Está escrito en Rails, el cual un framework de aplicaciones web escrito en el lenguaje de programación Ruby, que siga el paradigma del patrón Modelo Vista Controlador.

Ilustración 9. Sitio Web Pl@ntNet

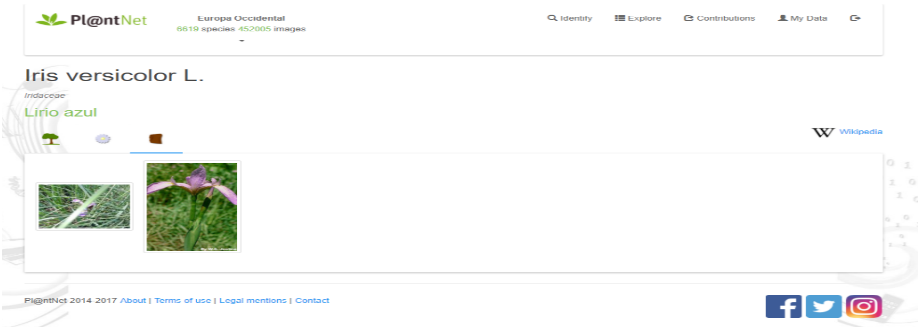
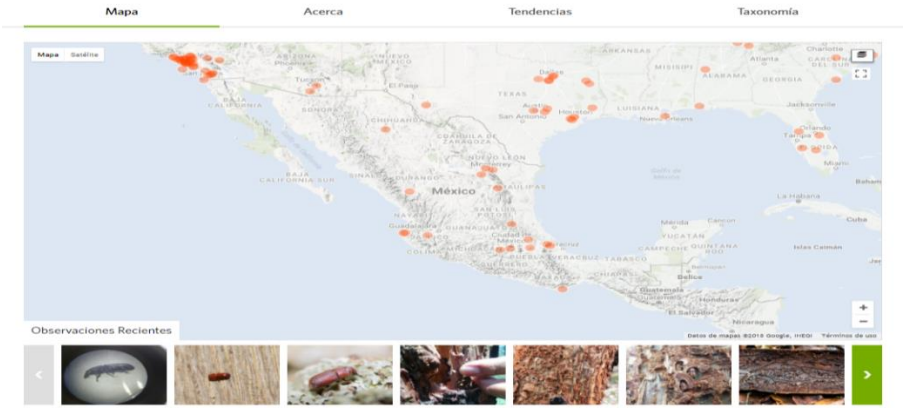


Ilustración 10. Sitio web Pl@ntNet página ubicación de insecto Scolytinae.



2.2.3. Una nueva herramienta de exploración y generación de informes de minería de datos basada en web para los que toman decisiones

Realizado en Turquía, investigación del Departamento de ingeniería Informática, Universidad Hacettepe Ankara. ASMINER es una herramienta para la generación de informes y exploración de minería de datos basado en la web. Permite explorar tres técnicas de minería (decision tree, clustering, roles association rules) de SSAS. Utilizaron ASP.NET 2.0 y Analysis Services por su soporte nativo. Los módulos y los componentes se detallan en la Ilustración 13.Como requisito de visualización, las tecnologías Adobe Flash, Microsoft Silverlight y Sun Java se emplean en el lado del cliente [8]. Puntos destacados del proyecto, para la resolución de problemas:

- **Rendimiento**: el ancho de banda, la cantidad de datos transportados se redujo mediante el uso de componentes eficientes y la arquitectura AJAX. Uso de componentes de visualización interactivos de pequeño tamaño escritos en Flash y Java, por su almacenamiento en el cache del navegador.
- **Escalabilidad**: Dado que la cantidad de datos puede aumentar a terabytes. Se selecciono a Analysis Services porque tiene la capacidad de manejar muchas solicitudes de clientes e incluso procesar modelos de minería al mismo tiempo en el lado del servidor al beneficiarse de los procesadores de varios núcleos, computación en grupo.
- **Visualización**: Para la exploración de correlación, gráficos de dependencia de red. La creación se llevó a cabo con GraphViz y para la visualización en la web el applet

de ZGRViewer. En el caso de exploración de árbol de decisión utilizaron la tecnología de scripting.

2.2.4. Darwin Core: Estándar para la gestión de datos biológicos primarios en la UTN

Proyecto realizado en la Universidad Técnica del Norte Ibarra, tuvo como objetivo mejorar el proceso de gestión de datos biológicos de los laboratorios de investigación biológica y ambiental de la UTN. Este objetivo se logró mediante la creación del sistema web SIGBIO que gestiona datos biológicos. El sistema se desarrolló bajo la metodología ágil SCRUM. La arquitectura tecnológica de la plataforma web se puede ver en la Ilustración 14. Unos de los resultados que obtuvieron, fue que al implementar la plataforma web se mejora de la gestión de datos biológicos primarios lo que permitió el intercambio de conocimientos, el enriquecimiento de la base de datos y el diálogo de saberes sobre el bioconocimiento; aumentó notablemente la centralización de la información, facilitando el registro y publicación de los datos en tiempo real y de acceso abierto beneficiando a la sociedad en general [40].

2.2.5. Diseño e implementación de una solución de almacenamiento de datos basada en la web para el análisis de costos

El proyecto presenta una aplicación web que ofrece a análisis de inteligencia empresarial para datos de facturas. Unos de los procesos que lleva a cabo para cumplir con los objetivos de la aplicación web es la extracción de datos de un sistema externo para luego transformar a una forma estructural que se carga en un cubo OLAP que ejecuta SSAS. Desarrollado mediante la metodología ágil Scrum, con iteraciones de dos semanas. Como motor de análisis SSAS, usando un cliente AMO. La capa de presentación se construyó utilizando ASP.NET web forms, controles de servidor integrados en el marco .NET y el componente RadXMLHttpPanel de la biblioteca Telerik UI. Al culminar el proyecto concluyeron que unos de los factores para que la investigación concluyera en el periodo esperado fue el uso de componentes de terceros y frameworks, esto minimizó las horas de trabajo de implementación, por lo tanto, el enfoque se dirigió al dominio del problema en lugar de problemas técnicos [27].

2.2.6. GBIF Global Biodiversity Information Facility

Es una red internacional e infraestructura de investigación financiada por gobiernos del mundo y destinada a proporcionar a cualquier persona, en cualquier lugar, acceso abierto a datos sobre todos los tipos de vida en la tierra. Proporciona almacenamiento de datos bajo estándares comuniones y herramientas de código abierto que permiten compartir información sobre dónde y cuándo se han encontrado las especies. Para unir todas las fuentes de datos GBIF usa el estándar Darwin Core, los editores brindan acceso abierto a sus conjuntos de datos mediante la designación de licencias de Creative Commons legibles por máquina, lo que permite a los científicos, investigadores y otros aplicar los datos en cientos de publicaciones y documentos de políticas.

Ilustración 11. Página Index Gbif.org

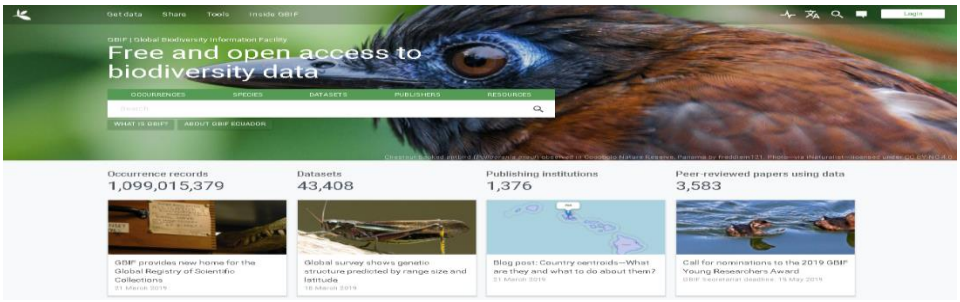
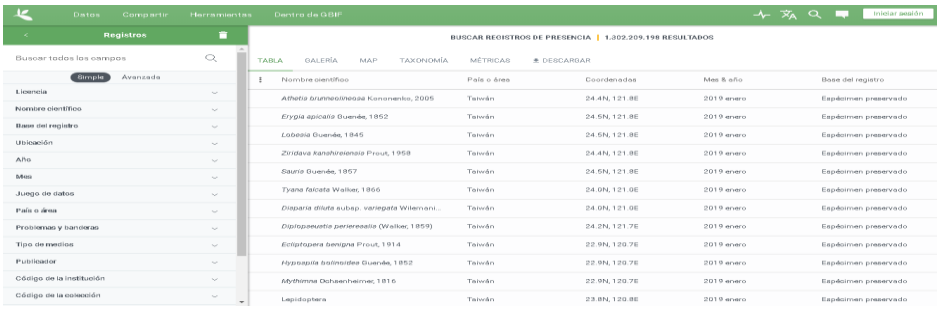


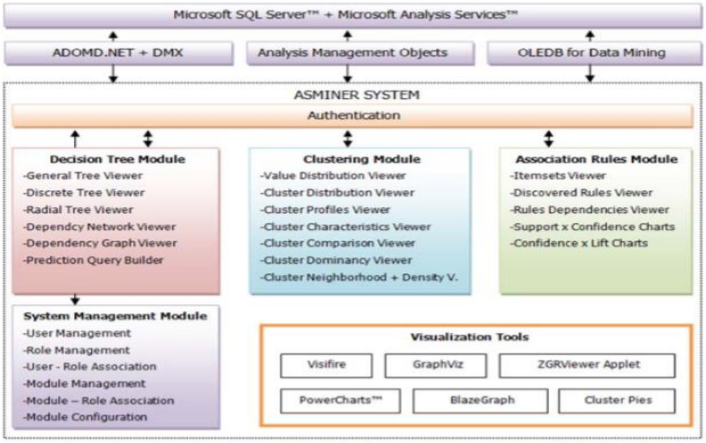
Ilustración 12. Página GBIF de búsqueda registros de presencia.



El sistema web está montado sobre un servidor Express para Node.js 7.10.0, para la presentación utilizan Nunjucks por su gran alcance con herencia de bloque, auto escapé, macros, control asíncrono, los CSS fueron construidos utilizando el precompilador Stylus y los complementos de PostCSS. En la arquitectura utilizan controladores, modelo, vista y una API que proporciona los datos, por otro lado, se aseguran de mantener la vista (pantalla) sin lógica compleja. La estructura del proyecto es bien definida con módulos back y front end y

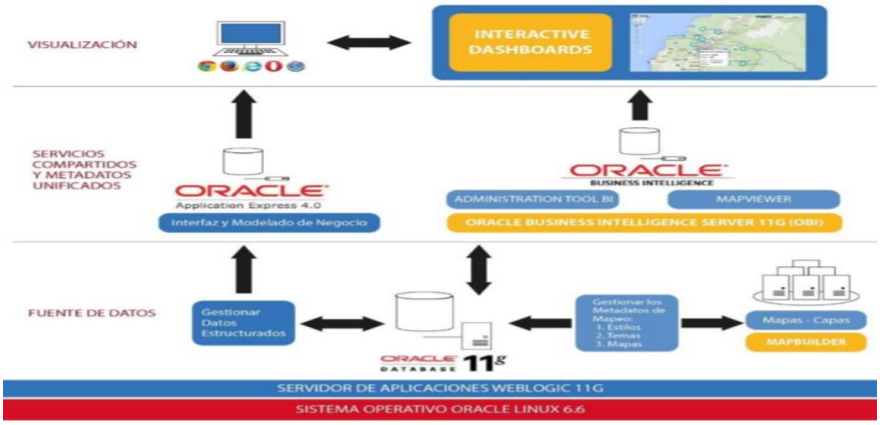
clases para controlar errores, express middlewares que procesan las solicitudes entrantes antes de enviarlas a los controladores.

Ilustración 13. Detalle módulos y componentes de ASMINER.



FUENTE: [8]

Ilustración 14. Arquitectura Tecnológica de la plataforma web para la gestión datos bilógicos.



ELABORADO POR: ORTEGA, GUEVARA

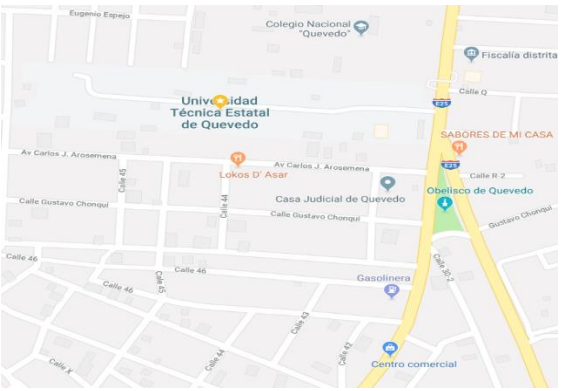
FUENTE: UTN

CAPÍTULO III
METODOLOGÍA DE LA INVESTIGACIÓN

3.1. Localización

El proyecto de investigación se implementó en la UTEQ, que se encuentra ubicada exactamente en el campus "Ingeniero Manuel Agustín Haz Álvarez", Av. Quito km. 1.5 vía a Sto. Domingo de los Tsáchilas.

Ilustración 15. Ubicación de UTEQ por Api Google Map



3.2. Tipos de Investigación

En el proyecto se utilizó dos tipos de investigaciones bibliográfica y documental, las cuales se exponen a continuación:

3.2.1. Bibliográfica

La investigación bibliográfica suministró los conocimientos, conceptos de investigaciones relacionadas al tema que se estudia, el estudio de los libros, sitios web oficiales, investigaciones permitió conocer y comprender como es la clasificación de los insectos, que tipos variables son relevantes y cuáles son los análisis necesarios cuando se requiere investigar el comportamiento de población, fluctuación, diversidad.

3.2.2. Descriptiva

El objetivo de la investigación descriptiva consiste en llegar a conocer las situaciones, costumbres y actitudes predominantes a través de la descripción exacta de las actividades, objetos, procesos y personas.

La utilización de este tipo de investigación se debe a las siguientes razones, investigar y conocer el proceso típico de almacenamiento de información física de los documentos de los estudios entomológicos e identificar las actividades que siguen los investigadores al momento de registrar al espécimen y de proceder a realizar cálculos de estos. Estas características sirvieron como guías para determinar que diseño y tecnologías son idóneas para el proyecto.

3.3. Métodos de investigación

En el proyecto de investigación se utilizó dos métodos bibliográfica-documental, analítico y la metodología de desarrollo de software llamada Proceso Unificado Racional, los cuales se exponen a continuación:

3.3.1. Bibliográfica-documental

Se estudió información de sitios web oficiales, libros, módulos, tesis propias del alma mater, revistas con el objetivo de tener una base de información teórica que sustente la implementación de las tecnologías de internet. Revisar investigaciones relacionadas, ayudó a descubrir componentes esenciales y tecnologías aplicables al proyecto. A parte de esto contribuyó a brindar soluciones prácticas a situaciones q se presentaron como fue el caso de mostrar un modelo de minería de Analysis Services en el ambiente web.

3.3.2. Analítica

Con el método analítico se revisó y analizó las funciones que realizan los investigadores de la UTEQ al momento de realizar las investigaciones de los insectos coleópteros, se observó la información recopilada e identificó cuáles son los datos de relevancia según el tipo de investigación y los cálculos estadísticos que realizan. Se tomó como base los cálculos estadísticos que realizan para determinar la librería estadística adecuada para implementar en la aplicación.

3.3.3. Método de desarrollo: Proceso unificado racional (RUP)

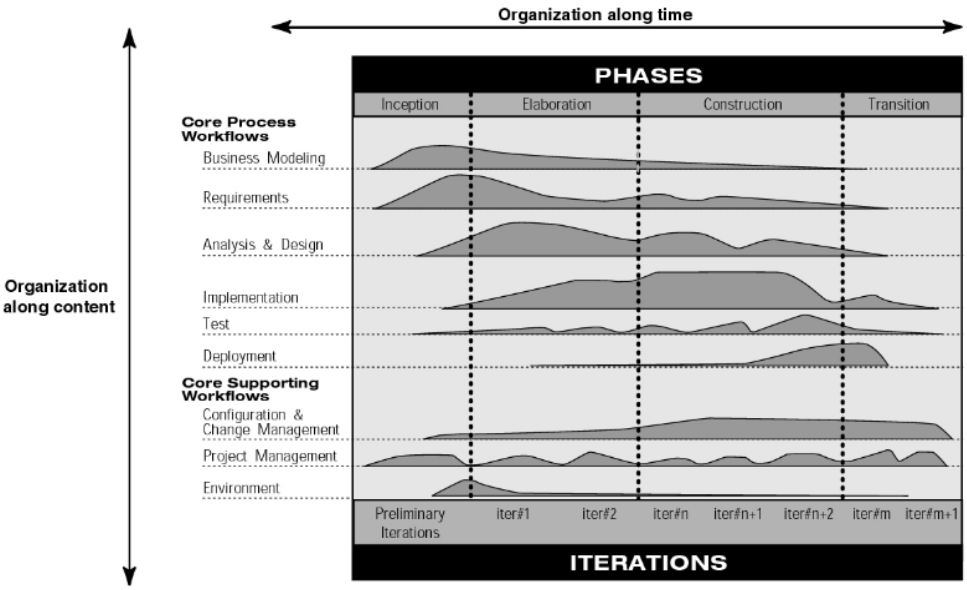
Se utilizó para el desarrollo la metodología RUP por el motivo de que el proyecto de investigación tiene ciertos requerimientos que son la implementación en dos ambientes web y móvil los cuales tienen diferentes tipos de patrones de diseño de software (MVC para web, MVP para móvil y REST para comunicar el cliente móvil con el sistema) la metodología RUP tiene un alto nivel de abstracción lo quiere decir que motiva el uso de conceptos reutilizables tales como patrones de diseño de software y frameworks [41].

Además, tiene como característica incluir artefactos (que son los productos tangibles del proceso como, por ejemplo: documento, el modelo de casos de uso, el código fuente) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso) [41].

Esta metodología tiene las siguientes fases (ver Ilustración 16):

- **Inicial**: La cual está diseñada para definir y acordar el alcance del proyecto con los patrocinadores y analizar cuál es la arquitectura que se ajusta al proyecto mediante el uso de artefactos como modelo de dominio, reglas de negocios, especificación de los casos de uso de negocio.
- **Elaboración**: En esta fase se plantean las actividades necesarias y lo recursos necesarios; esto permite definir las arquitecturas base del sistema que se desarrollaran, mediante uso de artefactos como diagramas.
- **Construcción**: En esta fase se llega a obtener la capacidad operacional inicial (prototipo), mediante el uso de artefactos como diagrama de secuencias, diagrama de clases, diagrama de despliegue.
- **Transición**: La finalidad de esta fase es ajustar errores, defectos encontrados en las pruebas y capacitar a los usuarios.

Ilustración 16. Fases de RUP Modelo iterativo muestra cómo el proceso se estructura a lo largo de dos dimensiones



FUENTE: [20]

3.4. Fuentes de recopilación de información

3.4.1. Fuentes primarias

Se dialogó con la Ing. Malena Martínez docente del área de entomológica de la UTEQ y la Ing. Joselyn Averos graduada de la Universidad, donde se intercambió ideas del proceso que se realiza al plantear un estudio y de los datos necesarios de los mismo.

3.4.2. Fuentes secundarias

La tesis de la UTEQ denominada "FLUCTUACIÓN POBLACIONAL DE Premnobius cavipennis Eichhoff (COLEOPTERA; CURCULIONIDAE; SCOLYTINAE) EN PLANTACIONES DE BALSA [Ochroma pyramidale (Cav. Ex. Lam) Urb] EN LA ZONA CENTRO DEL LITORAL ECUATORIANO" de Joselyn Averos, se usó para proveer a la aplicación web de datos; los mismos que fueron proporcionados por la autora. Otra tesis utilizada fue DIVERSIDAD DE ESCOLÍTIDOS (Coleoptera: Curculionidae: Scolytinae: Xyleborini), EN BOSQUE NATURAL Y EN PLANTACIONES DE BALSA (Ochroma

pyramidale) de Luis Cedeño. Se puede evidenciar tablas de registros de observaciones en el **anexo 4** registros de investigaciones entomológicas.

Otra fuente de información necesaria para cumplir con los objetivos planteados, fueron sitios web oficial como Math.NET Numerics que es una librería matemática que proporciona documentación de los métodos y algoritmos para cálculos numéricos en ciencias, ingeniería; esta librería está disponible de forma gratuita bajo licencia MIT/X11.

Además de lo mencionado se usó libros electrónicos, artículos para recopilar información esencial que ayudo al desarrollo del proyecto.

3.5. Recursos y Materiales

Tabla 15. Recursos humanos.

Nombre	Función	Labores
Kevin Manuel Onofre Wong	Desarrollo	Análisis requerimientos, Codificación.
Ing. Washington Alberto Chiriboga Casanova	Director de Proyecto	Análisis requerimientos, diseño de arquitectura,
Ing. Joselyn Averos Suarez	Consultora Forestal	Requerimientos, pruebas de usuario.

ELABORADO POR: AUTOR

Tabla 16. Recursos hardware.

Equipo	Características	COSTO UNITARIO	TOTAL
DELL INSPIRON 15	Procesador Intel(R) Core (TM) I7-7500U CPU @ 2.70GHz 8GB de RAM 250 GB SSD de disco duro	$692.00	$692.00
Celular HTC Bolt	Procesador Snapdragon 810	$160.00	$160.00

	3GB de RAM		
Unidad flash SanDisk Cruzer CZ36 16 GB USB 2.0	Pendrive SanDisk cz36, 16 GB	$12.00	$12.00
	Total de recursos de hardware:		$864.00

ELABORADO POR: AUTOR

Tabla 17. Recursos software.

Nombre	Descripción	Costo unitario	Total
SO Windows 10 Home 64 bits	Administra los recursos del Sistema	$0.00	$0.00
IDE Microsoft Visual Studio 2015 versión 14.0.25431.01 Update 3	IDE gratuito que proporciona los recursos para un desarrollo software	$0.00	$0.00
SO Android versión 4.1.2	Administra los recursos del sistema móvil	$0.00	$0.00
IDE Android Studio versión 3.0	IDE gratuito que proporciona los recursos para un desarrollo software	$0.00	$0.00
Postman versión 5.5.3 Chrome 70.0.35	Aplicación para probar servicios servicios Web REST.	$0.00	$0.00
Firefox Developer Edition versión 64.0b2	Navegador web que ofrece una gama de herramientas para el desarrollo y funciones de soporte de CSS Grid y depuración de marco.	$0.00	$0.00

Sublime Text version 3.1.1	Editor de código fuente de múltiples lenguajes, es multiplataforma.	$0.00	$0.00
Microsoft Office 2016	Editor de archivos electrónicos, la laptop utilizada para el desarrollo cuenta con licencia.	$0.00	$0.00
Total de recursos de software:			$0.00

ELABORADO POR: AUTOR

Tabla 18. Presupuesto total.

Tipo recurso	Costo total
Hardware	$ 864.00
Software	$ 0.00
Subtotal	$ 864.00
Total	$ 864.00

ELABORADO POR: AUTOR

Los costos detallados en la tabla 15 fueron asumidos por el autor del proyecto.

CAPÍTULO IV
RESULTADOS Y DISCUSIÓN

4.1. Resultados de la metodología RUP

4.1.1. Fase inicial

En esta fase se logró obtener los requerimientos y otros componentes necesarios. Los requerimientos fueron recopilados durante los diálogos con la Ing. Martínez y la consultora Ing. Joselyn Averos. Adicionalmente se incluyeron requerimientos obtenidos por el análisis de los registros de investigaciones entomológicas. Otros aspectos y funcionalidades fueron considerados para el sistema, por las aplicaciones mencionadas en la sección de marco referencial. Detalle de los requisitos:

- La aplicación debe permitir el mantenimiento de los perfiles de usuario. El perfil que tenga cada usuario abrirá acciones permitidas y restringirá el acceso a las páginas.
- La aplicación debe permitir control de acceso mediante uso de credenciales validas, previamente adquiridas.
- La aplicación debe permitir al investigador gestionar un proyecto de estudio (crear, modificar, eliminar)
- Cada proyecto de estudio debe especificar los respectivos datos que se recolectaran.
- La aplicación debe permitir establecer la privacidad (Privado o público) del proyecto.
- La aplicación debe permitir almacenar la información de la recolecta en campo y modificar en cualquier momento en el laboratorio.
- Permitir georreferenciar el lugar de la observación.
- Permitir capturar imágenes relacionadas a la observación.
- Permitir consultar información de la recolecta.
- La aplicación debe permitir resumir la información de la recolecta.
- La aplicación debe permitir realizar cálculos estadísticos a la información recolectada y mostrar los cálculos de manera simple.
- La aplicación deberá divulgar la información del proyecto en formatos simples como archivos formato Excel.

Requisitos no funcionales:

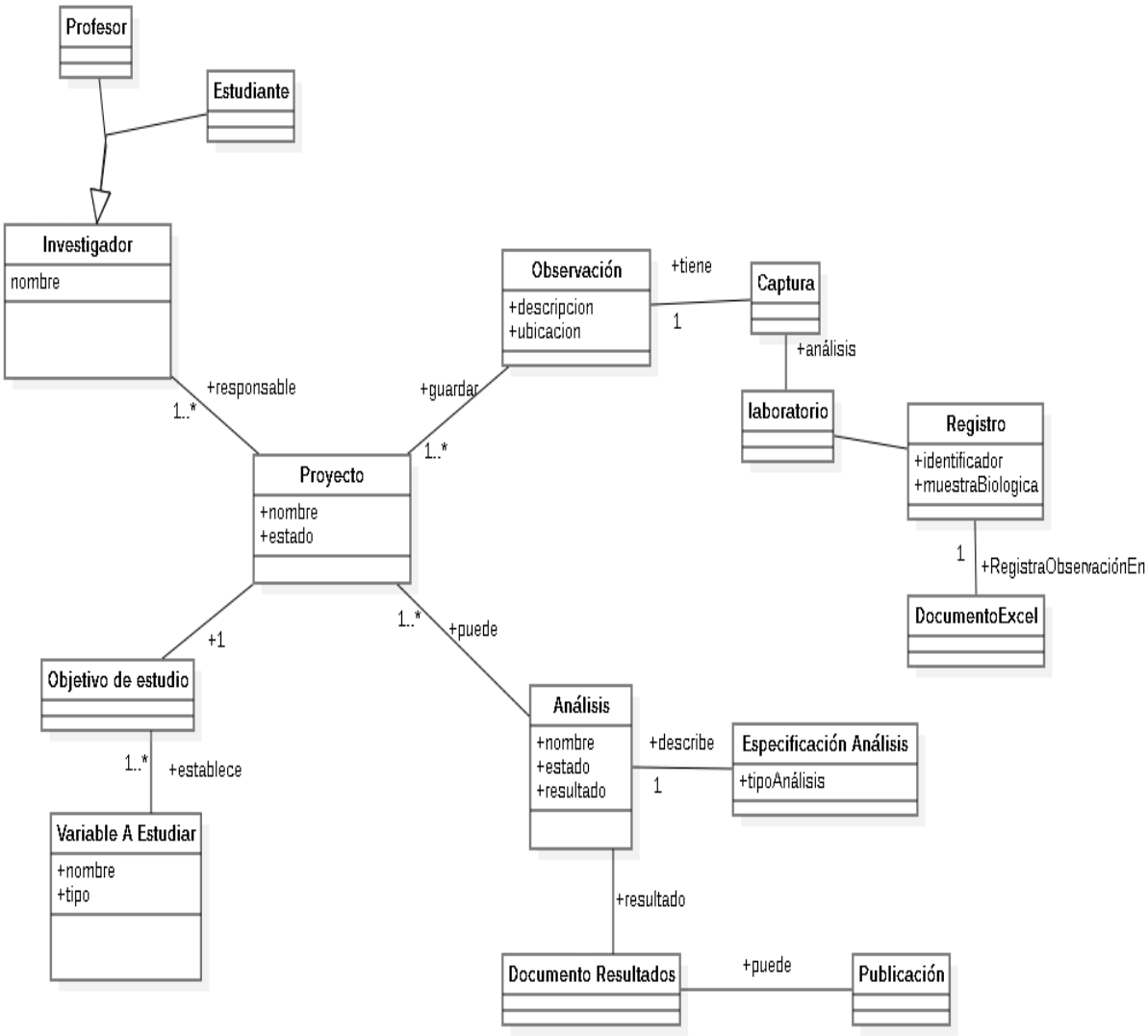
- La aplicación debe tener una interfaz limpia y usable.

- La aplicación deberá ser mantenible ya que el sistema es propenso a ser modificado por demanda del cliente.
- La aplicación será accesible desde cualquier equipo que utilice navegador Web Google Chrome y Mozilla Firefox.
- Las tecnologías que integren el sistema deben ser compatibles con ASP.NET.

4.1.1.1. Modelo conceptual o dominio

El modelo de dominio representa las clases conceptuales significativas en un dominio del problema, no representa componentes de software.

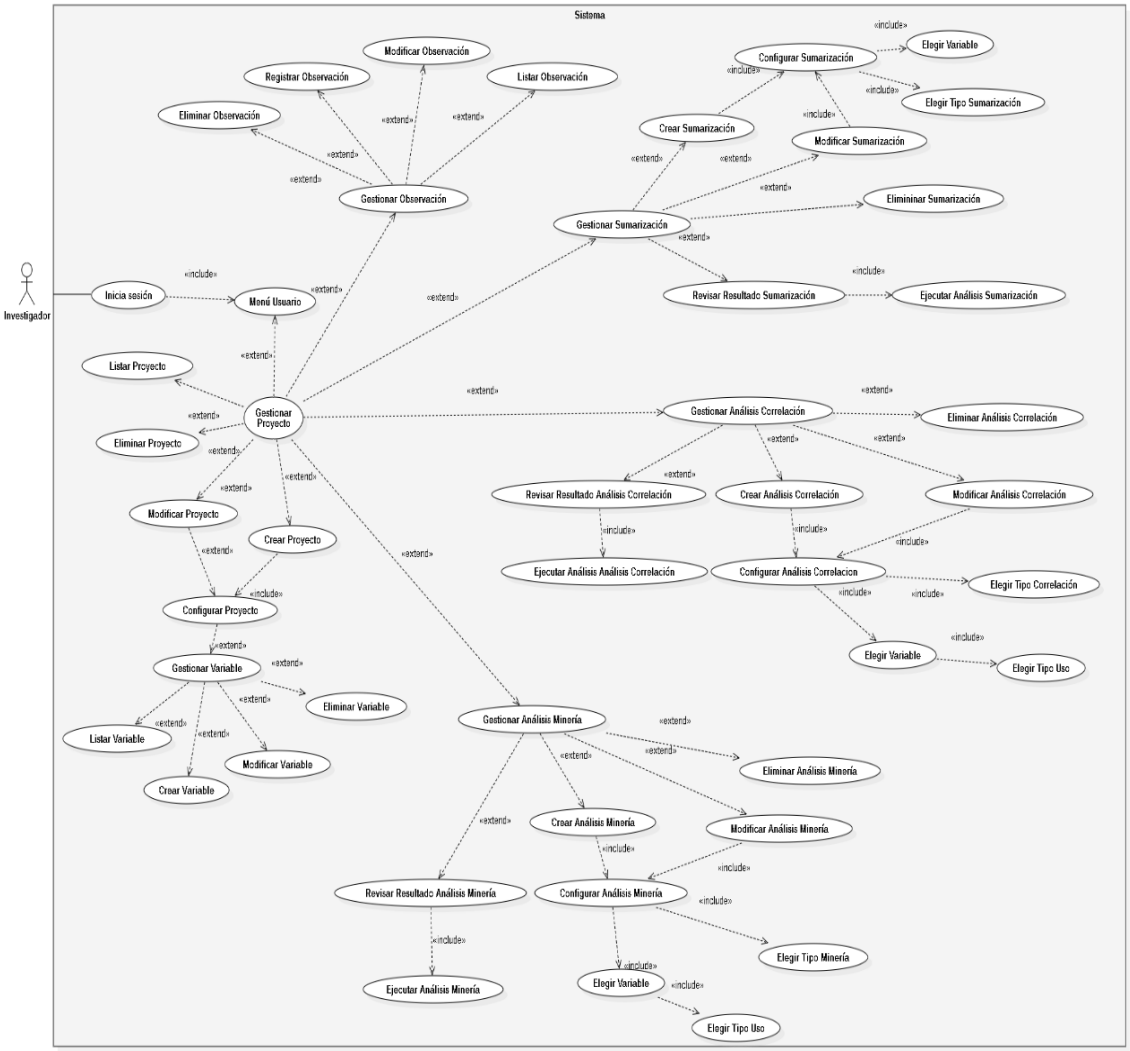
Ilustración 17. Modelo de dominio del problema.



FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

4.1.1.2. Caso uso usuario (Investigador)

Ilustración 18. Caso de uso usuario investigador.



FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

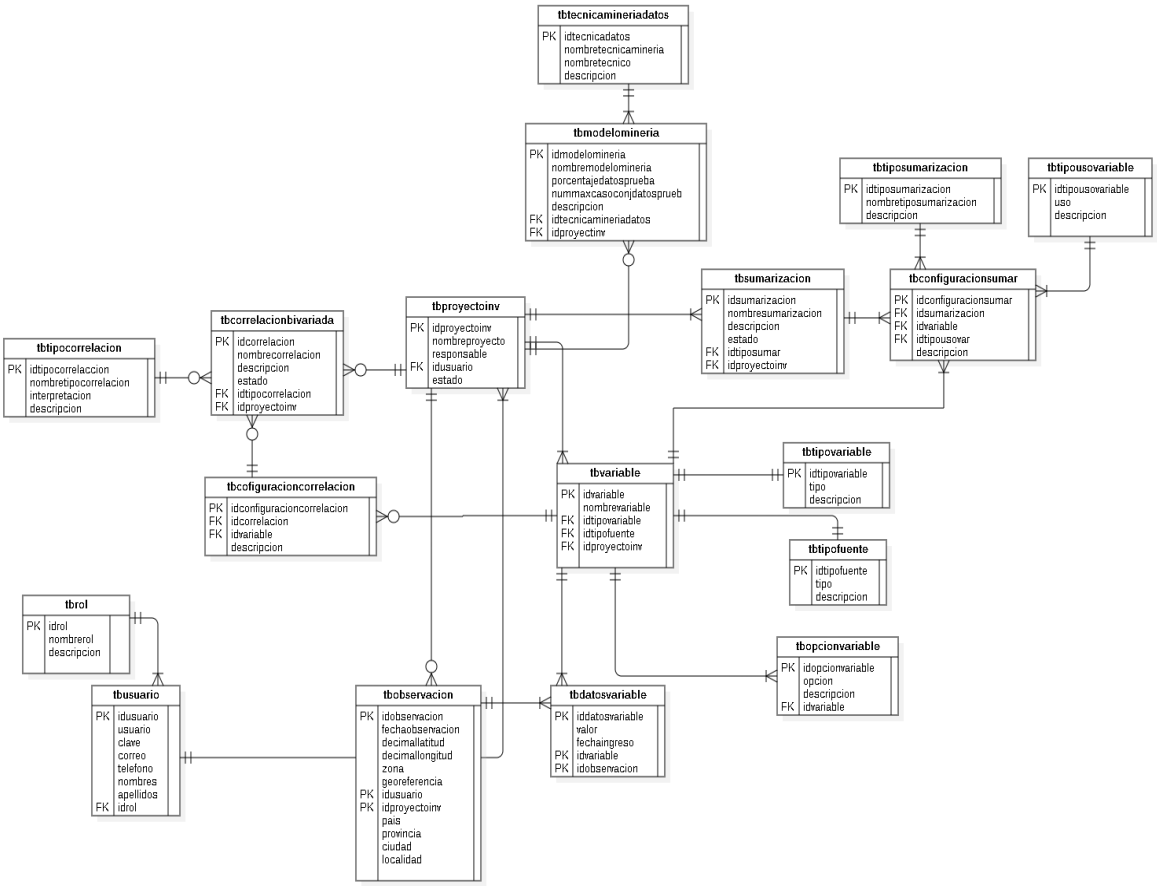
4.1.2. Fase elaboración

En esta fase se plantean las actividades necesarias y lo recursos necesarios; esto permite definir las arquitecturas base del sistema.

4.1.2.1. Modelo de datos

El **diagrama entidad-relación** presenta las entidades para las operaciones del sistema y las relaciones entre ellas.

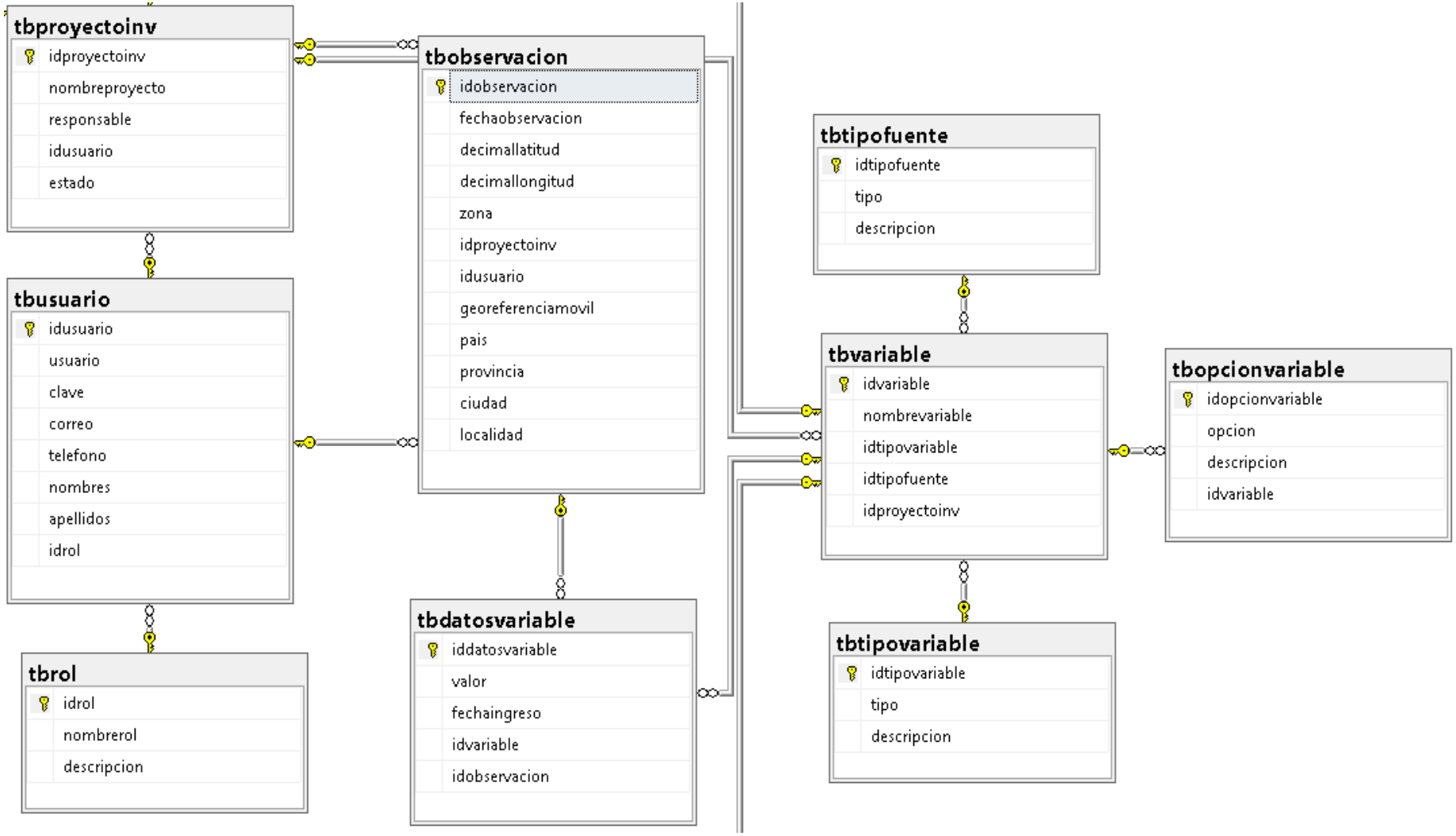
Ilustración 19. Diagrama entidad relación.



FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

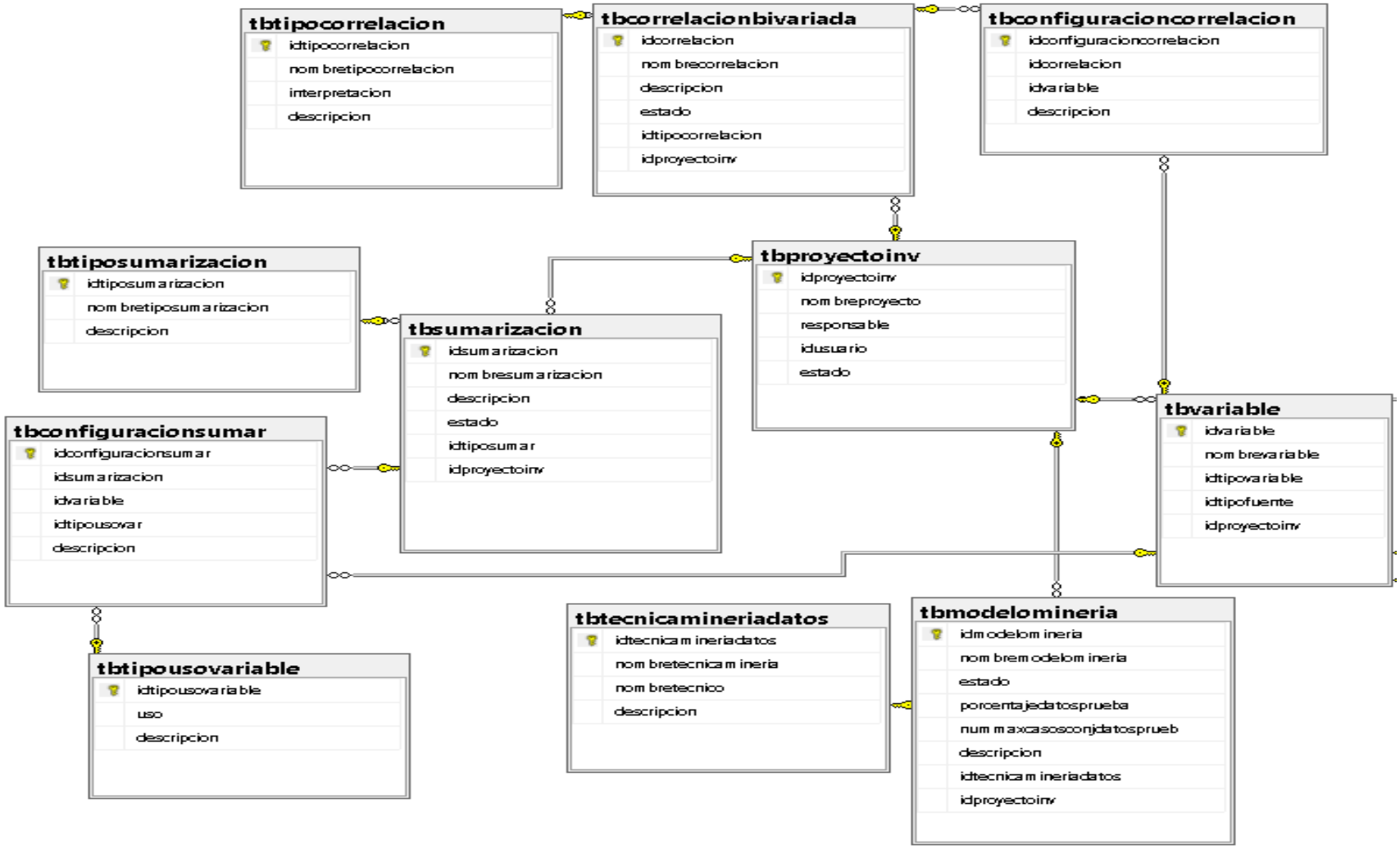
El **modelo físico de la base de datos** se creó a partir de las entidades y relaciones del diagrama entidad-relación. Para un mejor entendimiento del modelo físico se divide en dos partes la Ilustración 20 primera usuarios (rol, usuario), entrada de datos (proyectos, observaciones) y Ilustración 21 segunda (configuración de análisis).

Ilustración 20. Parte I. Base de datos Física de PIentomología.



FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

Ilustración 21. Parte II. Base de datos Física de PIentomología.



FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

Diccionario de datos se lo puede evidenciar en el **anexo 2**.

4.1.2.2. Clases del sistema

Ilustración 22. Clase usuario relacionada con rol.

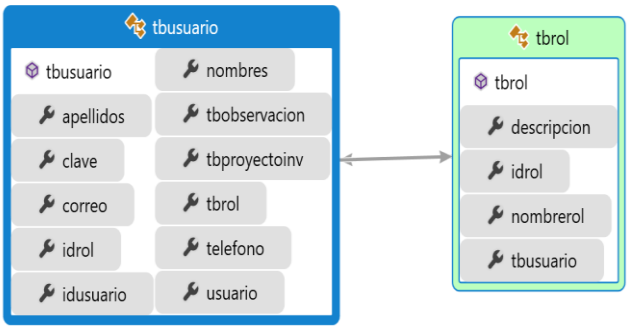


Ilustración 23. Clase proyecto relacionada con usuario y variable.

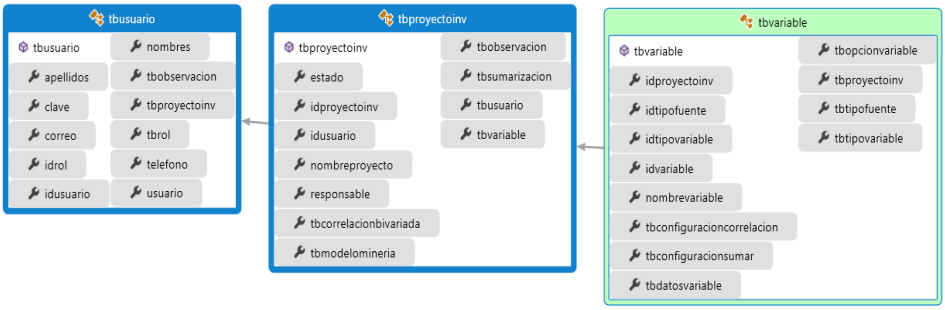


Ilustración 24. Clase variable relacionada con opción variable, tipo variable, tipo fuente.

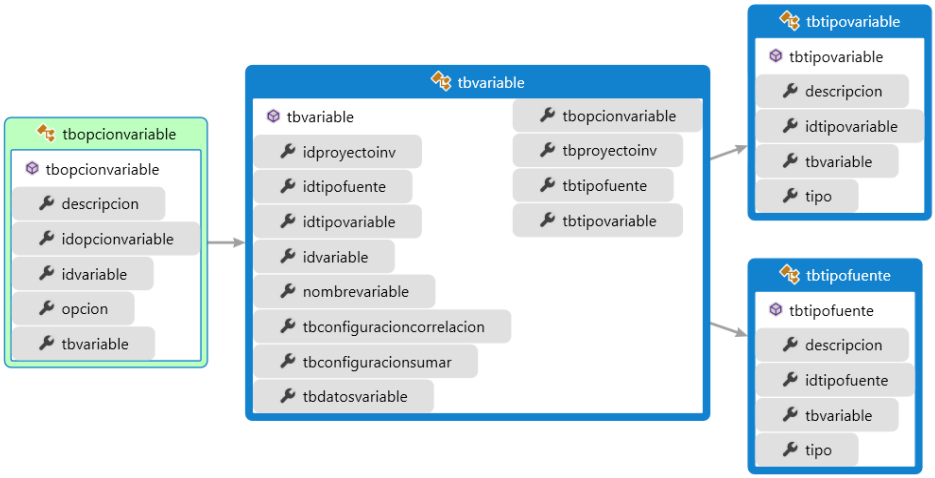


Ilustración 25. Clase datos variable relacionada con observación y variable.

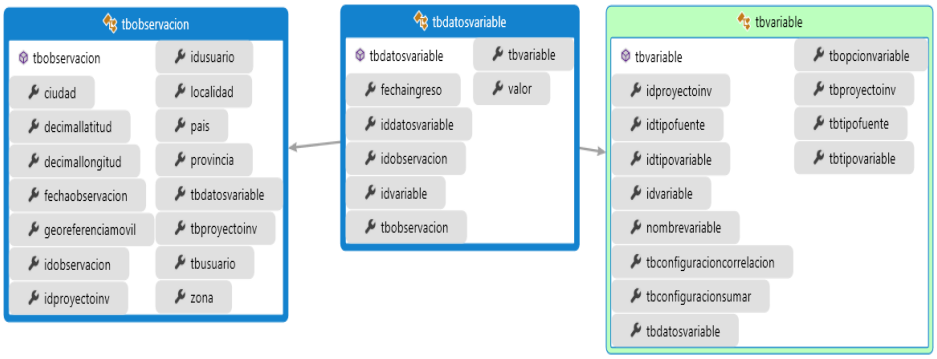


Ilustración 26. Clase sumarización relacionada con configuración.

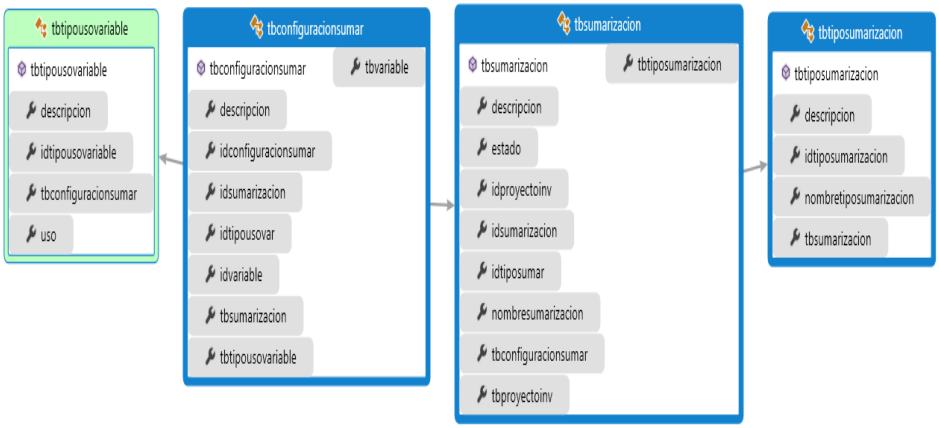


Ilustración 27. Clase correlación.

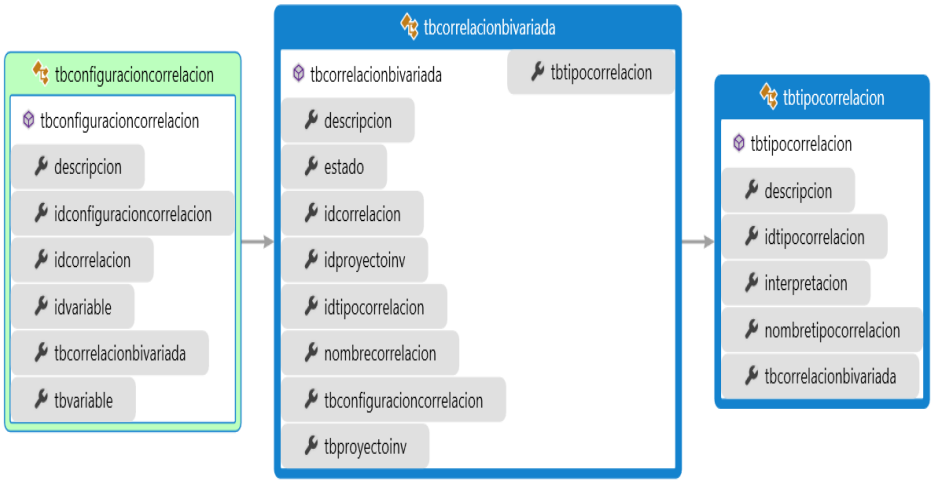


Ilustración 28. Clase modelo de minería.

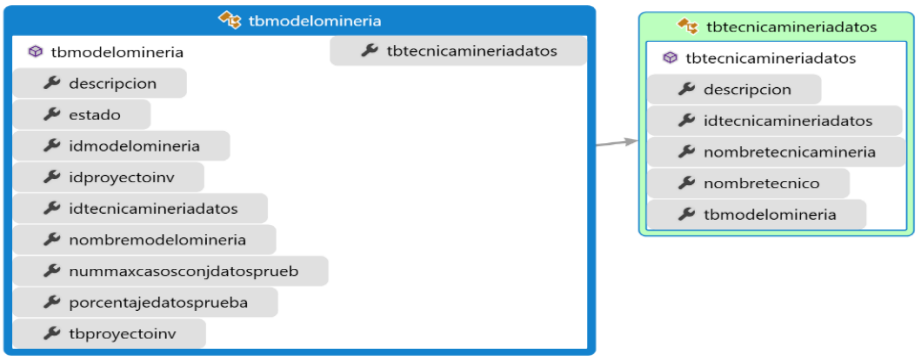


Ilustración 29. Clases para minería de datos, base de datos, fuente de datos, vista de fuentes datos.

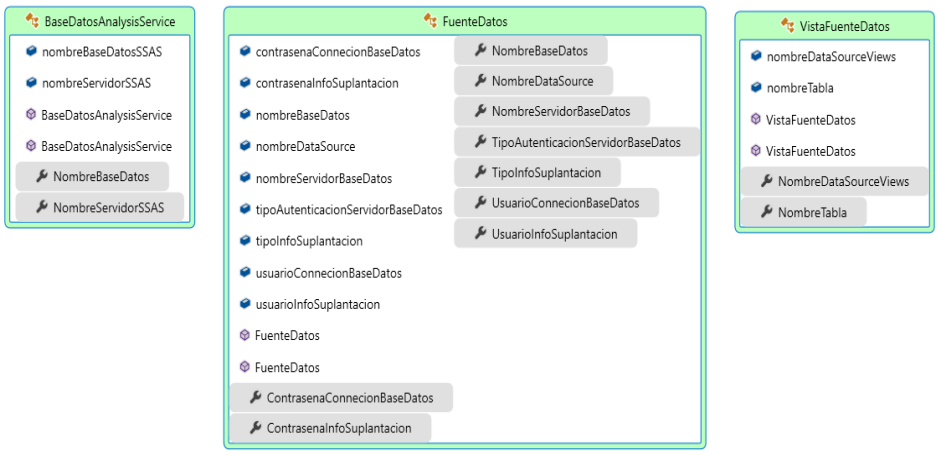


Ilustración 30. Clases de minería de datos, estructura de minería, columna estructura de minería.

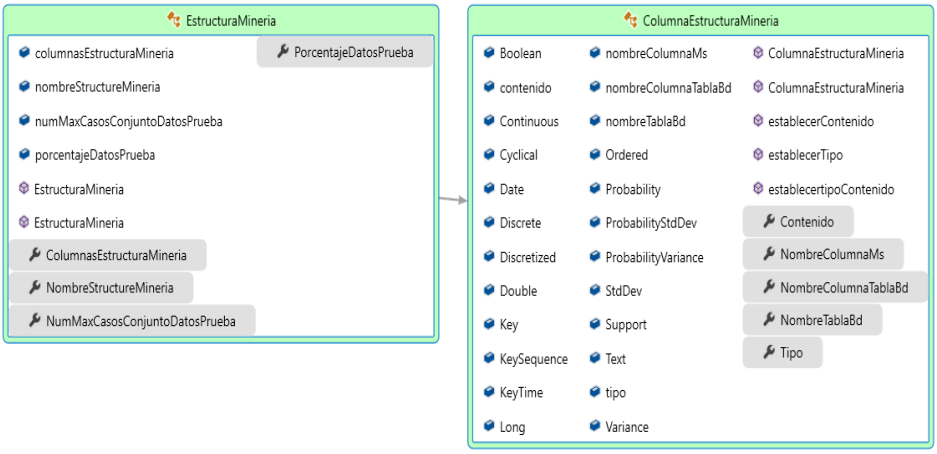
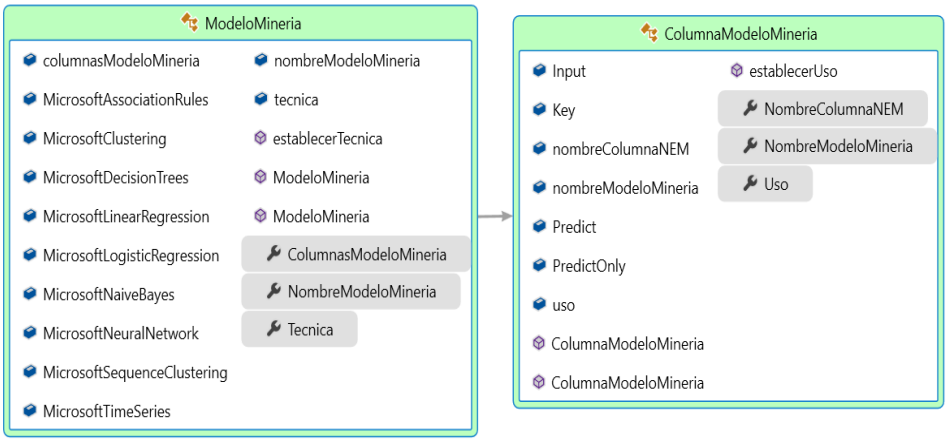


Ilustración 31. Clases de minería de datos, modelo de minería, columna modelo de minería.



4.1.2.3. Casos de uso extendido

Representa una descripción de alto nivel de las acciones que realiza el usuario contra el sistema.

Tabla 19. Caso de uso iniciar sesión.

Caso de Uso:	Iniciar sesión
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Ingresar al sistema PIentomología en modo de investigador.
Resumen:	Usuario inicia sesión con sus credenciales para entrar al modo de investigador.
Precondiciones:	1.-Usuario abre la aplicación PIentomología. 2.-Administrador creo usuario y otorgo credenciales al investigador.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra la interfaz para autenticar credenciales del usuario.
2.-Ingresa nombre de usuario y clave.	

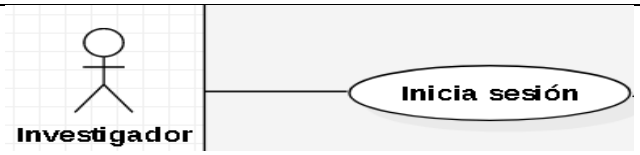
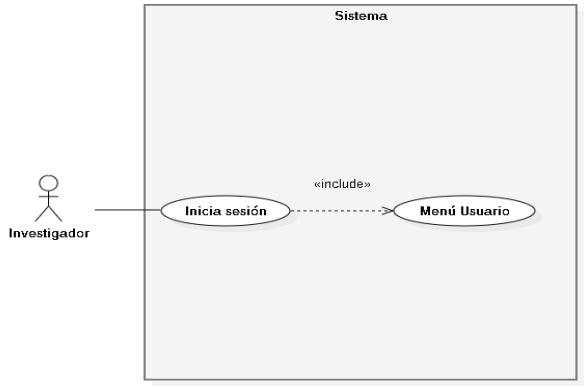
3.- Pulsa en el botón iniciar sesión.	4.-Comprueba autenticidad de las credenciales proporcionadas.
	5.-Redirecciona a la página inicial del investigador y carga menú correspondiente.
Flujo alterno:	4.-La comprobación de credenciales falla, el sistema muestra el detalle de lo sucedido.
Postcondiciones:	Se accede a las opciones investigador.
C.U relacionado:	
Formularios:	**Web** **Móvil**

Tabla 20. Caso de uso menú de usuario.

Caso de Uso:	Menú de usuario
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Mostrar las opciones con las que cuenta el usuario de tipo investigador.
Resumen:	Al iniciar sesión el investigador, visualiza las opciones de gestionar proyectos, observaciones y análisis.
Precondiciones:	Usuario autenticado como investigador.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra las opciones de gestionar proyectos, observaciones y análisis.
Flujo alterno:	1.-El sistema no muestra las opciones de gestión.
Postcondiciones:	El investigador selecciona entre proyectos, observaciones y análisis.
C.U relacionado:	
Formulario:	**Web**

	Móvil

Tabla 21. Caso de uso crear proyecto.

Caso de Uso:	Crear proyecto
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Crear proyecto de estudio y definir variables a utilizar.
Resumen:	El investigador crea un nuevo proyecto de estudio ingresando nombre del proyecto, el responsable y el estado.
Precondiciones:	El investigador debe ingresar a la opción Proyecto.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra la interfaz de todos los proyectos creados por el investigador y la opción de crear nuevo proyecto.
2.-Realiza un clic en crear nuevo proyecto.	3.-Muestra la pantalla para ingresar información del proyecto.
4.-Ingresa el nombre, responsable, estado (privado o público).	
5.-Pulsa en el botón crear proyecto.	6.-Muestra notificación de la creación correcta del proyecto y un enlace para seguir con la configuración de las variables a estudiar.

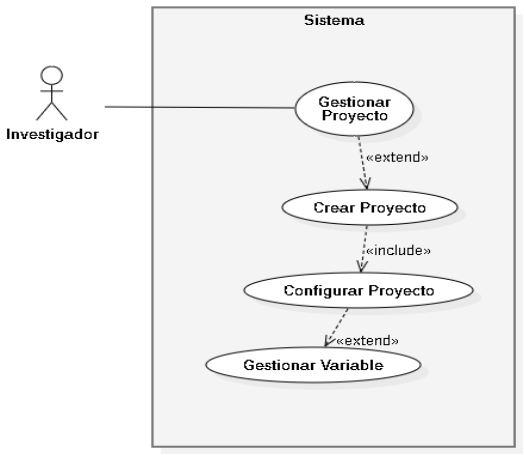
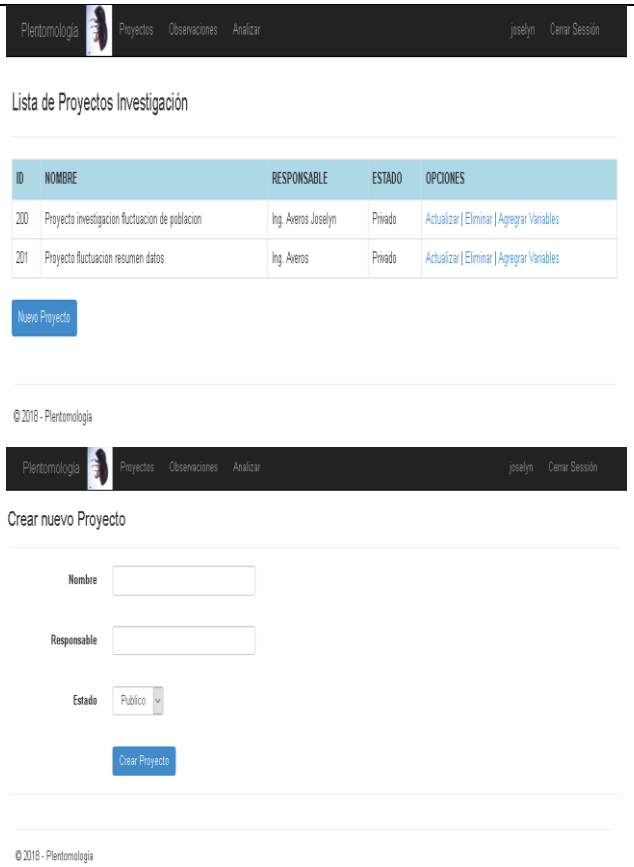
7.-Pulsa en el enlace.	8.-Muestra la pantalla para gestionar variable. Ir al caso de uso expandido crear variable.
Flujo alterno:	6.-El sistema notifica el detalle del problema por el cual no se pudo crear.
Postcondiciones:	Se crea el proyecto.
C.U relacionado:	
Formulario:	

Tabla 22. Caso de uso crear variable.

Caso de Uso:	Crear Variable
Tipo: Primaria	**Prioridad:** Alta

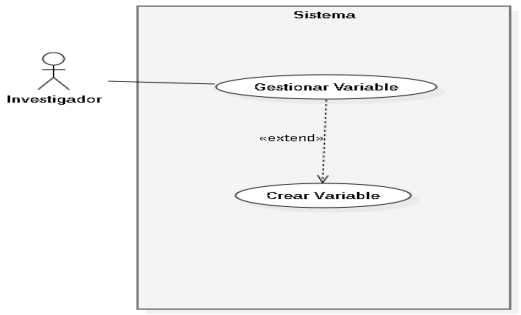
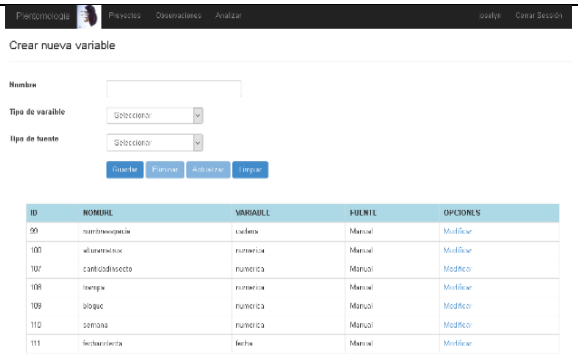
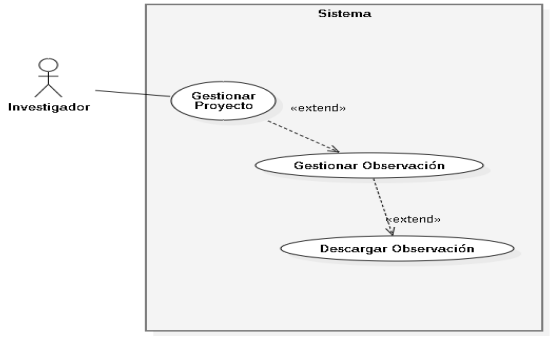
Actor:	Investigador
Propósito:	Crear y agregar variable a un proyecto de estudio.
Resumen:	El investigador crea y agrega la variable al proyecto de estudio previamente registrado.
Precondiciones:	El investigador debe crear un proyecto de estudio.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra la interfaz para la gestión de las variables del proyecto especificado y las opciones de crear, modificar, eliminar variables.
2.- Ingresa el nombre, el tipo de variable (decimal, cadena, numérica, fecha).	
3.- Pulsa el botón guardar.	4.-Registra la variable en el proyecto de estudio.
	5.-Muestra mensaje de creación de variable con éxito.
Flujo alterno:	4.-El sistema no registra la variable en el proyecto. 5.-Muestra mensaje del problema por el cual no se pudo crear la variable en el proyecto.
Postcondiciones:	Se crea la variable en el proyecto especificado.
C.U relacionado:	
Formulario:	

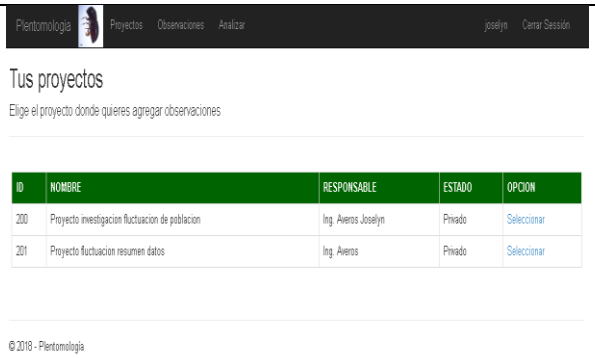
Tabla 23. Caso de uso descargar observaciones.

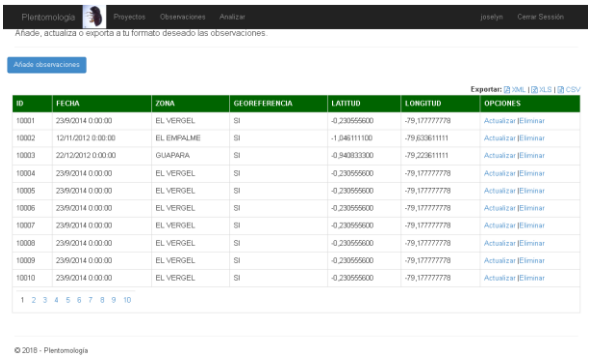
Caso de Uso:	Descargar Observaciones
Tipo: Primaria	**Prioridad:** Media
Actor:	Investigador
Propósito:	Descargar las observaciones del proyecto seleccionado.
Resumen:	Todas las observaciones que el investigador realiza estarán disponibles para la descarga, en formatos (XML, CSV, XLS).
Precondiciones:	Investigador debe registrar observaciones.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra lista de los proyectos del investigador y la opción de selección.
2.-Clic en el proyecto que desea descargar la observación.	
	3.-Muestra lista de las observaciones del proyecto seleccionado y las opciones de exportar (XML, CSV, XLS).
4.-Clic en la opción de exportar que desea.	
	5.-Genera el archivo en el formato seleccionado y muestra la opción de guardar.
6.-Clic en el botón guardar.	
	7.-Descarga el archivo en el ordenador del usuario.
Flujo alterno:	5.-El sistema no puede generar el archivo.
Postcondiciones:	Se generar archivo para la descarga.

C.U relacionado:



Formulario:





Archivo XML

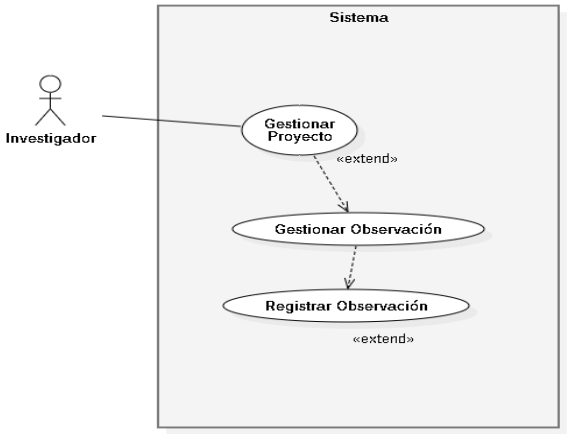

```
<DATA>
  <observacionId>10001</observacionId>
  <observacionFecha>2014-09-23</observacionFecha>
  <zona>EL VERGEL</zona>
  <georeferencia>1</georeferencia>
  <decimalLatitud>-0.230555600</decimalLatitud>
  <decimalLongitud>-79.177777778</decimalLongitud>
  <pais>ECUADOR</pais>
  <provincia>Santo Domingo de los Tsáchilas</provincia>
  <ciudad>Santo Domingo de los Tsáchilas</ciudad>
  <fechacolecta>2014-09-23</fechacolecta>
  <semana>1</semana>
  <bloque>1</bloque>
  <alturametros>1</alturametros>
  <trampa>1</trampa>
  <cantidadinsecto>0</cantidadinsecto>
  <nombreespecie>P. cavipemnis</nombreespecie>
</DATA>
<DATA>
  <observacionId>10004</observacionId>
  <observacionFecha>2014-09-23</observacionFecha>
  <zona>EL VERGEL</zona>
  <georeferencia>1</georeferencia>
  <decimalLatitud>-0.230555600</decimalLatitud>
  <decimalLongitud>-79.177777778</decimalLongitud>
  <pais>ECUADOR</pais>
  <provincia>Santo Domingo de los Tsáchilas</provincia>
  <ciudad>Santo Domingo de los Tsáchilas</ciudad>
  <fechacolecta>2014-09-23</fechacolecta>
  <semana>1</semana>
  <bloque>1</bloque>
  <alturametros>1</alturametros>
  <trampa>2</trampa>
  <cantidadinsecto>0</cantidadinsecto>
  <nombreespecie>P. cavipemnis</nombreespecie>
</DATA>
```

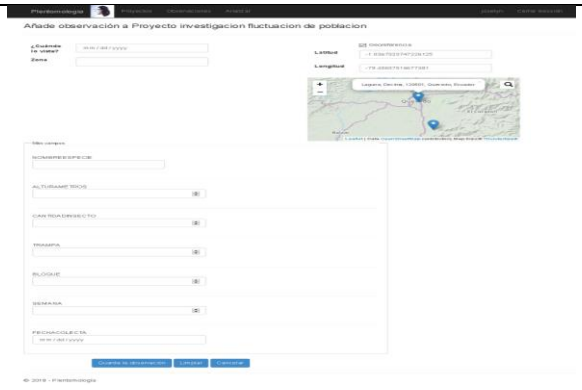

Archivo XLS

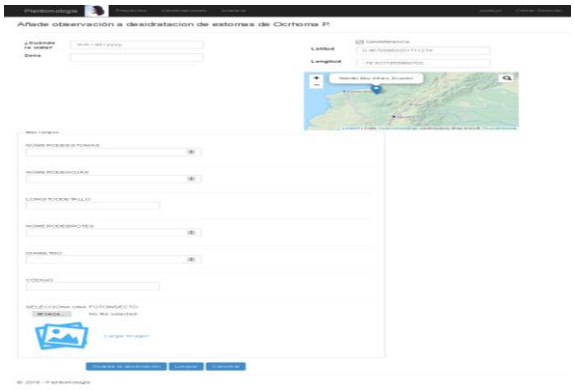
	CSV

Tabla 24. Caso de uso registrar observación.

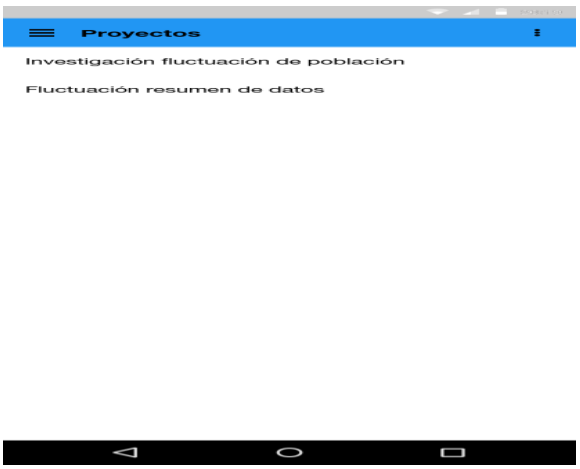
Caso de Uso:	Registrar observación
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Registrar observación a un proyecto.
Resumen:	Se registrará una observación al proyecto de estudio seleccionado, detallando cuando vio, en que zona, la referencia geográfica y las variables establecidas en el proyecto.
Precondiciones:	Debe crear un proyecto de estudio.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra lista de los proyectos del investigador y la opción de selección.
2.-Clic en el proyecto que desea ingresar observación.	

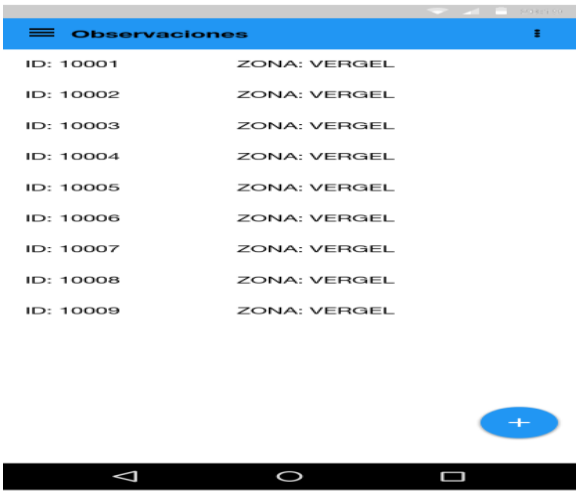
	3.-Muestra lista de las observaciones del proyecto seleccionado y enlace para añadir observación.
4.-Clic en el enlace de añadir observación.	
	5.-Muestra pantalla para ingresar información de la observación. Cuando fue visto, zona, georreferencia y los campos preestablecidos para el proyecto.
6.-Ingresa la fecha, la zona, georreferencia y los campos del proyecto.	
7.-Clic en el botón guardar observación.	
	8.-Registra la observación y muestra mensaje de registro ingresado con éxito.
Flujo alterno:	5.-El sistema muestra la pantalla para ingresar información, pero no permite ingresar la georreferencia. 8.-El sistema no registra la observación y muestra mensaje del detalle del problema.
Postcondiciones:	Se registra la observación.
C.U relacionado:	
Formulario:	**Web**


Añade observación a Proyecto investigacion fluctuacion de poblacion


Añade observación a desidratacion de estomas de Ocrhoma P

Movil


Proyectos
Investigación fluctuación de población
Fluctuación resumen de datos


Observaciones
ID: 10001 ZONA: VERGEL
ID: 10002 ZONA: VERGEL
ID: 10003 ZONA: VERGEL
ID: 10004 ZONA: VERGEL
ID: 10005 ZONA: VERGEL
ID: 10006 ZONA: VERGEL
ID: 10007 ZONA: VERGEL
ID: 10008 ZONA: VERGEL
ID: 10009 ZONA: VERGEL

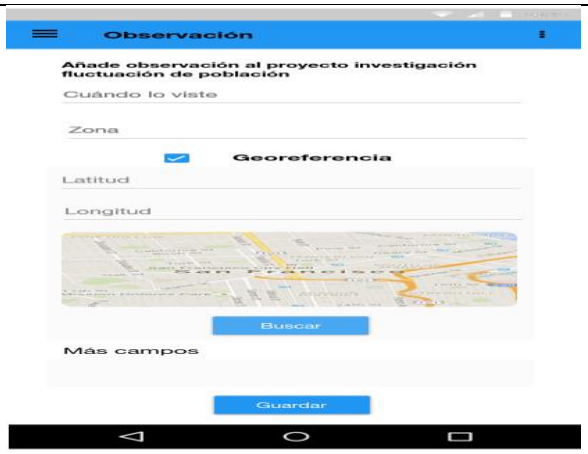
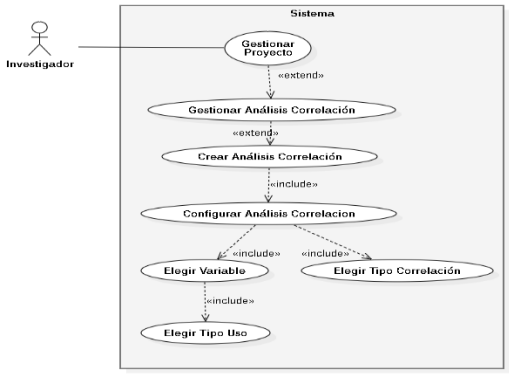
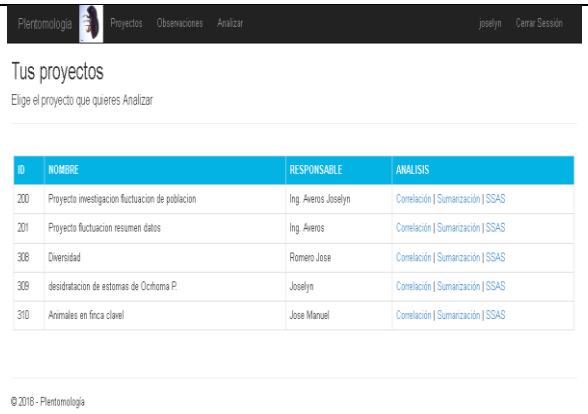
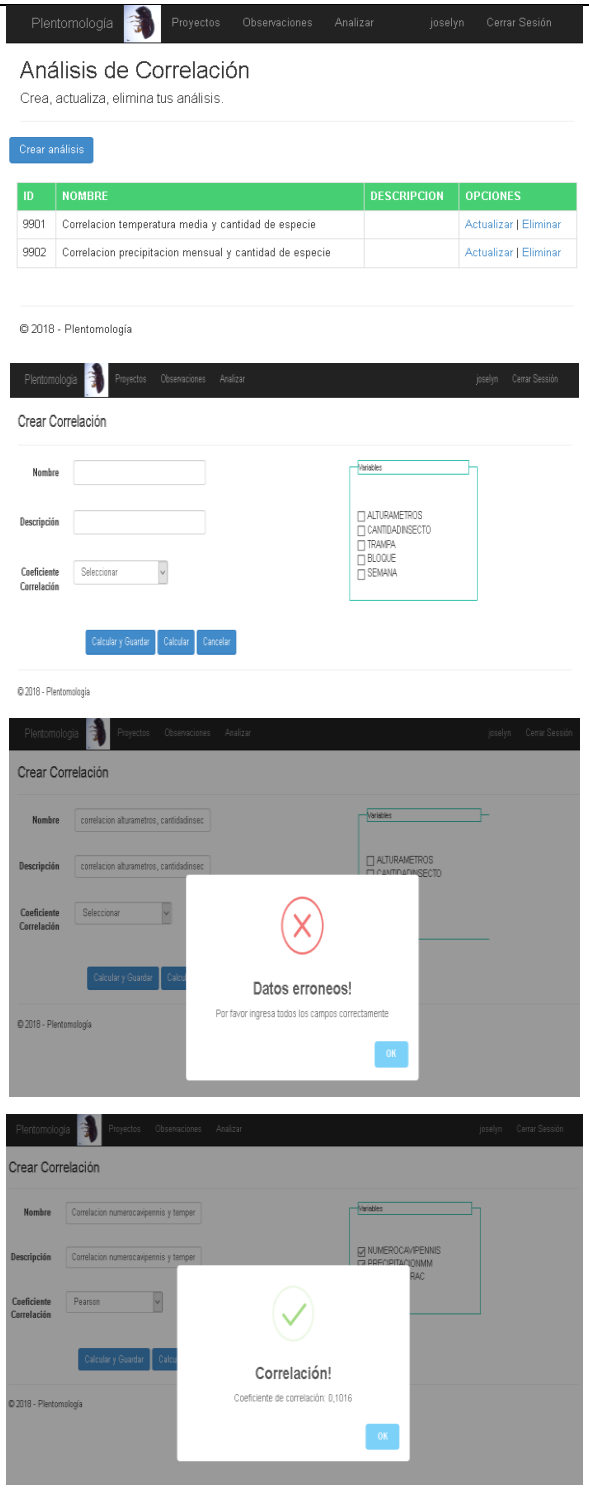
	
Sistema externo:	El sistema interactuar con la interfaz REST de Opencage para obtener la localización exacta y completar el registro.

Tabla 25. Caso de uso crear análisis de correlación.

Caso de Uso:	Crear análisis de correlación
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Crear análisis de correlación de un proyecto.
Resumen:	Se crea un análisis de correlación del proyecto de estudio seleccionado, se debe configurar las variables a utilizar.
Precondiciones:	1.-Investigador debe crear un proyecto de estudio. 2.-Investigador debe registrar observaciones en proyecto.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra lista de los proyectos del investigador y la opción de selección.
2.-Clic en el proyecto que desea crear correlación.	
	3.-Muestra lista de los análisis de correlación del proyecto y opción de crear nuevo análisis.

4.-Clic en el enlace de crear análisis.	
	5.-Muestra pantalla para configurar análisis de correlación. Nombre, descripción, el coeficiente de correlación (Pearson, Spearman) y los nombres de las variables del proyecto. Además, muestra opciones de guardar, calcular o cancelar.
6.-Ingresa el nombre, descripción, elige el coeficiente de correlación y selecciona las variables a utilizar en la correlación.	
7.-Clic en el botón guardar.	
	8.-Crea el análisis de correlación y muestra mensaje de registro ingresado con éxito. Ir al caso de uso ejecutar análisis de correlación
Flujo alterno:	8.-El sistema no registra el análisis y muestra el detalle del problema.
Postcondiciones:	El investigador registra análisis de correlación.
C.U relacionado:	
Formulario:	


Plentomología
Proyectos
Observaciones
Analizar
joselyn
Cerrar Sesión
Análisis de Correlación
Crea, actualiza, elimina tus análisis.
Crear análisis
ID
NOMBRE
DESCRIPCION
OPCIONES
9901
Correlacion temperatura media y cantidad de especie
Actualizar | Eliminar
9902
Correlacion precipitacion mensual y cantidad de especie
Actualizar | Eliminar
© 2018 - Plentomología
Crear Correlación
Nombre
Descripción
Coeficiente Correlación
Seleccionar
Variables
ALTURAMETROS
CANTIDADINSECTO
TRAMPA
BLOQUE
SEMANA
Calcular y Guardar
Calcular
Cancelar
Datos erroneos!
Por favor ingresa todos los campos correctamente
OK
Correlación!
Coeficiente de correlación: 0,1016
Pearson
NUMEROCAVIPENNIS

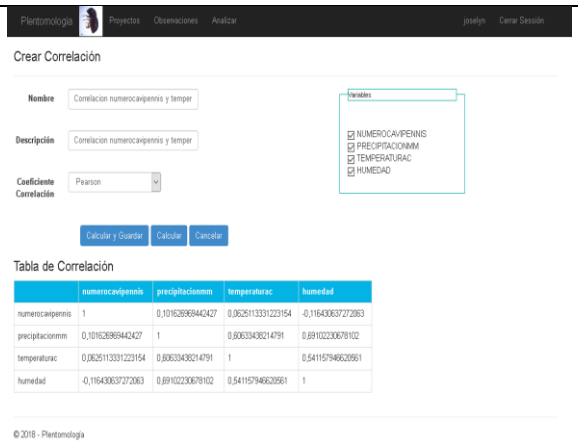
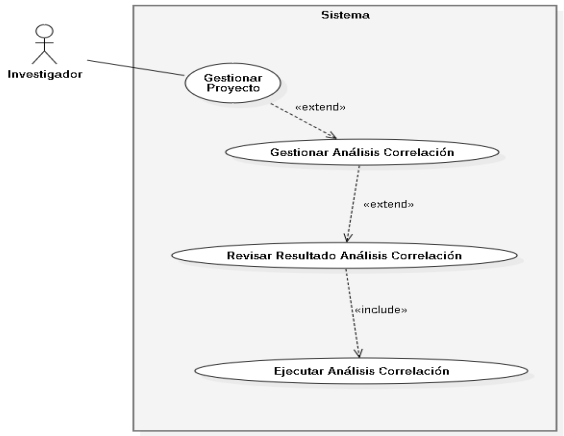
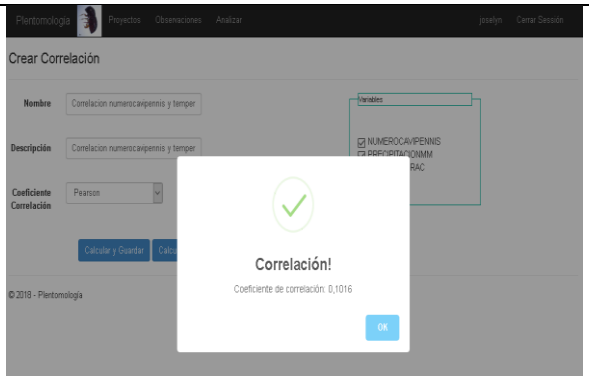
	
Validaciones:	El sistema muestra nombres de variable de tipo decimal y entera.

Tabla 26. Caso de uso ejecutar análisis correlación.

Caso de Uso:	Ejecutar análisis correlación
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Ejecuta análisis de correlación de un proyecto.
Resumen:	Se ejecuta un análisis de correlación del proyecto de estudio seleccionado.
Precondiciones:	1.-Investigador debe crear un proyecto de estudio. 2.-Investigador debe registrar observaciones en proyecto.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra lista de los proyectos del investigador y la opción de selección.
2.-Clic en el proyecto que desea crear correlación.	
	3.-Muestra lista de los análisis de correlación del proyecto y opción de crear nuevo análisis.
4.-Clic en el enlace de crear análisis.	
	5.-Muestra pantalla para configurar análisis de correlación. Nombre, descripción, el coeficiente de

	correlación (Pearson, Spearman) y los nombres de las variables del proyecto. Además, muestra opciones de guardar, calcular o cancelar.
6.-Ingresa el nombre, descripción, elige el coeficiente de correlación y selecciona las variables a utilizar en la correlación.	
7.-Clic en el botón calcular.	
	8.-Calcula la correlación utilizando la configuración establecida y muestra el cálculo en pantalla.
Flujo alterno:	8.-El sistema no puede realizar el cálculo y muestra el detalle del problema.
Postcondiciones:	Se calcula el análisis de correlación.
C.U relacionado:	
Formulario:	

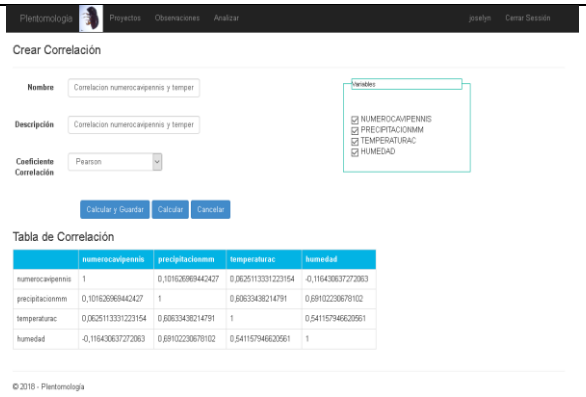
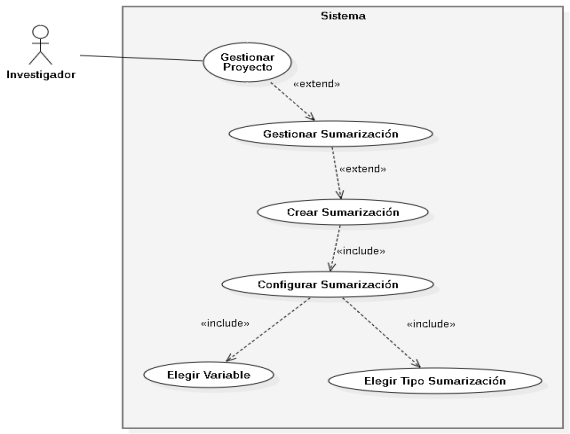
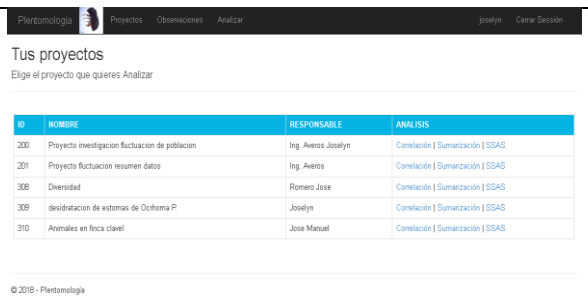
	
Procedimiento almacenado:	pcdobtenerdatosproyectoinv @idproyecto
Validaciones:	El sistema muestra nombres de variable de tipo decimal y entera.

Tabla 27. Crear análisis de sumarización.

Caso de Uso:	Crear análisis de sumarizacón
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Crear sumarización de un proyecto.
Resumen:	Se crea un análisis de sumarización del proyecto de estudio seleccionado, se debe configurar las variables a utilizar.
Precondiciones:	1.-Investigador debe crear un proyecto de estudio. 2.-Investigador debe registrar observaciones en proyecto.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra lista de los proyectos del investigador y la opción de selección.
2.-Clic en el proyecto que desea crear sumarización.	
	3.-Muestra lista de los análisis de sumarización del proyecto y opción de crear nuevo análisis.
4.-Clic en el enlace de crear análisis.	

	5.-Muestra pantalla para configurar análisis de sumarización. Nombre, descripción, el estado (procesado, no procesado), el tipo de sumarización (suma, promedio, varianza, mínimo máximo, varianza estadística, desviación estadística estándar, desviación típica estándar) y los nombres de las variables del proyecto con sus respectivas opciones de uso (Agrupación, valor totalizar). Además, muestra opciones de guardar, calcular o cancelar.
6.-Ingresa el nombre, descripción, elige el estado y selecciona el tipo de uso que se dará a cada variable.	
7.-Clic en el botón guardar.	
	8.-Crea el análisis de sumarización y muestra mensaje de registro ingresado con éxito. Ir al caso de uso ejecutar análisis de sumarización
Flujo alterno:	8.-El sistema no registra el análisis y muestra el detalle del problema.
Postcondiciones:	El investigador registra análisis de sumarización.
C.U relacionado:	
Formulario:	

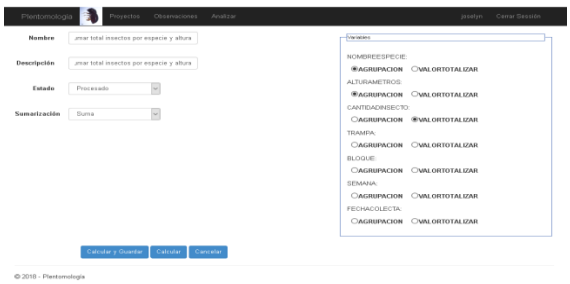
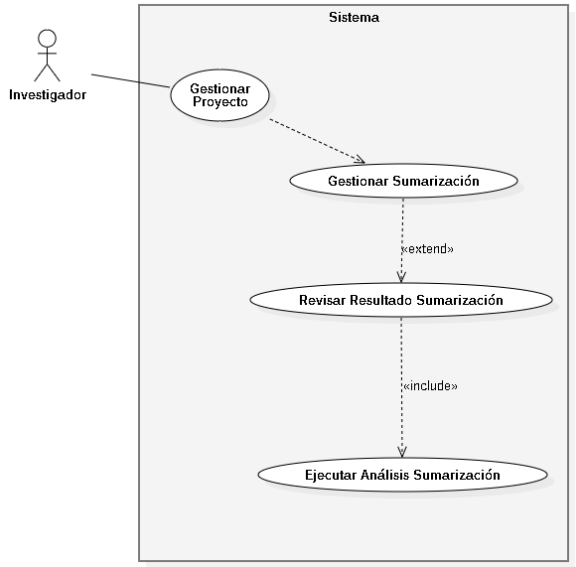
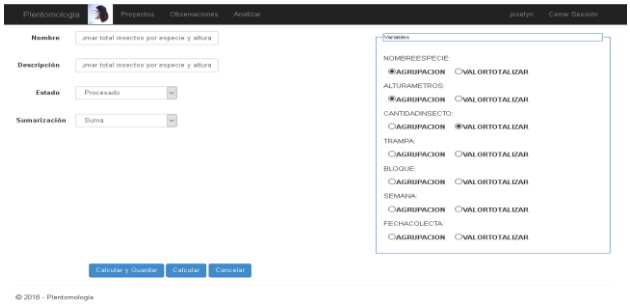
	
Validaciones:	El sistema no muestra nombres de variable de tipo foto.

Tabla 28. Caso de uso ejecutar análisis sumarización.

Caso de Uso:	Ejecutar análisis sumarización
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Ejecuta análisis de sumarización de un proyecto.
Resumen:	Se ejecuta un análisis de sumarización del proyecto de estudio seleccionado.
Precondiciones:	1.-Investigador debe crear un proyecto de estudio. 2.-Investigador debe registrar observaciones en proyecto.
Flujo normal:	
Actor	**Sistema**
	1.-Muestra lista de los proyectos del investigador y la opción de selección.
2.-Clic en el proyecto que desea crear sumarización.	
	3.-Muestra lista de los análisis de sumarización del proyecto y opción de crear nuevo análisis.
4.-Clic en el enlace de crear análisis.	
	5.- Muestra pantalla para configurar análisis de sumarización. Nombre, descripción, el estado (procesado, no procesado), el tipo de sumarización (suma, promedio, varianza, mínimo máximo,

	varianza estadística, desviación estadística estándar, desviación típica estándar) y los nombres de las variables del proyecto con sus respectivas opciones de uso (Agrupación, valor totalizar). Además, muestra opciones de guardar, calcular o cancelar.
6.-Ingresa el nombre, descripción, elige el estado y selecciona el tipo de uso que se dará a cada variable.	
7.-Clic en el botón calcular.	
	8.-Calcula la sumarización utilizando la configuración establecida.
	9.-Genera el archivo xls con los datos de resultante del cálculo y muestra la opción de guardar.
10.-Clic en el botón guardar.	
	11.-Descarga el archivo en el ordenador del usuario.
Flujo alterno:	8.-El sistema no puede realizar el cálculo y muestra el detalle del problema.
Postcondiciones:	Se calcula el análisis de sumarización.
C.U relacionado:	

<table>
<tr><td>Formulario:</td><td>


Archivos xls

nombreespecie	alturametros	TOTAL
P. cavipemnis	1	6
P. cavipemnis	4	0
P. cavipemnis	8	0

alturametros	fechacolecta	TOTAL
1	23/9/2014 0:00	0
4	23/9/2014 0:00	0
8	23/9/2014 0:00	0
1	1/10/2014 0:00	6
4	1/10/2014 0:00	0
8	1/10/2014 0:00	0
8	23/10/2014 0:00	0
</td></tr>
<tr><td>Procedimiento almacenado:</td><td>pcdcalcularsumarizacion
@idproyecto,
@cantidadVariables,
@variablesagrupacion,
@variabletotalizar,
@tiposumarizacion</td></tr>
<tr><td>Validaciones:</td><td>El sistema verifica que el investigador configure el uso de las variables.</td></tr>
</table>

Tabla 29. Caso de uso crear modelo de minería de datos.

Caso de Uso:	Crear modelo de minería datos.
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Crear modelo de minería datos.
Resumen:	Se crea un modelo de minería del proyecto de estudio seleccionado, se debe configurar las variables a utilizar.
Precondiciones:	1.-Investigador debe crear un proyecto de estudio. 2.-Investigador debe registrar observaciones en proyecto.

Flujo normal:	
Actor	**Sistema**
	1.-Muestra lista de los proyectos del investigador y la opción de selección.
2.-Clic en el proyecto que desea crear modelo de minería.	
	3.-Muestra lista de los análisis de minería del proyecto y opción de crear nuevo análisis.
4.-Clic en el enlace de crear análisis.	
	5.-Muestra pantalla para configurar el modelo de minería. Nombre, descripción, el estado (procesado, no procesado), porcentaje de datos de prueba, el número máximo de casos en el conjunto de datos de prueba, técnica (Regresión lineal, arboles de decision, naive bayes, clústeres) y los nombres de las variables del proyecto con sus respectivas configuraciones: Tipo de contenido (discretizado, discreta, cíclico, ordenado, continuo) Definición de cómo se usará en el modelo (entrada, predecir)
6.-Ingresa el nombre, descripción, elige el estado, selecciona la técnica y estable la configuración de las variables.	
7.-Clic en el botón crear y guardar.	
	8.-Crea el análisis de minería y muestra mensaje de registro ingresado con éxito. Ir al caso de uso ejecutar análisis minería.
Flujo alterno:	8.-El sistema no registra el análisis y muestra el detalle del problema.
Postcondiciones:	Se crea el modelo de minería.

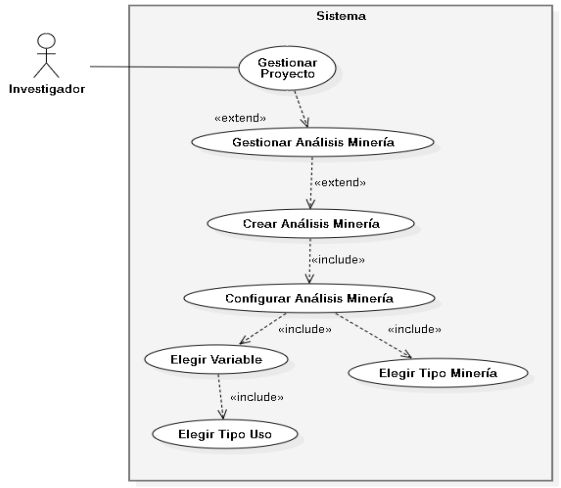
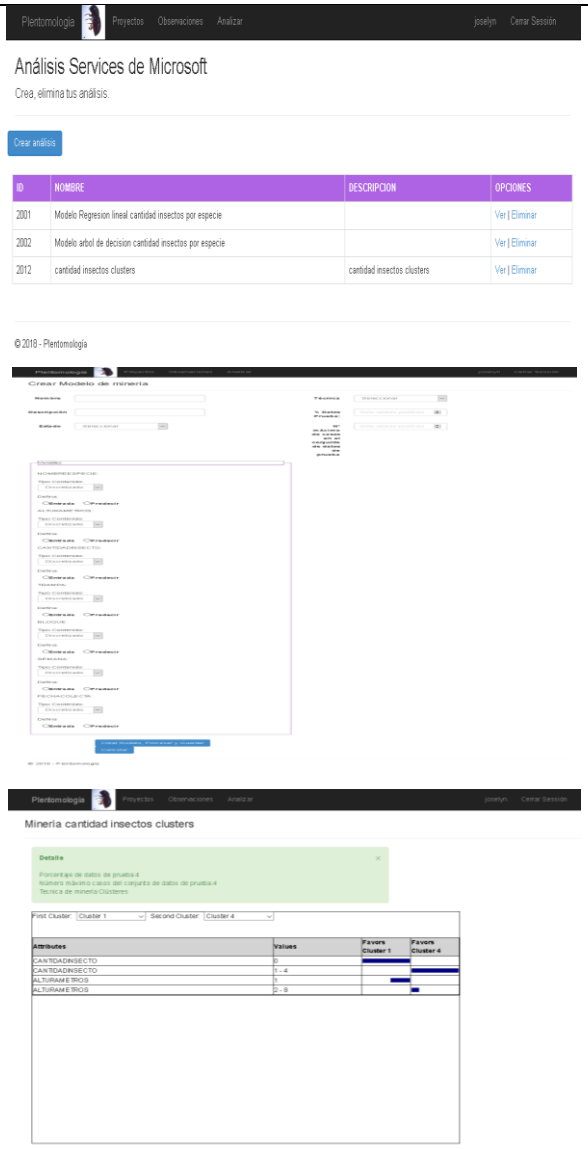
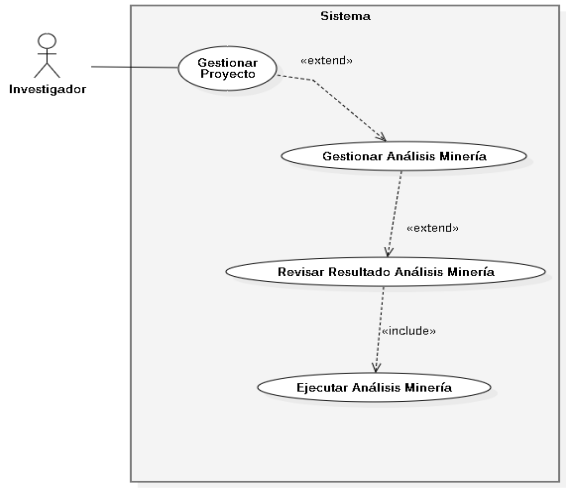
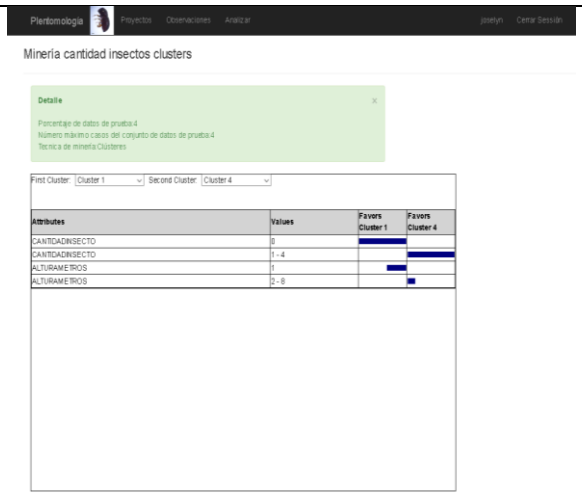
C.U relacionado:	
Formulario:	
Validaciones:	El sistema verifica que las variables, tengan la configuracion correcta.

Tabla 30. Caso de uso ejecutar análisis de minería.

Caso de Uso:	Ejecutar análisis minería
Tipo: Primario	**Prioridad:** Alta
Actor:	Investigador
Propósito:	Ejecuta análisis minería de un proyecto.
Resumen:	Se ejecuta análisis de minería del proyecto de estudio seleccionado.
Precondiciones:	1.-Investigador debe crear un proyecto de estudio. 2.-Investigador debe registrar observaciones en proyecto. 3.-Investigador debe crear un análisis de minería.
Flujo normal:	
Actor	**Sistema**
	1.- Muestra pantalla de detalle con la configuración del proyecto de minería y la opción de ejecutar.
2.-Clic en el botón ejecutar.	
	3.-Calcula la minería utilizando la configuración establecida.
	4.-Muestra el componente con datos del resultante del cálculo.
Flujo alterno:	4.-El sistema no puede realizar el cálculo y muestra el detalle del problema.
Postcondiciones:	Se calcula el análisis de minería.
C.U relacionado:	

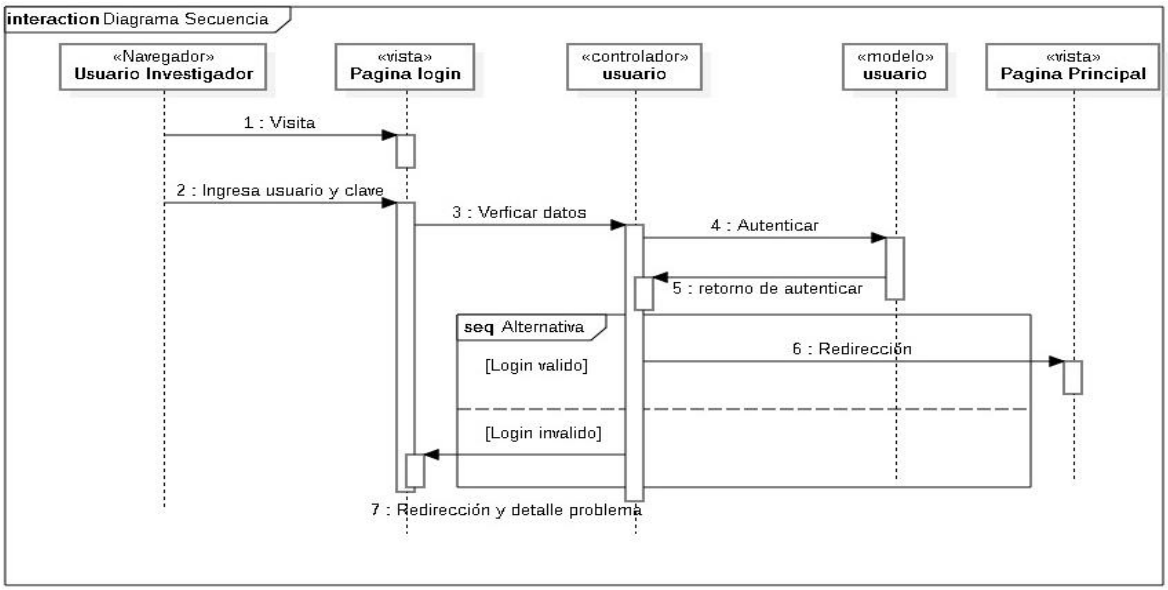
Formulario:	 Plentomologia Proyectos Observaciones Analizar joselyn Cerrar Sesión Minería cantidad insectos clusters Detalle Porcentaje de datos de prueba:4 Número máximo casos del conjunto de datos de prueba:4 Tecnica de minería:Clústeres First Cluster: Cluster 1 Second Cluster: Cluster 4 Attributes Values Favors Cluster 1 Favors Cluster 4 CANTIDADINSECTO 0 CANTIDADINSECTO 1 - 4 ALTURAMETROS 1 ALTURAMETROS 2 - 8

4.1.2.4. Diagrama de secuencias

Presenta una descripción de bajo nivel, como los objetos funcionan en conjunto. Se muestran los diagramas de mayor importancia para el sistema.

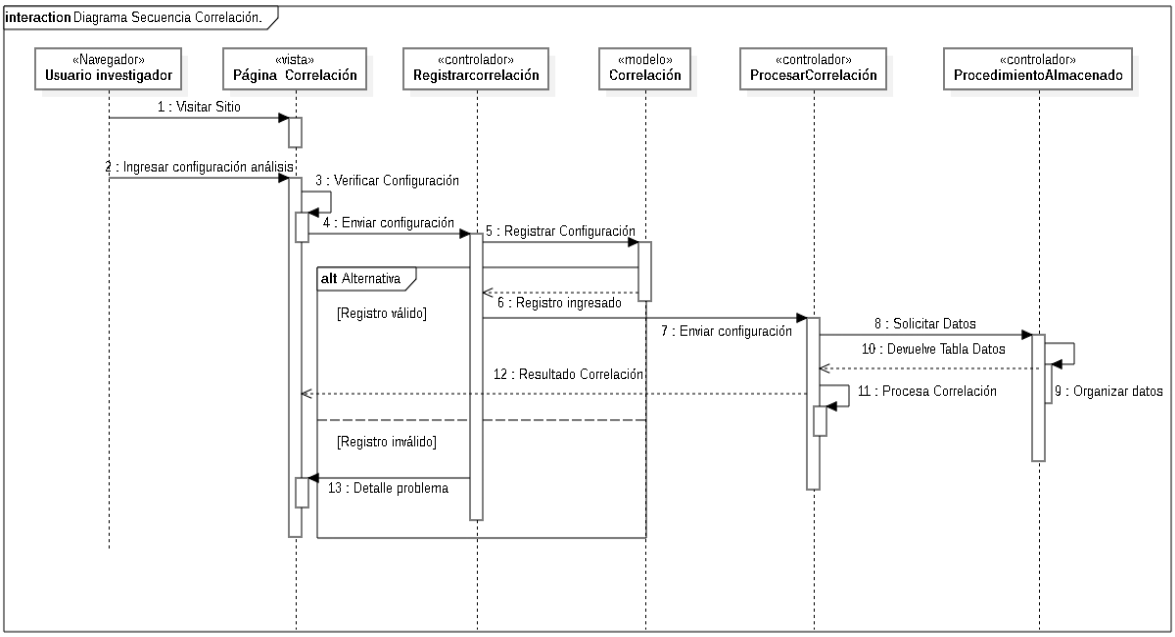
Inicio sesión

Ilustración 32. Diagrama de secuencia inicio sesión.



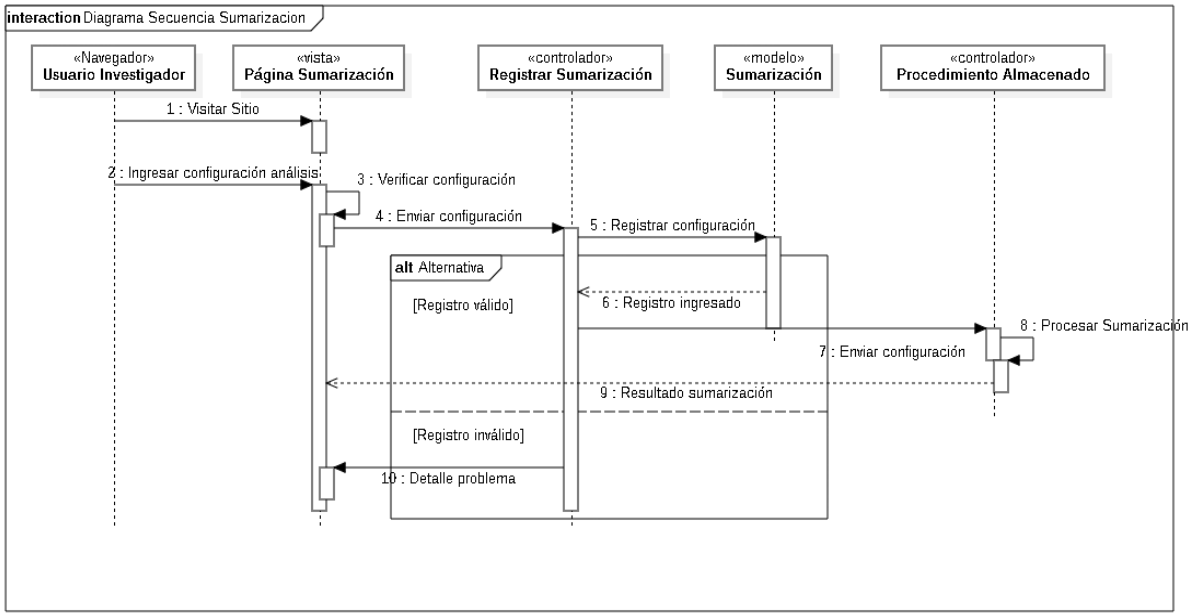
Crear análisis de correlación

Ilustración 33. Diagrama de secuencia crear análisis de correlación.



Crear análisis de sumarización

Ilustración 34. Diagrama de secuencia crear análisis de sumarización.



4.1.2.5. Arquitectura del sistema

La aplicación utiliza la arquitectura C/S. Donde el cliente web o móvil solicita recursos al servidor de aplicación el cual responde con la respuesta. Adicionalmente para lograr la comunicación con diferentes clientes se utiliza **REST** junto a JSON para tener una alternativa ligera y eficiente con los recursos de la web debido al uso que hace de HTTP.

4.1.2.6. Patrón de diseño

El patrón diseño MVC se utilizó por su idea principal de realizar una división entre objetos de dominio que modelan la forma en que se perciben los objetos del mundo real y objetos de presentación que son los elementos interfaz de usuarios que vemos en la pantalla [1].

En la aplicación web se aplicó de la siguiente manera. La vista muestra la información del modelo al usuario mediante la interfaz y se encarga de llamar al controlador correspondiente dependiendo de la acción que el usuario elija. El controlador recibe la llamada (evento clic) de la vista, y se encarga de gestionar los objetos modelo o vista dependiendo del caso, además de esto se encarga de implementar la lógica del negocio. El modelo se encarga de la

capa de datos, representando la estructura del dato. A demás de esto, se utilizó el patrón DAO porque provee una interfaz entre el modelo y la base de datos (ver Ilustración 36). Por otro lado, la aplicación móvil utilizará el patrón de MVP por su arquitectura basada en componentes que se enfoca en la descomposición del diseño en componentes funcionales o lógicos que expongan interfaces de comunicación bien definidas (ver Ilustración 35) [2].

Ilustración 35. Patrón de diseño de la aplicación móvil PIentomología.

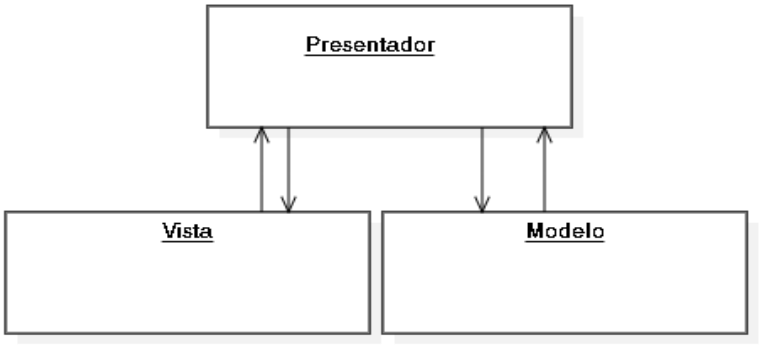
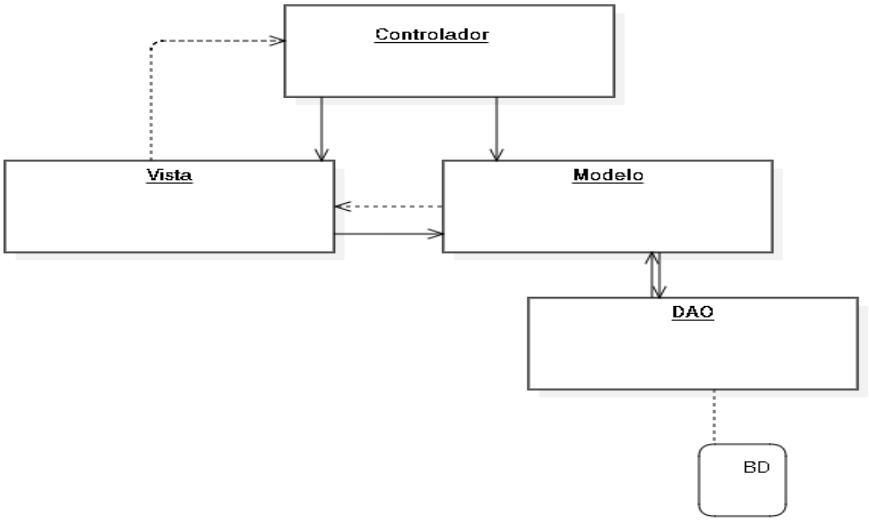


Ilustración 36. Patrón de diseño de la aplicación web PIentomología.



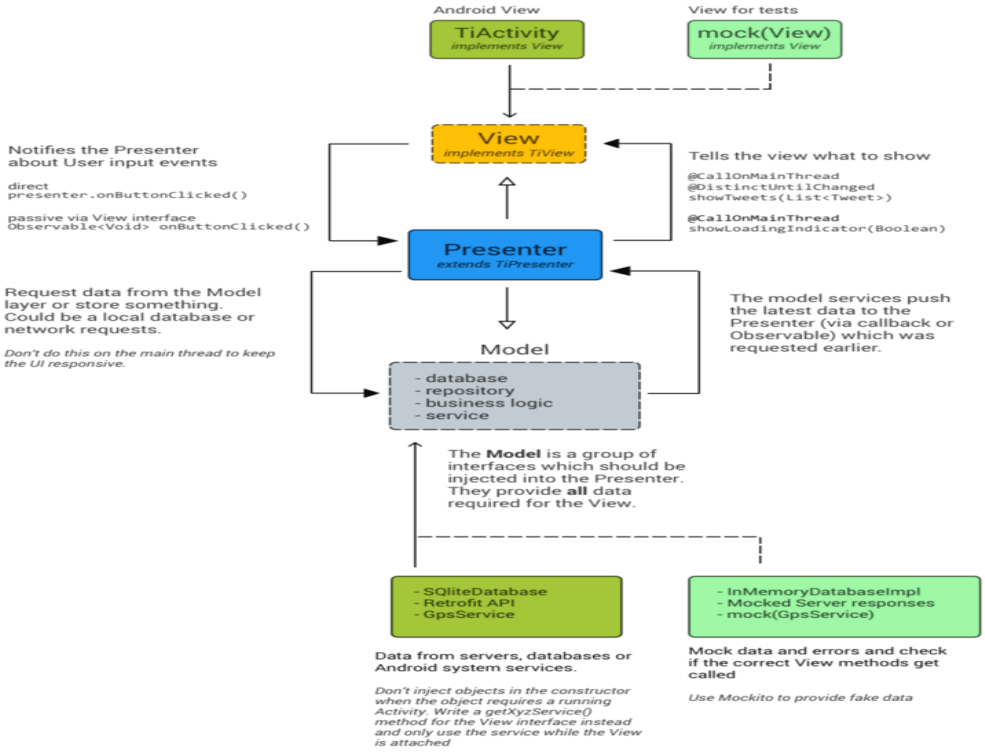
En la aplicación móvil se aplicó MVP mediante la utilización de la librería ThirtyInch, la descripción general de cómo implementa la arquitectura MVP con ThirtyInch (ver Ilustración 37). Detalles de métodos que se deben implementar:

- TiPresenter: objeto que vive durante toda la vida útil de una actividad, incluso cuando la actividad se recrea, se traslada al fondo y se elimina. Tiene cuatro eventos de ciclo de vida:

- onCreate: se llama una vez para la inicialización, la vista no está adjunta actualmente.
- onAttachView: la vista se adjuntó y es visible para el usuario.
- onDetachView: la vista se desconectará después de esta llamada y ya no será visible para el usuario.
- onDestroy: se llama una vez cuando la Actividad / Fragmento se destruye completamente y nunca volverá a aparecer. ¡Detén todo tu trabajo!

- TiView: las anotaciones de interfaz para los métodos:
  - CallOnMainThread: se asegura de que el método de vista se llame automáticamente en el hilo principal.
  - DistinctUntilChanged: no llama al método de vista dos veces cuando ya se llama con el mismo parámetro.
  - Las anotaciones funcionan mediante la presentación de la vista.

Ilustración 37. Arquitectura MVP con ThirtyInch.

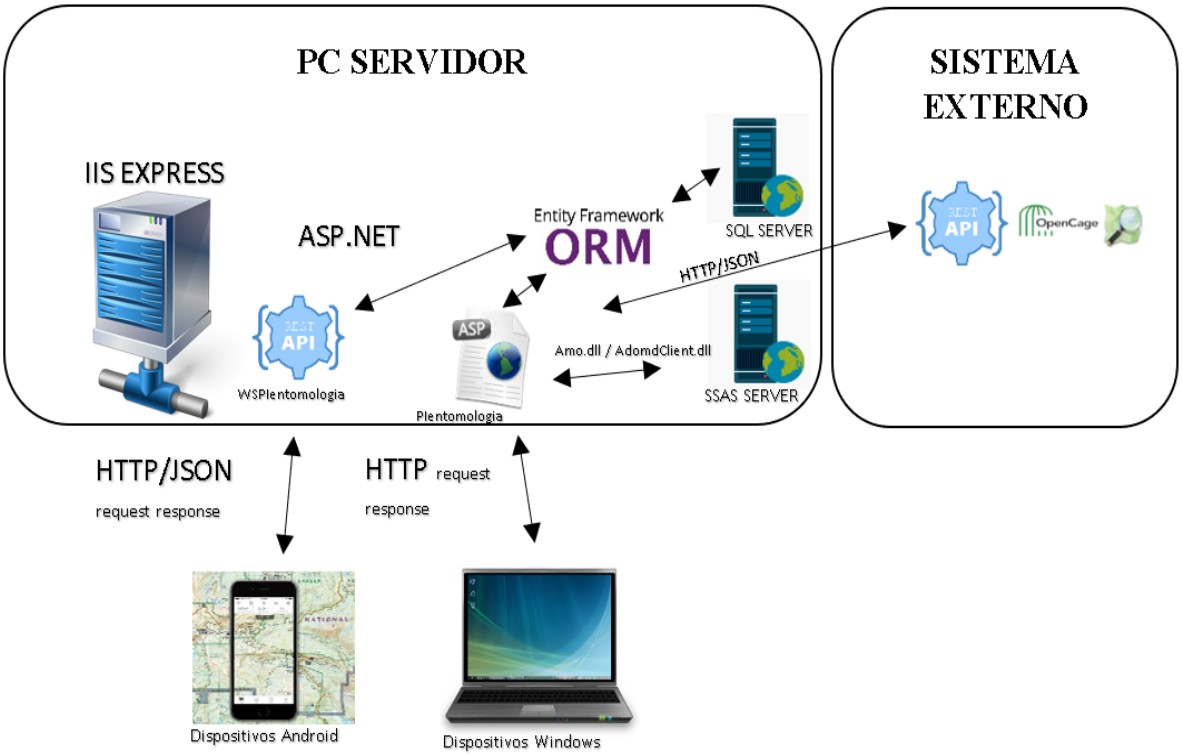


FUENTE: THIRTYINCH —A NEW MVP LIBRARY FOR ANDROID

ELABORADO POR: PASCAL WELSCH

4.1.2.7. Arquitectura prototipo del hardware sistema

Ilustración 38. Arquitectura prototipo del hardware sistema

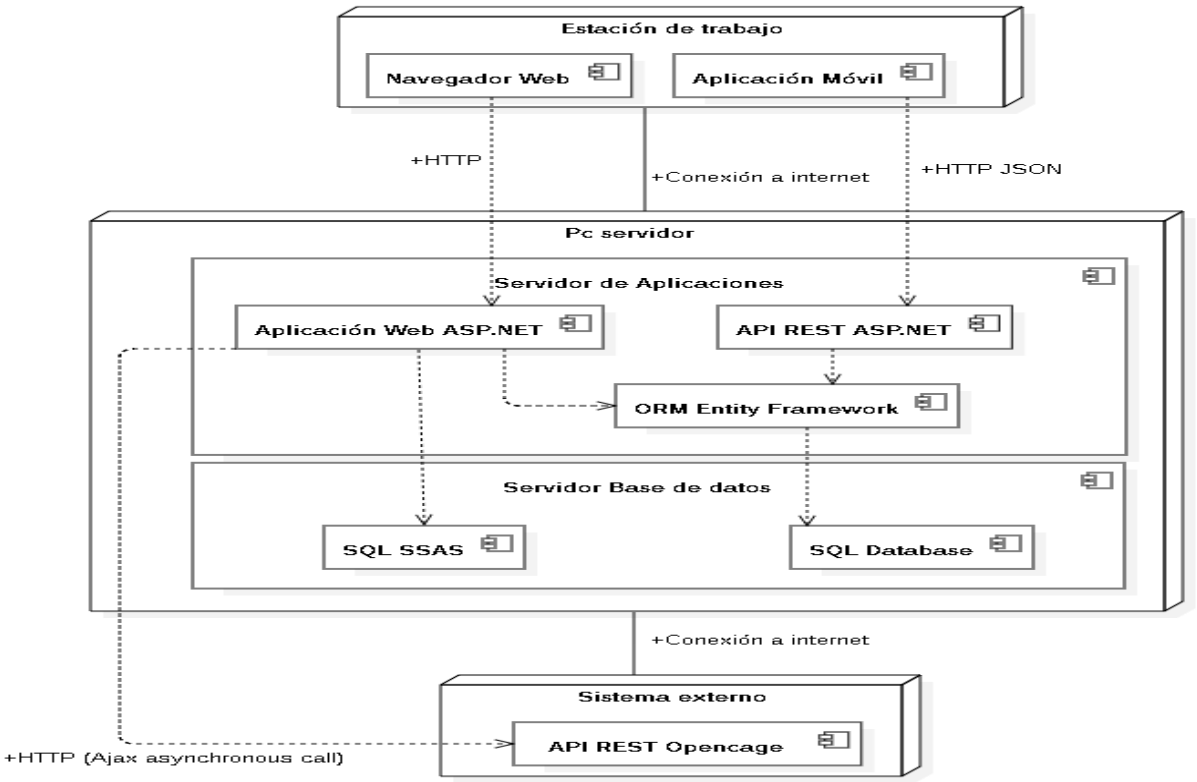


4.1.3. Fase construcción

En esta fase se llegó a obtener la capacidad operacional inicial (prototipo). Mediante el desarrollo de las características de la aplicación, implementación de los procesos e integración de todos los componentes.

4.1.3.1. Diagrama de despliegue

Ilustración 39. Diagrama de despliegue PIentomología.



4.1.3.2. Entorno tecnológico

A continuación, se explica brevemente cada una de las tecnologías utilizadas. Para el control de versiones se utilizó bitbucket, el cual ofrece un servicio de control de versiones Git basado en la web, es gratuito y permite tener varios repositorios privados. Permitió guardar las diversas versiones del proyecto PIentomología y de la Api PIentomología, se puede evidenciar en el **anexo 3**.

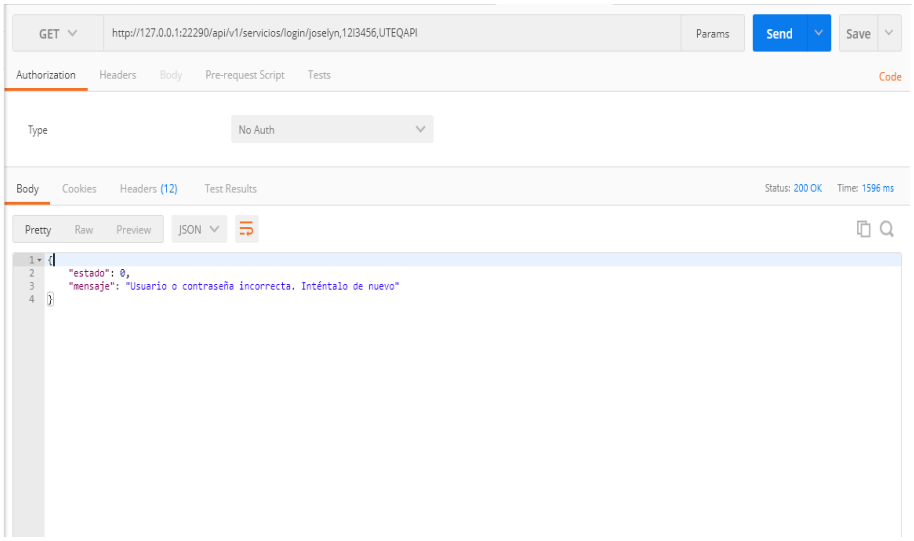
Microsoft SQL Server 13.0.1 es un sistema de bases de datos del modelo relacional:

- Soporta TSQL medio de interacción con el Servidor.
- Permite crear scripts de comandos TSQL llamados procedimientos almacenados.
- Arquitectura cliente-servidor.
- Soporta tipos de datos numéricos, textos, fechas, XML, datos binarios.
- Permite creación de tablas temporales y índices.

Marco tecnológico para la construcción de la aplicación REST:

- ASP.NET web API 5.2.6 para crear y exponer los servicios HTTP.
- Entity framework 6.1.3 para mapear y realizar operaciones de CRUD.
- Newtonsoft 6.0.4 para la serialización y deserialización de objetos.

Ilustración 40. Solicitud Get login en aplicación Postman.



Marco tecnológico para la construcción de la aplicación web:

- PIentomología usa para la presentación páginas web forms, las mismas que primero se ejecuta en el servidor, a su vez el framework es el que genera el marcado HTML5 con los respectivos CSS y JavaScript. En el nivel de aplicación el entorno de programación fue ASP.Net C#. A nivel acceso a datos, la aplicación implementa un ORM llamado Entity framework 6.1.3 el cual permite interactuar con datos almacenadas en la base de datos SQL Server mediante los objetos POCO utilizando las consultas LINQ. Se lo puede evidenciar el código fuente en el **anexo 5**.
- Mantnet.Numerics 4.5.1 proporciona módulo estadístico para cálculos de coeficiente de correlación.
- OpenCage Geocoder 1.0.1 proporciona geo codificación inversa, el acceso a los recursos de georreferencia se obtiene mediante un clave de api generada por la página oficial.
- Map thunderforest librería JavaScript de mapas interactivos, del mismo modo para el acceso a los recursos de map se tiene que solicitar un clave de api.

Se indica el marco tecnológico para la construcción de la aplicación móvil:

- Android proporciona dos lenguajes para desarrollar código fuente, Java y kotlin. El primero es antiguo con una gran cantidad de documentación y comunidades, en cuanto a Kotlin fue difundido como lenguaje oficial para Android en 2017, unos de los puntos a favor para ambos es que son totalmente interoperables, lo cual se define como la capacidad que tiene diferentes aplicaciones a conectarse y comunicarse de manera coordinada, sin esfuerzo. En el caso de Java y Kotlin, las partes del código fuente podrían estar escritas en Java y en Kotlin. En la API PIentomología se utiliza Java.
- Retrofit 2.5.0 como cliente HTTP, simplifica la comunicación HTTP al convertir la API remota en interfaz declarativa y de tipo seguro. Además, permite consumir datos JSON o XML de forma sencilla. La forma de funcionamiento de Retrofit es la siguiente al utilizar una de las interfaces definidas previamente, el mismo se encarga de realizar la llamada a la API, al terminar el llamado convierte el resultado al modelo de datos que previamente se especificó. La conversión es de texto JSON a POJO.
- OkHttp 3.14.1 es un cliente HTTP que es eficiente porque entre las razones más destacables tiene: compatibilidad con HTTP/2 permite que todas las solicitudes al mismo host compartan un socket y GZIP transparente reduce el tamaño de descarga.
- ListView muestra una colección de vista (ítem) de desplazamiento vertical donde cada vista se coloca inmediatamente debajo de la vista anterior en la lista. Donde la colección de vista será el objeto de dominio que se desea presentar al usuario.
- RxJava 2.0.0 sirve para manejar procesos bajo hilo secundario, los consumos de API es un proceso que puede llevar una cantidad de tiempo elevado, durante ese tiempo no es conveniente dejar al usuario con la aplicación congelada, por lo tanto, es preferible manejar un hilo secundario (llamado en segundo plano). La librería usa secuencias de observables para manejar eventos asíncronos. Por otro lado, extiende el patrón de observable para admitir secuencias de datos/ eventos y agregar operadores que le permitan componer secuencias de manera declarativa al tiempo que abstrae preocupaciones sobre cosas como subprocesos de bajo nivel, sincronización, seguridad de subprocesos y estructuras de datos concurrentes.
- ThirtyInch 0.9.0 esta biblioteca agrega presentadores a actividades y fragmentos (elementos ligados a interfaz usuario). Favorece el patrón de presentador con estado,

donde el presentador sobrevive a los cambios de configuración y el patrón de vista tonto, donde la vista solo envía eventos de usuario y recibe información del presentador, pero nunca solicita datos de manera activa. Esto hace que las pruebas sean muy fáciles porque ninguna lógica vive en la Vista (Actividad, Fragmento).

Los siguientes puntos tienen como objetivo explicar los procesos llevados a cabo para realizar los diferentes análisis, algunos se describen en detalle, mientras que otros solo se describen brevemente:

4.1.3.3. Análisis Sumarización

La aplicación web se encarga de mostrar la interfaz de configuración. Código para la crear de variables dinámicas y mostrar en pantalla.

Ilustración 41. Creación de variables dinámicas y mostrar en pantalla.

```
1 reference | kevin, 32 days ago | 1 author, 2 changes
private Label crearLabel(string id, string etiqueta) {
    Label lblNuevo = new Label();
    lblNuevo.ID = id;
    lblNuevo.Text = etiqueta.ToUpper();
    lblNuevo.CssClass = "control-label";
    return lblNuevo;
}
1 reference | kevin, 32 days ago | 1 author, 3 changes
private RadioButtonList crearRadioButtonList(string id
                                            , List<Models.tbtipousovariable> listaACargar
                                            , string dataTextField, string dataValuedField){
    RadioButtonList radioButtonList = new RadioButtonList();
    radioButtonList.ID = id;
    radioButtonList.RepeatDirection = RepeatDirection.Horizontal;
    radioButtonList.CellPadding = 10;
    radioButtonList.DataSource = listaACargar;
    radioButtonList.DataTextField = dataTextField;
    radioButtonList.DataValueField = dataValuedField;
    radioButtonList.DataBind();
    radioButtonList.ClearSelection();
    return radioButtonList;
}
0 references | kevin, 32 days ago | 1 author, 5 changes
protected void Page_Load(object sender, EventArgs e){
    if (!Page.IsPostBack){//CargarDropdowlist
        configurarDropDownList("ddltipoSumarizacion",
                               new Controller.TipoSumarizacionController()
                               .listaTiposSumarizacion(),"nombretiposumarizacion","idtiposumarizacion");}
    //Obtener las variables a presentar
    int idProyecto = obtenerIdProyectoDSession();
    if (!idProyecto.Equals(-1)){
        Models.tbproyectoinv proyecto = new Controller.ProyectoInvController()
                                            .buscarProyectoXid(idProyecto);

        if (proyecto != null){
            List<Models.tbtipousovariable> tiposUsosVariables = new Controller.TipoUsoVariableController()
                                                                    .listaTiposUsosVariable();
            foreach (Models.tbvariable itemVariable in proyecto.tbvariable) {
                if (esSumarizable(itemVariable.tbtipovariable.tipo)) {
                    //Agregar las variables al panelControl
                    panelCampos.Controls.Add(crearLabel(
                                             LABEL + itemVariable.idvariable.ToString()
                                              , itemVariable.nombrevariable + ":   "));
                    if (tiposUsosVariables != null && tiposUsosVariables.Any()) {
                        panelCampos.Controls.Add(crearRadioButtonList(RADIOBUTTONLIST + itemVariable.idvariable.ToString()
                                                                      , tiposUsosVariables, "uso", "idtipousovariable"));}
                }
            }
        }
    }
}
```

Se deben establecer parámetros como el tipo sumarización las opciones soportadas son suma, promedio, varianza, mínimo máximo, varianza estadística, desviación estadística estándar, desviación típica estándar y establecer el uso que se le dará a cada variable puede ser de

agrupación o de valor totalizar. Al terminar de configurar, se solicitará el cálculo. El resultado será un archivo formato xls con los datos del cálculo. Procesos llevados a cabo para calcular la sumarización se detallan a continuación. La configuración del análisis se envía a un script de base de datos llamado pcdcalcularsumarizacion. El mismo que se encarga de organizar los datos para poder procesarlos, al culminar la organización se ejecuta el análisis. Código fuente de script pcdcalcularsumarizacion.

Ilustración 42. Script para calcular la sumarización.

```
ALTER PROCEDURE [dbo].[pcdcalcularsumarizacion]
@idproyecto INT,
@cantidadvariable INT,
@idVariable VARCHAR(MAX),
@idVariableReferente VARCHAR(MAX),
@tipoSumarizacion VARCHAR(MAX)
AS
BEGIN
/*PARAMETROS*/
DECLARE @scriptcompleto VARCHAR(MAX)
SET @scriptcompleto = ''
SET @cantidadvariable = @cantidadvariable -1
/****************************** CREAR TABLA TEMPORAL ************************************/
/* Parametros Crear tabla */
DECLARE @tablatemporal VARCHAR(MAX)
SET @tablatemporal = 'DECLARE @tablatemporal TABLE (idobservacion INT NULL, '
DECLARE @contadorcreartabtemp INT
SET @contadorcreartabtemp = 0
DECLARE @nombrevariab VARCHAR(MAX)
SET @nombrevariab = ''
DECLARE @descrionvariab VARCHAR(MAX)
SET @descrionvariab = ''
DECLARE cursorvariables CURSOR SCROLL FOR...
OPEN cursorvariables
FETCH FIRST FROM cursorvariables INTO @nombrevariab,@descrionvariab
WHILE @@FETCH_STATUS = 0...
/*script creacion table temporal completo*/
SET @tablatemporal = @tablatemporal + ')'
SET @scriptcompleto =@tablatemporal
CLOSE cursorvariables
DEALLOCATE cursorvariables
/************************ TERMINAR CREACION TABLA TEMPORAL ***/

/********************** CREAR DATA Y INGRESAR DATA TABLA TEMPORAL ***/
/*PARAMETROS*/
DECLARE @data VARCHAR(MAX)
SET @data =''
DECLARE @datatabltemp VARCHAR(MAX)
SET @datatabltemp = ''
DECLARE @idobservacion INT
DECLARE @valor VARCHAR(MAX)
DECLARE @nombrevariable VARCHAR(40)
DECLARE @contadorIdoObservacion INT
DECLARE @contadorDatosVariable INT
SET @contadorIdoObservacion = 0
DECLARE @informacionDatoVariable VARCHAR(MAX)
DECLARE @informacionTabla VARCHAR(MAX)
DECLARE cursorIdobservacion CURSOR SCROLL FOR...
OPEN cursorIdobservacion
FETCH FIRST FROM cursorIdobservacion INTO @idobservacion
WHILE @@FETCH_STATUS = 0...
SET @datatabltemp = ' INSERT INTO @tablatemporal VALUES'+@datatabltemp + ')'
SET @scriptcompleto = @scriptcompleto + @datatabltemp
CLOSE cursorIdobservacion
DEALLOCATE cursorIdobservacion
/****************************** PROCESAR DATOS EN CADENA   ************************************/
/* PDC*/
DECLARE @comando VARCHAR(MAX)
SET @comando = ''
IF @idVariable = 'NO'
SET @comando = ' SELECT '+ @tipoSumarizacion +'('+@idVariableReferente + ') AS TOTAL'+' FROM @tablatemporal'
ELSE
SET @comando = ' SELECT ' + @idVariable+','+ @tipoSumarizacion +'('+@idVariableReferente + ') AS TOTAL'+' FROM @tablatemporal GROUP BY '+ @idVariable
SET @scriptcompleto = @scriptcompleto + @comando
PRINT(@scriptcompleto)
EXEC(@scriptcompleto)
END
GO
```

El código usado para generar el archivo xls se muestra a continuación.

Ilustración 43. Código usado para generar el archivo xls de resultado de sumarización.

```
1 reference | kevin, 36 days ago | 1 author, 2 changes
private void exportToExcel(System.Data.DataTable dataTable)
{
    if (dataTable != null){
        try{
                Response.ClearContent();
                System.IO.StringWriter write = new System.IO.StringWriter();
                System.Web.UI.HtmlTextWriter htmlWrite = new HtmlTextWriter(write);

                GridView gridViewsExport = new GridView();
                gridViewsExport.DataSource = dataTable;
                gridViewsExport.DataBind();
                gridViewsExport.RenderControl(htmlWrite);
                Response.AddHeader("content-disposition", "attachment;filename=SUMARIZACION.xls");
                Response.AddHeader("Content-Type", "application/vnd.ms-excel");
                this.EnableViewState = false;
                Response.Write(write.ToString());
                Response.End();
            }catch (Exception detalleError){
                mostrarMensajeScript("Error", "Error al crear Archivo .xls .", "error");
            }
     }else
       mostrarMensajeScript("Error!", "Error al crear Archivo .xls .", "error");

}
```

4.1.3.4. Análisis Correlación

Se realiza siguiendo los mismos procesos del análisis de sumarización, con ciertas modificaciones. La web se encarga de mostrar la interfaz de configuración. Código para la crear de variables dinámicas y mostrar en pantalla.

Ilustración 44. Código crear variables dinámicas y mostrar en pantalla.

```
#region Configuracion de componentes de WebControls
1 reference | kevin, 31 days ago | 1 author, 2 changes
private Label crearLabel(string id, string etiqueta){
    Label lblNuevo = new Label();lblNuevo.ID = id;lblNuevo.Text = etiqueta.ToUpper();
    lblNuevo.CssClass = "control-label"; return lblNuevo;
}
1 reference | kevin, 46 days ago | 1 author, 1 change
private Literal crearLiteralHTML(string id, string contenido){
    Literal ltr = new Literal();ltr.ID = id;
    ltr.Text = contenido;return ltr;
}
1 reference | kevin, 45 days ago | 1 author, 2 changes
private CheckBox crearCheckBox(string id){
    CheckBox checkBoxNuevo = new CheckBox();checkBoxNuevo.Checked = false;
    checkBoxNuevo.ID = id;return checkBoxNuevo;
}
#endregion
0 references | kevin, 37 days ago | 1 author, 4 changes
protected void Page_Load(object sender, EventArgs e)
{

    if (!Page.IsPostBack) {
        //CargarDropdowlist
        configurarDropDownList("ddltipoCorrelacion",
            new Controller.TipoCorrelacionController().listaTipoCorrelacion()
            ,"nombretipocorrelacion","idtipocorrelacion");
    }
    //Obtener las variables a presentar
    int idProyecto = obtenerIdProyectoDSession();
    if (!idProyecto.Equals(-1)){
        Models.tbproyectoinv proyecto = new Controller.ProyectoInvController()
                                        .buscarProyectoXid(idProyecto);
        if (proyecto != null){
            foreach (Models.tbvariable itemVariable in proyecto.tbvariable){
                if (verificarTipoVariableNumerica(itemVariable.tbtipovariable.tipo)) {
                    //Agregar las variables al panelControl
                    panelCampos.Controls.Add(crearLabel(
                                                LABEL + itemVariable.idvariable.ToString()
                                                , itemVariable.nombrevariable));
                    panelCampos.Controls.Add(crearCheckBox(CHECKBOX + itemVariable.idvariable.ToString()));
                    panelCampos.Controls.Add(crearLiteralHTML(
                                                LITERAL + itemVariable.idvariable.ToString()
                                                , "<br/>"));
                }
            }
        }else return;
    }
}
```

Se deben establecer parámetros como el coeficiente de correlación a utilizar, las opciones soportadas son Pearson, Spearman y seleccionar las variables que se correlacionaran. Al terminar de configurar, se solicitará el cálculo. El resultado será la matriz de correlación la cual se mostrará como una tabla utilizando etiquetas HTML. Procesos llevados a cabo para calcular la correlación se detallan a continuación. La configuración del análisis se envía a una clase controlador de la aplicación. La misma que se encarga de llamar al script de organizar los datos llamado pcdobtenerdatosproyectoinv. Código fuente del script.

Ilustración 45. Script obtener datos del proyecto de investigación.

```
ALTER PROCEDURE [dbo].[pcdobtenerdatosproyectoinv]
@idproyecto INT
AS
BEGIN

DECLARE @idobservacion INT
DECLARE @valor VARCHAR(MAX)
DECLARE @nombrevariable VARCHAR(40)

DECLARE @contadorIdoObservacion INT
DECLARE @contadorDatosVariable INT
SET @contadorIdoObservacion = 0

DECLARE @informacionDatoVariable VARCHAR(MAX)
DECLARE @informacionTabla VARCHAR(MAX)

DECLARE cursorIdobservacion CURSOR SCROLL FOR
(SELECT tbdatosvariable.idobservacion FROM tbobservacion...
OPEN cursorIdobservacion
FETCH FIRST FROM cursorIdobservacion INTO @idobservacion
WHILE @@FETCH_STATUS = 0
BEGIN
    DECLARE cursorDatos CURSOR SCROLL FOR
    (SELECT tbvariable.nombrevariable, tbdatosvariable.valor...
    SET @informacionDatoVariable =''
    SET @contadorDatosVariable =0
    OPEN cursorDatos
    FETCH FIRST FROM cursorDatos INTO @nombrevariable,@valor
    WHILE @@FETCH_STATUS = 0
    BEGIN
        SET @contadorDatosVariable += 1
        IF @contadorDatosVariable = 1
            SET @informacionDatoVariable = @informacionDatoVariable + 'SELECT OBSERVACION='
            +CONVERT(varchar,@idobservacion)+','
            +@nombrevariable+'='''+CONVERT(varchar(MAX),@valor)+''''
        ELSE
            SET @informacionDatoVariable = @informacionDatoVariable + ','
            +@nombrevariable+'='''+CONVERT(varchar(MAX),@valor)+''''

        FETCH NEXT FROM cursorDatos INTO @nombrevariable,@valor
    END
    CLOSE cursorDatos
    DEALLOCATE cursorDatos

    SET @contadorIdoObservacion +=1
    IF @contadorIdoObservacion =1
        SET @informacionTabla = @informacionDatoVariable
    ELSE
        SET @informacionTabla = @informacionTabla + ' UNION ' + @informacionDatoVariable
    FETCH NEXT FROM cursorIdobservacion INTO @idobservacion
END
EXEC(@informacionTabla)
CLOSE cursorIdobservacion
DEALLOCATE cursorIdobservacion
END
```

El cual devuelve los datos en forma de DataTable, los mismos que serán enviados a la clase controlador CorrelacionController que implementa los métodos de la librería Math.NET Numerics que contiene los métodos de correlación Pearson, Spearman. Código fuente de la clase CorrelacionController.

Ilustración 46. Código fuente para calcular Correlaciones.

```
public double calcularCorrelacionSpearman(List<double> valoresVariable1
                                          , List<double> valoresVariable2)
{
    try {
        return Correlation.Spearman(valoresVariable1, valoresVariable2);
    } catch (Exception detalleError) {
        return double.NaN;
    }
}
1 reference | kevin, 44 days ago | 1 author, 1 change
public Matrix<double> calcularCorrelacionSpearman(double[][] valoresVariables)
{
    try{
        return Correlation.SpearmanMatrix(valoresVariables);
    }catch (Exception detalleError){
        return null;
    }
}
1 reference | kevin, 40 days ago | 1 author, 2 changes
public double calcularCorrelacionPearson(List<double> valoresVariable1
                                         , List<double> valoresVariable2)
{
    try{
        return Correlation.Pearson(valoresVariable1, valoresVariable2);
    }catch (Exception detalleError){
        return double.NaN;
    }
}
1 reference | kevin, 44 days ago | 1 author, 1 change
public Matrix<double> calcularCorrelacionPearson(double[][] valoresVariables)
{
    try{
        return Correlation.PearsonMatrix(valoresVariables);
    }catch (Exception detalleError){
        return null;
    }
}
```

El código fuente usado para mostrar los resultados en tabla HTML es el siguiente.

Ilustración 47. Código fuente para convertir tabla a etiquetas HTML.

```
2 references | kevin, 30 days ago | 1 author, 3 changes
private string convertirAEtiquetaTable(MathNet.Numerics.LinearAlgebra.Matrix<double> matrixCorrelacion, List<string> nombreVariables)
{
    string etiquetaTAble = "<h3>Tabla de Correlación</h3> <table class='table table-bordered' style='border-collapse:collapse;' border='1'>";

    //Encabezado columna
    etiquetaTAble = etiquetaTAble + "<tr> <th style='color:White;background-color:#02B3E4;'></th> ";
    foreach (string item in nombreVariables)
        etiquetaTAble = etiquetaTAble + "<th style='color:White;background-color:#02B3E4;'>" + item + "</th>";
    etiquetaTAble = etiquetaTAble + "</tr>";
    //Datos
    for (int contadorColumna = 0; contadorColumna < matrixCorrelacion.ColumnCount; contadorColumna++)
    {
        etiquetaTAble = etiquetaTAble + "<tr>";
        etiquetaTAble = etiquetaTAble + "<td>" + nombreVariables[contadorColumna] + "</td>";
        for (int contadorFila = 0; contadorFila < matrixCorrelacion.RowCount; contadorFila++)
        {
            etiquetaTAble = etiquetaTAble + "<td>" + matrixCorrelacion[contadorFila, contadorColumna] + "</td>";
        }
        etiquetaTAble = etiquetaTAble + "</tr>";
    }
    etiquetaTAble = etiquetaTAble + "</table>";
    return etiquetaTAble;
}
```

4.1.3.5. Análisis de minería SSAS

Con el objetivo de integrar el análisis SSAS a la aplicación PIentomolgia se debió configurar e implementar varios componentes, los cuales se detallan a continuación. **Servidor SSAS** funciona como servidor de base de datos analítica que hospeda modelos tabulares, como modelos de minería de datos. La instalación se llevó a cabo mediante el uso del asistente de SQL Server. Configuración necesaria fue la especificación del modo del Servidor, se seleccionó modo multidimensional y de minería de datos.

Ilustración 48. Configuración del servidor SSAS.

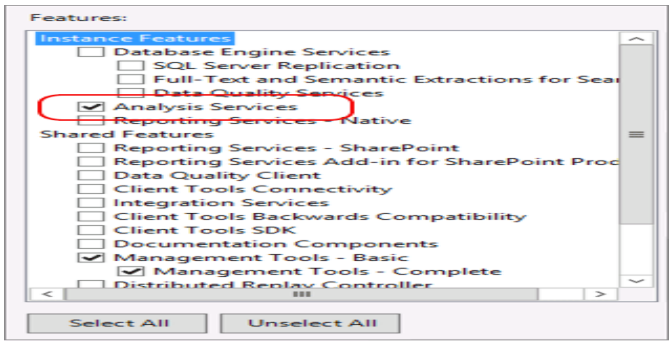
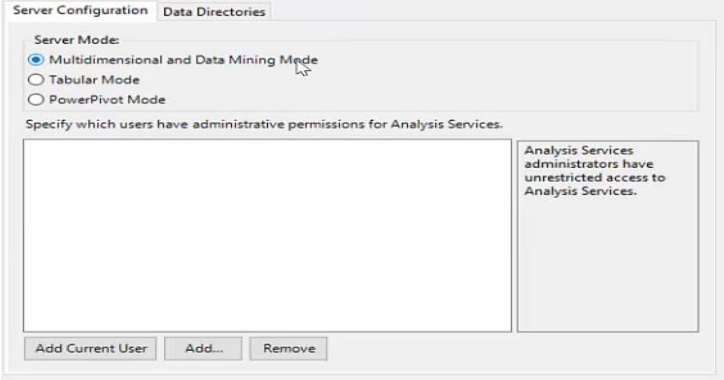
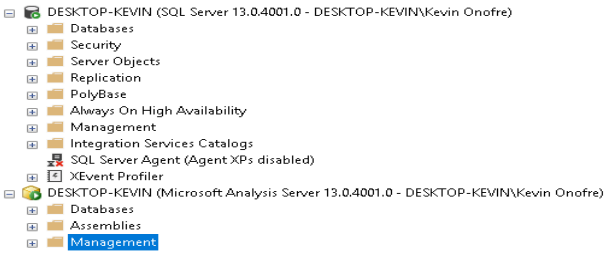


Ilustración 49. Configuración en modo de Modelo de minería.



Estado final de los servidores de base datos que usa la aplicación:

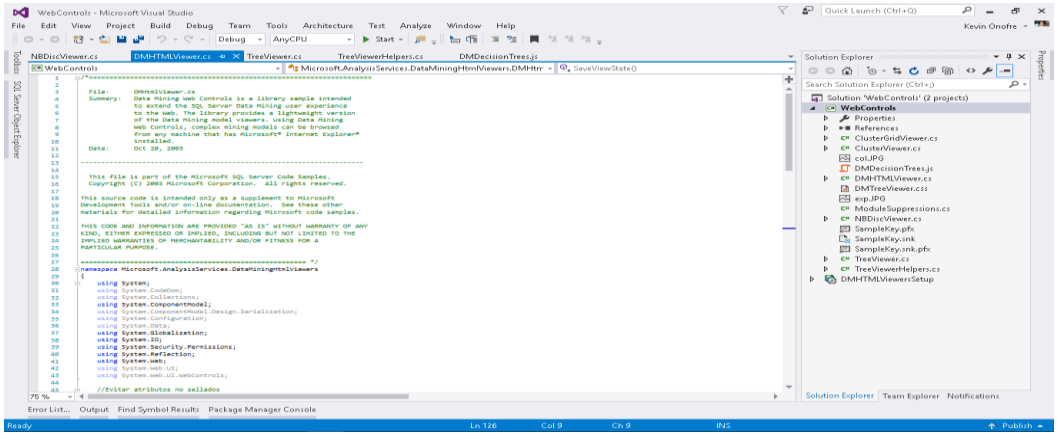
Ilustración 50. Servidores de base de datos de PIentomología.



Otro componente necesario fue el control web de minería de datos. El control es una biblioteca destinada a extender la experiencia del usuario de SQL Sever Data Mining a la Web. La biblioteca proporciona una versión ligera de los visores de modelos de Data Mining. El control funciona del lado del servidor es decir viven en el espacio de memoria de un servidor web. Por lo tanto, se debe instalar en la misma máquina donde se encuentra el servidor web y el servidor de base de datos relacional y de analítica.

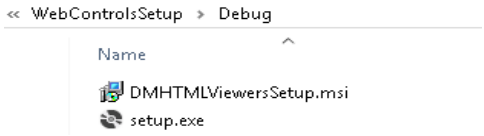
Para instalar el control se debió descargar la solución WebControls.sln del sitio oficio https://archive.codeplex.com/?p=msftasprodsamples, el archivo contiene el código fuente de la biblioteca. La solución se abre en el Visual Studio donde se procese a la reconstrucción.

Ilustración 51. Pantalla de Visual Studio de la solución WebControls.



Archivos de instalación generados por la reconstrucción de la solución WebControls.sln.

Ilustración 52. Archivos de instalación del componente Web.

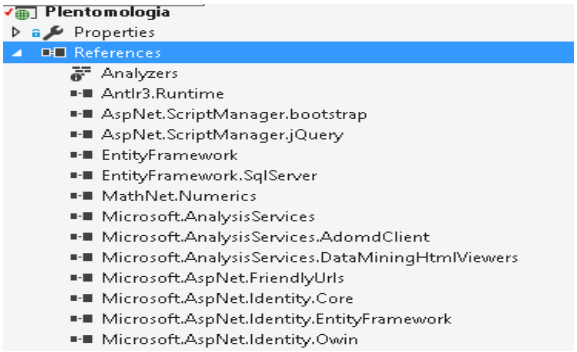


Se procede a instalar el componente. Al finalizar el proceso, se habrá agregado las siguientes carpetas al servidor.

Ilustración 53. Archivos en el servidor.

La .dll contiene la biblioteca y el código fuente de los controles. Este archivo se agregará a las bibliotecas del proyecto PIentomología.

Ilustración 54. Bibliotecas de PIentomología



La biblioteca soporta los siguientes controles web de minería: DMClusterViewer para algoritmo de Clustering de Microsoft y DMNaiveBayesViewer para el algoritmo Microsoft_Naive_Bayes.

Creación del análisis SSAS se realiza siguiendo los mismos procesos de los análisis antes mencionados, con ciertas modificaciones. La web se encarga de mostrar la interfaz de configuración. Código para la crear de variables dinámicas y mostrar en pantalla.

Ilustración 55. Código fuente para crear variables dinámicas y mostrar en pantalla.

```
protected void Page_Load(object sender, EventArgs e)
{
    if (!Page.IsPostBack)
    {//CargarDropdowlist
        configurarDropDownList("ddlTecnicaMineria",new Controller.TecnicaMineriaDatosController().listaTecnicaMineria()
            ,"nombretecnicamineria","idtecnicamineriadatos");
    }
     //Obtener las variables a presentar
        int idProyecto = obtenerIdProyectoDSession();
    if (!idProyecto.Equals(-1)){
        Models.tbproyectoinv proyecto = new Controller.ProyectoInvController()
                                        .buscarProyectoXid(idProyecto);

        if (proyecto != null) {
            foreach (Models.tbvariable itemVariable in proyecto.tbvariable) {
                if (esAnalizableSSAS(itemVariable.tbtipovariable.tipo)) {
                //Agregar las variables al panelControl
                panelCampos.Controls.Add(
                                        crearLabel(LABEL + itemVariable.idvariable.ToString()
                                                , itemVariable.nombrevariable + ":   "));
                panelCampos.Controls.Add(
                                        crearLiteralHTML(LITERAL + itemVariable.idvariable.ToString()
                                        , "<br/><br/><span class='control-label'>Tipo Contenido: </span>"));
                panelCampos.Controls.Add(
                                        crearDropDowList(DROPDOWLIST + itemVariable.idvariable.ToString()));
                panelCampos.Controls.Add(
                                        crearLiteralHTML(LITERAL+1.ToString() + itemVariable.idvariable.ToString()
                                        , "<br/><span class='control-label'>Defina: </span>"));
                panelCampos.Controls.Add(
                                        crearRadioButtonList(RADIOBUTTONLIST + itemVariable.idvariable.ToString()));

                }
            }
        }
    }
}
```

Se debe establecer la configurar del modelo de minería. El nombre, la descripción, el estado (procesado, no procesado), porcentaje de datos de prueba, el número máximo de casos en el conjunto de datos de prueba, la técnica (Regresión lineal, arboles de decisión, Naive Bayes, clústeres) y configurar las variables. En la configuración de la variable se especifica el tipo de contenido (discretizado, discreta, cíclico, ordenado, continuo) y se define el tipo de uso que se le dará en el modelo (entrada, predecir). Al terminar de configurar, se solicitará el cálculo. El resultado será visualizado en la web con la ayuda del controlador web. Procesos llevados a cabo para calcular el modelo de minería se detallan a continuación. La configuración del análisis se envía a una clase controlador de la aplicación. La misma que se encarga de gestionar los objetos SSAS mediante el uso de la librería AMO.

Ilustración 56. Código fuente de la clase control de objetos SSSAS.

```
using PIentomologia.Models.MineriaDatos;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace PIentomologia.Controller
{
    6 references | kevin, 31 days ago | 1 author, 2 changes
    public class BaseDatosSSASController
    {
        private ProveedorDatosAMO dataBaseSSAS;
        2 references | kevin, 32 days ago | 1 author, 1 change
        public BaseDatosSSASController() [...]

        0 references | kevin, 32 days ago | 1 author, 1 change
        public BaseDatosSSASController(string nombreServidorSSAS,string nombreBaseDatosSSAS) [...]

        1 reference | kevin, 31 days ago | 1 author, 1 change
        public bool conectarServidorSSAS() [...]
        1 reference | kevin, 32 days ago | 1 author, 1 change
        public bool crearBaseDatosSSAS(string nombre) [...]
        0 references | kevin, 32 days ago | 1 author, 1 change
        public bool conectarBaseDatosSSSAS(string nombre)[...]

        1 reference | kevin, 32 days ago | 1 author, 1 change
        public bool agregarDataSource(string nombreDataSource
                                      ,string nombreServidorBaseDatos
                                      ,string nombreBaseDatos
                                      ,string tipoInfoSuplantacion
                                      ,string usuarioInfoSuplantacion
                                      ,string contrasenaInfoSuplantacion
                                      ,string tipoAutenticacionServidorBaseDatos
                                      ,string usuarioConnecionBaseDatos
                                      ,string contrasenaConnecionBaseDatos) [...]
        1 reference | kevin, 32 days ago | 1 author, 1 change
        public bool agregarDataSourceView(string nombreDataSourceView
                                      ,string nombreDataSource
                                      ,string nombreTabla) [...]

        1 reference | kevin, 32 days ago | 1 author, 1 change
        public bool agregarMinigStructure(string nombreMiningStructure
                                      ,string nombreDataSourceView
                                      , List<ColumnaEstructuraMineria> columnas
                                      , int porcentajeDatosPrueba
                                      , int numMaxCasosConjuntoDatosPrueba) [...]

        1 reference | kevin, 32 days ago | 1 author, 1 change
        public bool agregarModelMining(string nombreStructureMining, string nombreMiningModel,
                                      List<ColumnaModeloMineria> columnas, string nombreAlgoritmo)

        1 reference | kevin, 32 days ago | 1 author, 1 change
        public string procesarModelMinig(String nombreStrutureMining
                                      , String nombreMiningModel) [...]
        2 references | kevin, 31 days ago | 1 author, 1 change
        public void eliminarBaseDatosSSAS(string nombre) [...]
        3 references | kevin, 32 days ago | 1 author, 1 change
        public void desconectar() [...]
    }
}
```

Por otra parte, la clase también se encarga de organizar los datos mediante el uso del script llamado pcdcreartablaproyectoinv, código fuente del script.

Ilustración 57. Script para crear tabla de datos del proyecto.

```
ALTER PROCEDURE [dbo].[pcdcreartablaproyectoinv]
@idproyecto INT,
@cantidadvariable INT,
@nombretabla NVARCHAR(MAX)
AS
BEGIN
/*PARAMETROS*/
DECLARE @scriptcompleto VARCHAR(MAX)
SET @scriptcompleto = ''
SET @cantidadvariable = @cantidadvariable -1
/****************************** CREAR TABLA TEMPORAL ************************************/
/* Parametros Crear tabla */
DECLARE @tablatemporal VARCHAR(MAX)
SET @tablatemporal = 'USE BDDatosMineria CREATE TABLE ' +@nombretabla +'(idobservacion INT NOT NULL,PRIMARY KEY (idobservacion), '
DECLARE @contadorcreartabtemp INT
SET @contadorcreartabtemp = 0
DECLARE @nombrevariab VARCHAR(MAX)
SET @nombrevariab = ''
DECLARE @descrionvariab VARCHAR(MAX)
SET @descrionvariab = ''
DECLARE cursorvariables CURSOR SCROLL FOR
SELECT [tbvariable].[nombrevariable],[tbtipovariable].[descripcion]
FROM [dbo].[tbvariable]
INNER JOIN [tbtipovariable] on [tbvariable].[idtipovariable] = [tbtipovariable].[idtipovariable]
WHERE [tbvariable].[idproyectoinv] = @idproyecto ORDER BY tbvariable.nombrevariable DESC
OPEN cursorvariables
FETCH FIRST FROM cursorvariables INTO @nombrevariab,@descrionvariab
WHILE @@FETCH_STATUS = 0
BEGIN
    SET @contadorcreartabtemp += 1
    SET @tablatemporal = @tablatemporal+  @nombrevariab+' '+@descrionvariab + ' NULL'
    if @contadorcreartabtemp < @cantidadvariable
    SET @tablatemporal = @tablatemporal+','

    FETCH NEXT FROM cursorvariables INTO @nombrevariab,@descrionvariab
END
/*script creacion table temporal completo*/
SET @tablatemporal = @tablatemporal + ')'
SET @scriptcompleto =@tablatemporal
CLOSE cursorvariables
DEALLOCATE cursorvariables
/********************** TERMINAR CREACION TABLA TEMPORAL ***/
```

```
/************************ CREAR DATA Y INGRESAR DATA TABLA TEMPORAL ***/
/*PARAMETROS*/
DECLARE @data VARCHAR(MAX)
SET @data =''
DECLARE @datatabltemp VARCHAR(MAX)
SET @datatabltemp = ''
DECLARE @idobservacion INT
DECLARE @valor VARCHAR(MAX)
DECLARE @nombrevariable VARCHAR(40)
DECLARE @contadorIdoObservacion INT
DECLARE @contadorDatosVariable INT
SET @contadorIdoObservacion = 0
DECLARE @informacionDatoVariable VARCHAR(MAX)
DECLARE @informacionTabla VARCHAR(MAX)
DECLARE cursorIdobservacion CURSOR SCROLL FOR...
OPEN cursorIdobservacion
FETCH FIRST FROM cursorIdobservacion INTO @idobservacion
WHILE @@FETCH_STATUS = 0...
    DECLARE cursorDatos CURSOR SCROLL FOR...

    SET @informacionDatoVariable =''
    SET @contadorDatosVariable =0

    OPEN cursorDatos
    FETCH FIRST FROM cursorDatos INTO @nombrevariable,@valor
    WHILE @@FETCH_STATUS = 0...
    CLOSE cursorDatos
    DEALLOCATE cursorDatos

    SET @contadorIdoObservacion +=1
    IF @contadorIdoObservacion =1 ...
    FETCH NEXT FROM cursorIdobservacion INTO @idobservacion
END

SET @datatabltemp = ' INSERT INTO ' +@nombretabla +' VALUES'+@datatabltemp + ')'
SET @scriptcompleto = @scriptcompleto + @datatabltemp
CLOSE cursorIdobservacion
DEALLOCATE cursorIdobservacion

/****************************** CREAR TABLA Y DATOS EN BDDatosMineria  ************************************/
/* PDC*/
DECLARE @comando NVARCHAR(MAX)
SET @comando = ''

SET @comando = @comando + @scriptcompleto
EXEC sp_executesql @comando
```

El script organiza los datos y crea una tabla de los mismo en la base de datos BDDatosMineria, esta base de datos se encarga de almacenar los registros en una sola tabla los mismo que servirán como DataSource para el análisis.

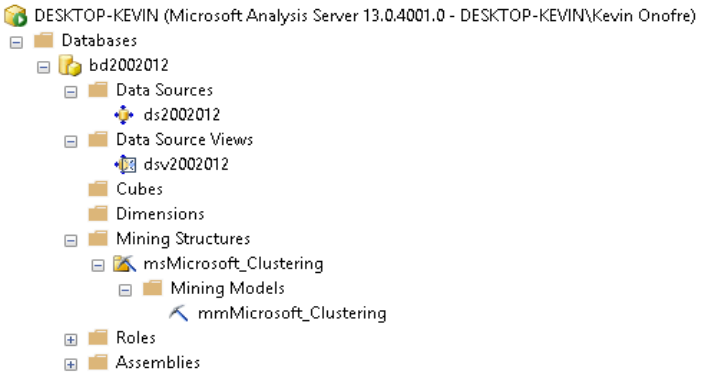
Otra tarea prioritaria para el análisis de minería es crear los objetos necesarios para efectuar el análisis. Los objetos necesarios fueron estudiados en la sección de conceptos clave para formular e implementar un escenario de minería del marco conceptual del capítulo fundamentación teórica de la investigación. A continuación, se muestra la codificación de los objetos:

Ilustración 58. Código para crear objetos para la minería de datos.

```
Controller.BaseDatosSSASController baseDatoSSAS = new Controller.BaseDatosSSASController();
bool okBaseDatoSSAS = false;
if (baseDatoSSAS.crearBaseDatosSSAS(BDSSAS.NombreBaseDatos))
{
    if (baseDatoSSAS.agregarDataSource(fuenteDatos.NombreDataSource
                                       , fuenteDatos.NombreServidorBaseDatos
                                       , fuenteDatos.NombreBaseDatos
                                       , fuenteDatos.TipoInfoSuplantacion
                                       , fuenteDatos.UsuarioInfoSuplantacion
                                       , fuenteDatos.ContrasenaInfoSuplantacion
                                       , fuenteDatos.TipoAutenticacionServidorBaseDatos
                                       , fuenteDatos.UsuarioConnecionBaseDatos
                                       , fuenteDatos.ContrasenaConnecionBaseDatos))
    {
        if (baseDatoSSAS.agregarDataSourceView(vistaFuenteDatos.NombreDataSourceViews
                                               , fuenteDatos.NombreDataSource
                                               , vistaFuenteDatos.NombreTabla))
        {
            if (baseDatoSSAS.agregarMinigStructure(estructuraMineria.NombreStructureMineria
                                                   , vistaFuenteDatos.NombreDataSourceViews
                                                   , estructuraMineria.ColumnasEstructuraMineria
                                                   , estructuraMineria.PorcentajeDatosPrueba
                                                   , estructuraMineria.NumMaxCasosConjuntoDatosPrueba))
            {
                if (baseDatoSSAS.agregarModelMining(estructuraMineria.NombreStructureMineria
                                                    , modeloMineria.NombreModeloMineria
                                                    , modeloMineria.ColumnasModeloMineria
                                                    , modeloMineria.Tecnica))
                    okBaseDatoSSAS = true;
            }
        }
    }
}
```

Objetos creados en el servidor SSAS.

Ilustración 59. Objetos en el servidor SSAS.



La ejecución del modelo de minería se lleva a cabo mediante las siguientes líneas de código.

Ilustración 60. Código fuente para ejecutar modelo de minería.

```
//Verificar el modelo de mineria fue creado
if (okBaseDatoSSAS)
{
    string estadoProcesoMineria = baseDatoSSAS
                                      .procesarModelMinig(
                                                          estructuraMineria.NombreStructureMineria,
                                                          modeloMineria.NombreModeloMineria);

    baseDatoSSAS.desconectar();
    establecerMensaOk(idModelomineria, estadoProcesoMineria);
    Response.Redirect("~/Analizar/SSASok");
}
```

El usuario debe esperar que el proceso termine para poder continuar con la visualización de los resultados. Los controles web proporcionan la visualización en la web, código para mostrar el componente.

Ilustración 61. Código fuente de los componentes web de minería.

```
<cc1:DMClusterViewer ID="DMClusterViewer1" runat="server" Visible="false"
                     BorderColor="Black" BorderStyle="Solid" BorderWidth="1px"
                     Height="550px" Width="900px" ForeColor="Black" EnableTheming = "true"/>
<cc1:DMNaiveBayesViewer ID="DMNaiveBayesViewer1" runat="server" Visible="false"
                        ViewerMode="Characteristics"/>
```

Dado que son una versión ligera de los visores de modelos de SSAS. Se hace necesario proporcionar el detalle de los nodos mediante un archivo plano. El acceso a la información del modelo de minería se realiza mediante la clase ProveedorDatosAdomd el cual implementa métodos de la librería AdomdClient que es un proveedor de datos de Microsoft .NET Framework diseñado para comunicarse con Microsoft SSAS.

Ilustración 62. Código fuente de Proveedor datos Adomd

```
using Microsoft.AnalysisServices.AdomdClient;

namespace WebApplicationWSSSAS.Models
{
    8 references | kevin, 54 days ago | 1 author, 1 change
    public class ProveedorDatosAdomd
    {
        //private String CADENACONEXION = "Data Source=DESKTOP-KEVIN; Initial Catalog=MultidimensionalProjectAW2012";
        private String CADENACONEXION = "Data Source=dataSourceName; Initial Catalog=initialCatalogName";
        private AdomdConnection conecion;
        private AdomdDataReader leectorData;
        0 references | kevin, 54 days ago | 1 author, 1 change
        public ProveedorDatosAdomd()...
        1 reference | kevin, 54 days ago | 1 author, 1 change
        public void formarCadenaConeccion(String dataSource, String initialCatalog)...

        /// <summary> Constructor Formatea la cadena de coneccion
        3 references | kevin, 54 days ago | 1 author, 1 change
        public ProveedorDatosAdomd(String nombreDataSource ,String nombreInitialCatalog)...
        3 references | kevin, 54 days ago | 1 author, 1 change
        public Boolean Conectar()...
        2 references | kevin, 54 days ago | 1 author, 1 change
        public void Cerrar()...
        2 references | kevin, 54 days ago | 1 author, 1 change
        public AdomdDataReader consultar(String comando) {
            //AdomdDataReader leectorData = null;
            AdomdCommand ejecutorComando = new AdomdCommand();

            try
            {
                if ((conecion == null) && (conecion.State == System.Data.ConnectionState.Closed))
                { return null;
                    //throw new ArgumentNullException("Connection");
                }
                ejecutorComando.Connection = conecion;
                ejecutorComando.CommandText = comando;
                leectorData = ejecutorComando.ExecuteReader();

            }
            catch (AdomdErrorResponseException) { ejecutorComando = null; throw; }
            return leectorData;
        }
    }
}
```

Para finalizar se elaboraron los manuales de usuario, evidencia en el **anexo 6**.

4.1.4. Fase transición

La finalidad de esta fase es ajustar errores, defectos encontrados en las integraciones de los componentes.

- Se solucionó errores de código repetitivo, en las clases que generan controladores de servidor web que se muestran en la página web.
- Se solucionó errores en el script para la transformar datos en tablas procesables, uno de los errores fue que el número de expresiones de valor de fila en la instrucción INSERT excedía el número máximo permitido de 1000 valores de fila.
- Se solucionó error Execution Timeout Expired (tiempo de espera caducado). Sucede porque el periodo de espera transcurrió antes de completar la operación.

Adicionalmente se describe el conjunto de datos mencionado en el capítulo 3 metodología de la investigación sección fuentes de recopilación de información sub sección fuentes secundarias. Los datos contienen información sobre observaciones de insectos concretamente del espécimen llamado Premnobius cavipennis, fueron capturados en plantaciones de balsa en tres localidades (Vergel, Guapara, Empalme) de la zona central del litoral ecuatoriano. A continuación, se describen las características del conjunto de datos llamado Datos_resultados_P.I_AVEROS_JOSELYN.xlsx, localidades:

- El Vergel cuenta con 1891 registros, número de variables 7.
- Guapara cuenta con 1891 registros, número de variables 7.
- El Empalme cuenta con 1531 registros, número de variables 7.

Análisis descriptivos de las variables se detalla en la Tabla 31.

Se dividió el archivo Datos_resultados_P.I_AVEROS_JOSELYN.xlsx en tres archivos extensión csv, cada uno contiene datos de localidad, esta manera permite organizar e ingresar rápido los datos a la aplicación PIentomología. En el anexo 5 se evidencia el código para ingresar los datos en la aplicación.

Tabla 31. Descripción de variables observación de insecto.

Variables	Alias	Medida	Valores posibles	Descripción
Fecha	fecha	Tiempo		Es la fecha de captura del insecto en estudio.
Semana	semana	Numérica	1 hasta 42	Es el número de semana, en el que se realiza la recolección de las capturas.
Bloque	bloque	Numérica	1, 2, 3, 4, 5	Es el punto de muestreo, donde se ubican tres postes que tendrán la trampa para cada una de las alturas.
Altura	altura	Metros	1 ,4 ,8	Es la altura en la que se encuentra ubicada la trampa del suelo.
Trampa	trampa	Numérica	1, 2 ,3	Es el número que identifica a cada trampa.
Nombre especie	nombreespecie	Cadena	P. cavipemnis	Es el nombre de la especie del insecto en estudio.
Cantidad especie	cantidadespecie	Numérica		Corresponde al conteo del insecto en estudio.

4.2. Discusión

Comparando las metodologías de desarrollo, los proyectos estudiados en el marco referencial utilizaron SCRUM, que les permitió cumplir con los requerimientos, al igual que la metodología RUP aquí utilizada, es decir que ambas metodologías resultaron útiles para cumplir con las especificaciones y en la obtención de los resultados deseados.

En [38] presentan un diseño de base de datos donde se almacena información del insecto como el orden, la familia, especie lo que permite informar a los usuarios sobre la descripción taxonómica de los organismos recolectados. En contraste con el diseño de la base de datos desarrollada en esta investigación (BDPIentomología), se observa una mejora por cuanto

permite especificar la información que se almacenara del insecto y por otra parte realizar operaciones sobre los datos.

En [27] consideran que uno de los factores para que el proyecto concluyera en el periodo esperado fue el uso de frameworks de terceros como la biblioteca de Telerik. En PIentomología fue importante la utilización de librerías de terceros para lograr los objetivos, una de las librerías de gran ayuda fue OpenCage.Geocode que permitió la localización de puntos mediante las coordenadas geográficas.

El proyecto revisado en [8] se caracterizaba por utilizar cubos OLAP; presenta una solución donde los datos de origen se extraen de un sistema externo y luego se transforman en una forma más estructural que se carga en un cubo OLAP SSAS, la interacción con OLAD la realizan mediante clases como MdxQuery que implementan los métodos necesario para comunicarse mediante declaraciones/consultas MDX. Mientras que en el sistema que aquí se ha desarrollado se implementa la minería SSAS haciendo uso de las clases y métodos de AMO. Las declaraciones/consultas MDX se usaron para obtener detalles de los modelos de minería procesados por AMO.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

Los objetivos planteados en el presente proyecto se cumplieron. Los mismos permiten concluir:

- Se consiguió implementar una solución capaz de gestionar datos entomológicos, incorporando los mecanismos necesarios para recolectar observaciones, lo cual permite que el usuario pueda guardar la ubicación (latitud y longitud), subir fotos o guardar diferentes datos previamente especificados, logrando crear información enriquecida. Además, durante el análisis de requerimientos se determinó como importante tener una funcionalidad para exportar datos de las recolecciones hacia otros formatos utilizados, a fin de facilitar la migración a otros paquetes estadísticos. La API de geolocalización Opencage se pudo integrar a la aplicación desarrollada en esta investigación facilitando la visualización espacial de los datos almacenados en el sistema.
- Se obtuvo un diseño de base de datos que permite la creación de proyectos de estudio, detallando los datos a recolectar especificando el tipo de contenido que puede almacenar. Entre los tipos soportados se tiene cadena de texto, fecha, fotos, números enteros y decimales. Esto permitirá a los investigadores crear sus esquemas personalizados según sus proyectos, asumiendo de igual manera la responsabilidad de sus planteamientos en cuanto a la estructura de sus datos y los resultados de los análisis estadísticos.
- La captura de datos mediante la aplicación móvil facilita la recolección en lugares de difícil acceso; por otro lado, la aplicación web permite complementar la información capturada en el sitio; es decir, se mejora la eficiencia de la recolección de datos mediante un registro *a priori* y *a posteriori*.
- Las técnicas estadísticas implementadas en el proyecto son: coeficiente de correlación (Pearson, Spearman) y de resumen de datos (sumarización). Adicionalmente cuenta con otra sección de análisis donde se puede realizar regresión lineal. Esto permite a los investigadores acceder a otras técnicas de análisis que les proporcionan perspectivas adicionales a las que obtenían mediante el proceso manual, con hojas electrónica u otro software estadístico sencillo.

5.2. Recomendaciones

En versiones futuras la aplicación Web se podría mejorar en puntos como:

- El registro de las observaciones puede tornarse tedioso, se recomienda desarrollar una interfaz de ingreso de observaciones mediante hojas electrónicas.
- La aplicación cuenta con un plan free trial que soporta 2500 solicitudes por día, para ubicar puntos y devolver la ubicación del lugar. Se sugiere que para investigaciones más complejas se contrate un plan como x-small que soporta hasta 10000 solicitudes por día, 10 peticiones/segundos o uno mejor en Opencagedata.
- El proceso de ejecutar los distintos tipos de análisis puede tomar un tiempo superior al permitido en una sesión web; por lo que es conveniente desarrollar ejecuciones en segundo plano para evitar el tiempo de espera por parte del usuario.
- Utilizar librerías JavaScript para desarrollar componentes web para mostrar vistas detalladas de los modelos de minera de datos.

Sobre el correcto uso del sistema PIentomología:

- Evitar la navegación del sitio en Microsoft Edge.
- Utilizar dispositivos móviles con sistema Android con versión superior a 4.1 (Jelly Bean).
- Activar la localización GPS al momento de recolectar datos.
- Activar la opción fuentes desconocidas al momento de instalar Api PIentomología.
- Verificar que el dispositivo cuente con una cámara en buen estado.
- Proporcionar los permisos de almacenamiento, cámara, ubicación a la Api PIentomología.
- Mantener activo la conexión de internet o datos móviles al momento de realizar las observaciones.
- Capacitar adecuadamente a los usuarios sobre el uso del software (**ver anexo 6**).

CAPITULO VI
BIBLIOGRAFÍA

6.1. Bibliografía

[1] Z. C. U. y. R. B. M. Á. C. N. Llorente de la Torre, Guía de la Arquitectura N-capas orientada al Dominio con .Net 4.0, España: Krasis Press, 2010.

[2] M. Potel, Artist, MVP: Model-View-Presenter The Taligent Programming Model for C++ and Java. [Art]. IBM Mainstream , 1996.

[3] S. M. Roos, Introducción a la estadística, Reverte, 2007.

[4] R. Á. Cáceres, Estadística aplicada a las ciencias de la salud, Ediciones Díaz de Santos, 2007.

[5] A. V. Sabadías, Estadística descriptiva e inferencial, Univ de Castilla La Mancha, 1995.

[6] A. W. G. G. Usama M. Fayyad, Introduccion to Data mining and Knowledge discovey, San Francisco: Morgan Kaufmann Publishers, 2006.

[7] C. Pérez, Minería de datos: técnicas y herramientas, Parainfo: Parainfo, 2007.

[8] A. a. S. E. Bozkir, Artist, A new Web Based Data Mining Exploration and Reporting Tool for Decision Makers. [Art]. Department of Computer Engineering, Hacettepe University, 2013.

[9] Z. T. B. C. Jamie MacLennan, Data mining with microsoft Sql server 2008, Wiley Publishing, Inc, 2011.

[10] O. Ducan, «Content Types (Data Mining),» Microsoft, 30 Abrir 2018. [En línea]. Available: https://docs.microsoft.com/es-es/sql/analysis-services/data-mining/content-types-data-mining?view=sql-server-2017. [Último acceso: 15 Octubre 2018].

[11] O. Duncan, «Mining Model Columns,» Microsoft, 07 Agosto 2018. [En línea]. Available: https://docs.microsoft.com/es-mx/sql/analysis-services/data-mining/mining-model-columns?view=sql-server-2017. [Último acceso: 15 Octubre 2018].

[12] O. Duncan, «Microsoft Naive Bayes Algorithm,» Microsoft, 07 Mayo 2018. [En línea]. Available: https://docs.microsoft.com/es-mx/sql/analysis-services/data-mining/microsoft-naive-bayes-algorithm?view=sql-server-2017. [Último acceso: 15 Octubre 2018].

[13] O. Duncan, «Microsoft Clustering Algorithm,» Microsoft, 07 Mayo 2018. [En línea]. Available: https://docs.microsoft.com/es-mx/sql/analysis-services/data-mining/microsoft-clustering-algorithm?view=sql-server-2017. [Último acceso: 15 Octubre 2018].

[14] O. Duncan, «Referencia técnica del algoritmo de regresión lineal de Microsoft,» 07 Mayo 2018. [En línea]. Available: https://docs.microsoft.com/es-es/sql/analysis-services/data-mining/microsoft-linear-regression-algorithm-technical-reference?view=sql-server-2017. [Último acceso: 10 Enero 2019].

[15] O. Duncan, «Microsoft Decision Trees Algorithm Technical Reference,» 07 Mayo 2018. [En línea]. Available: https://docs.microsoft.com/es-es/sql/analysis-services/data-mining/microsoft-decision-trees-algorithm-technical-reference?view=sql-server-2017. [Último acceso: 07 Enero 2019].

[16] F. J. G. O. Alberto Luceño Vázquez, Métodos estadísticos para medir, describir y controlar la variabilidad, Ed. Universidad de Cantabria, 2004.

[17] S. Owen, «Examinar un modelo usando el Visor de árbol de contenido genérico de MicrosoftBrowse a Model Using the Microsoft Generic Content Tree Viewer,» Docs.microsft.com, 30 Abril 2018. [En línea]. Available: https://docs.microsoft.com/es-es/sql/analysis-services/data-mining/mining-model-content-analysis-services-data-mining?view=sql-server-2017. [Último acceso: 10 Enero 2019].

[18] D. V. C. A. Yudi Bernal, «La información en gestión de proyectos,» Publicación de la Facultad de Ingenieria Tecnol.Investig.Academia TIA, vol. 4, nº 1, 2016.

[19] T. M. H. d. l. S. O. L. M. C. E. A. M. J. Moisés García, «Aplicación del proceso unificado en el desarrollo de un software que estima el inventario y el crecimiento-rendimiento maderable en plantaciones de eucalipto,» Scielo, vol. 23, pp. 163-178, 2017.

[20] R. Software, «Rational Unified Process Best Practices for Software Development Teams,» White Paper TP026B, 2011.

[21] L. Alegsa, «Información sobre aplicación web,» [En línea]. Available: http://www.alegsa.com.ar/Dic/aplicacion%20web.php. [Último acceso: 7 Febrero 2017].

[22] L. Villoria, APLICACIONES WEB 2.0 - Google docs, Eduvim - Editorial Universitaria Villa Mar, 2009.

[23] K. E. Kendall, Análisis y diseño de sistemas, 6ta ed., México: Pearson Educación, 2005.

[24] Á. D. Macías, Artist, Securización de REST. [Art]. Universidad Autonoma de Barcelona, 2016.

[25] A. Rodriguez, «RESTful Web services: The basics,» developerWorks, 2008.

[26] P. Deitel, C# 6 for Programmers, Boston: Prentice Hall, 2016.

[27] O. H. D. ERNSTSSON, Artist, Designing and implementing a web-based data warehouse solution for cost analysis. [Art]. Department of Computer Science and Engineering Division of Software Engineering and Technology CHALMERS UNIVERSITY OF TECHNOLOGY, 2010.

[28] M. E. G. Tuapante, Artist, Desarrollo de un sistema de control de procesos de entrada y salida de un producto mediante tecnología móvil en empresas comerciales. [Art]. 2011.

[29] G. Shepherd, Microsoft ASP.NET 4 Step by Step, Microsoft Press, 2010, p. 640.

[30] Newtonsoft, «Introduction,» Newtonsoft, [En línea]. Available: https://www.newtonsoft.com/json/help/html/Introduction.htm. [Último acceso: 15 Octubre 2018].

[31] J. V. G. C. Ruegg, «Math.NET Numerics,» Math.NET, [En línea]. Available: https://numerics.mathdotnet.com/. [Último acceso: 15 Octubre 2018].

[32] O. D. Ltd, «OpenCage Geocoder - Easy, Open, Worldwide, Affordable Geocoding,» Opencage Data Ltd, [En línea]. Available: https://opencagedata.com/api. [Último acceso: 15 Octubre 2018].

[33] A. Allan, «Thunderforest,» Gravitystorm Limited, [En línea]. Available: https://www.thunderforest.com/. [Último acceso: 15 Octubre 2018].

[34] «Tipos de datos (Transact-SQL),» Microsoft, 09 Diciembre 2017. [En línea]. Available: https://docs.microsoft.com/es-mx/sql/t-sql/data-types/data-types-transact-sql?view=sql-server-2017. [Último acceso: 15 Octubre 2018].

[35] M. Ray, «Cursor (Transact-SQL),» Microsoft, 22 Julio 2017. [En línea]. Available: https://docs.microsoft.com/es-mx/sql/t-sql/data-types/cursor-transact-sql?view=sql-server-2017. [Último acceso: 15 Octubre 2018].

[36] M. Ray, «Table (Transact-SQL),» Microsoft, 10 Octubre 2018. [En línea]. Available: https://docs.microsoft.com/es-mx/sql/t-sql/data-types/table-transact-sql?view=sql-server-2017. [Último acceso: 15 Octubre 2018].

[37] «Funciones de agregado (Transact-SQL),» Microsoft, 14 Agosto 2018. [En línea]. Available: https://docs.microsoft.com/es-es/sql/t-sql/functions/aggregate-functions-transact-sql?view=sql-server-2017. [Último acceso: 15 Octubre 2018].

[38] P. Centurión Garzón y J. Pardo Martínez, «Clave taxonómica de identificación virtual de la colección de insectos acuáticos del departamento de biología de la universidad pedagógica nacional,» Bogotá. Universidad Pedagógica Nacional, Bogotá, 2013.

[39] Pl@ntNet, «Pl@ntNet needs you! - Pl@ntNet,» [En línea]. Available: https://plantnet.org/en/donations/. [Último acceso: 2 1 2018].

[40] A. G. V. Sania Ortega Andrade, Artist, Darwin Core: Estándar para la Gestión de Datos Biológicos Primarios en la UTN. [Art]. Universidad Técnica del Norte, Facultad de Ingeniería en Ciencias Agropecuarias y Ambientales, 2017.

[41] M. Hidalgo, Diseño de un Sistema de Información para el control del Patrimonio Predial, Mario Meza Hidalgo.

CAPITULO VII

ANEXOS

7.1. Anexo 1: Matriz del Problema

Ilustración 63. Matriz de relación problema-objetivos

Problemas	Objetivos	Resultados Esperados
¿Es posible gestionar (recolectar, recuperar y procesar) los datos entomológicos recolectados por los investigadores de la UTEQ?	Desarrollar una aplicación Web y Móvil que permita la gestión automatizada de datos entomológicos recolectados por la UTEQ para análisis estadístico.	Los investigadores tendrán una herramienta que permita la gestión y consolidación de datos para análisis estadísticos.
¿Qué diseño físico de base de datos se acopla de manera adecuada a la información requerida para estudios entomológicos?	Crear el modelo físico que soporte las variables requeridas en investigaciones entomológicas en estudios transversales y longitudinales.	Garantizar que los datos de los registros de insectos perduren con el transcurso del tiempo.
¿Se puede disponer de datos de investigaciones entomológicas realizadas anteriormente por investigadores de la UTEQ?	Disponer de datos de investigaciones entomológicas realizadas mediante la utilización del sistema web	Sitio web donde contendrá las investigaciones entomológicas realizadas por la UTEQ.
¿La implementación de un servicio de análisis (modelos, formulas estadísticas) mejorará la obtención de resultado y conclusiones de los estudios?	Proveer de servicios de análisis estadísticos aplicables a bases de datos entomológicas.	Herramienta de análisis que disponga de un conjunto de diferentes técnicas de análisis estadístico y de minería, aplicable a base de datos entomológicas.

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

7.2. Anexo 2: Modelo de datos

7.2.1. Diccionario de datos

Tabla 32. Lista de tablas de base de datos PIentomología.

Servidor: SQL SERVER Base de datos: BDPIentomología		
Tabla	Esquema	Descripción
tbrol	dbo	Tabla de roles de los usuarios (administrador, investigador).
tbusuario	dbo	Tabla de usuario para el acceso al sistema.
tbproyectoinv	dbo	Tabla de los proyectos de los investigadores.
tbobservacion	dbo	Tabla de las observaciones de los investigadores.
tbvariable	dbo	Tabla de variables que puede tener el proyecto.
tbdatosvariable	dbo	Tabla datos variable almacena datos y variable a la que corresponde el dato.
tbtipovariable	dbo	Tabla tipo variable: decimal, cadena, imagen, numérica, fecha.
tbopcionvariable	dbo	Tabla opción variable, opciones de selección que puede tener la variable.
tbtipofuente	dbo	Tabla de los tipos fuente: manual, móvil.
tbsumarizacion	dbo	Tabla de las sumarización del proyecto
tbconfiguracionsumar	dbo	Tabla configuración de sumarización almacena la

		configuración como por ejemplo totaliza semana por nombre especie.
tbtiposumarizacion	dbo	Tabla de los tipos de sumarización: suma, promedio, varianza, desviación estadística estándar, desviación típica estadística, mínimo, máximo, varianza estadística.
tbtipousovariable	dbo	Tabla tipo de uso variable: Agrupación, valor Totalizar.
tbmodelomineria	dbo	Tabla modelo minería del proyecto.
tbtecnicamineriadatos	dbo	Tabla técnica del modelo: regresión Lineal, árboles de decisión, Naive Bayes, Clústeres.
tbcorrelacionbivariada	dbo	Tabla correlación bivariada del proyecto.
tbconfiguracioncorrelacion	dbo	Tabla configuración correlación almacena variable correlacionar.
tbtipocorrelacion	dbo	Tabla tipo correlación: Pearson, Spearman

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN
ELABORADO POR: AUTOR

Tabla 33. Diccionario de datos: tbrol

Entidad: tbrol			
Columna	Tipo	Nulo	Descripción
idrol	int	No	Clave primaria
nombrerol	varchar(20)	No	
descripcion	varchar(50)	Si	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

Tabla 34. Diccionario de datos: tbusuario

Entidad: tbusuario			
Columna	Tipo	Nulo	Descripción
idusuario	int	No	Clave primaria
usuario	varchar(40)	No	
clave	varchar(40)	No	
correo	varchar(50)	No	
telefono	varchar(10)	Si	
nombres	varchar(25)	Si	
apellidos	varchar(25)	Si	
idrol	int	No	Clave foránea

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 35. Diccionario de datos: tbproyectoinv.

Entidad: tbproyectoinv			
Columna	Tipo	Nulo	Descripción
idproyectoinv	int	No	Clave primaria
nombreproyecto	varchar(100)	No	
responsable	varchar(40)	No	
idusuario	int	No	Clave foránea
estado	varchar(40)	Si	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 36. Diccionario de datos: tbobservacion.

Entidad: tbobservacion			
Columna	Tipo	Nulo	Descripción
idobservacion	int	No	Clave primaria
fechaobservacion	date	No	
decimallatitud	decimal(12, 9)	SI	
decimallongitud	decimal(12, 9)	SI	
zona	varchar(50)	No	

idproyectoinv	int	No	Clave foránea
idusuario	int	No	Clave foránea
georeferenciamovil	bit	SI	
pais	varchar(30)	SI	
provincia	varchar(30)	SI	
ciudad	varchar(30)	SI	
localidad	varchar(100)	SI	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 37. Diccionario de datos: tbdatosvariable.

Entidad: tbdatosvariable			
Columna	Tipo	Nulo	Descripción
iddatosvariable	int	No	Clave primaria
valor	varchar(250)	Si	Varchar(250) porque existe la posibilidad de guardar un dirección de URL imagen extensa.
fechaingreso	date	No	
idvariable	int	No	Clave foránea
idobservacion	int	No	Clave foránea

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 38. Diccionario de datos: tbvariable.

Entidad: tbvariable			
Columna	Tipo	Nulo	Descripción
idvariable	int	No	Clave primaria
nombrevariable	varchar(40)	No	
idtipovariable	int	No	Clave foránea
idtipofuente	int	No	Clave foránea
idproyectoinv	int	No	Clave foránea

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 39. Diccionario de datos: tbtipovariable.

Entidad: tbtipovariable			
Columna	Tipo	Nulo	Descripción
idtipovariable	int	No	Clave primaria
tipo	varchar(20)	No	
descripcion	varchar(50)	No	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 40. Diccionario de datos: tbopcionvariable.

Entidad: tbopcionvariable			
Columna	Tipo	Nulo	Descripción
idopcionvariable	int	No	Clave primaria
opcion	varchar(50)	No	
descripcion	varchar(40)	No	
idvariable	int	No	Clave foránea

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 41. Diccionario de datos: tbtipofuente.

Entidad: tbtipofuente			
Columna	Tipo	Nulo	Descripción
idtipofuente	int	No	Clave primaria
tipo	varchar(20)	No	
descripcion	varchar(50)	Si	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 42. Diccionario de datos: tbsumarizacion.

Entidad: tbsumarizacion			
Columna	Tipo	Nulo	Descripción
idsumarizacion	int	No	Clave primaria
nombresumarizacion	varchar(70)	No	
descripcion	varchar(50)	Si	

estado	varchar(15)	No	
idtiposumar	int	No	Clave foránea
idproyectoinv	int	No	Clave foránea

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 43. Diccionario de datos: tbconfiguracionsumar.

Entidad: tbconfiguracionsumar			
Columna	Tipo	Nulo	Descripción
idconfiguracionsumar	int	No	Clave primaria
idsumarizacion	int	No	Clave foránea
idvariable	int	No	Clave foránea
idtipousovar	int	No	Clave foránea
descripcion	varchar(50)	Si	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 44. Diccionario de datos: tbtiposumarizacion.

Entidad: tbtiposumarizacion			
Columna	Tipo	Nulo	Descripción
idtiposumarizacion	int	No	Clave primaria
nombretiposumarizacion	varchar(50)	No	
descripcion	varchar(15)	No	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 45. Diccionario de datos: tbtipousovariable.

Entidad: tbtipousovariable			
Columna	**Tipo**	**Nulo**	**Descripción**
idtipousovariable	int	No	Clave primaria
uso	varchar(15)	No	
descripcion	varchar(50)	Si	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 46. Diccionario de datos: tbmodelomineria.

Entidad: tbmodelomineria			
Columna	Tipo	Nulo	Descripción
idmodelomineria	int	No	Clave primaria
nombremodelomineria	varchar(80)	No	
estado	varchar(15)	No	
porcentajedatosprueba	int	No	
nummaxcasosconjdatosprueb	int	No	
descripcion	varchar(50)	Si	
idtecnicamineriadatos	int	No	Clave foránea
idproyectoinv	int	Si	Clave foránea

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 47. Diccionario de datos: tbtecnicamineriadatos.

Entidad: tbtecnicamineriadatos			
Columna	Tipo	Nulo	Descripción
idtecnicamineriadatos	int	No	Clave primaria
nombretecnicamineria	varchar(30)	No	
nombretecnico	varchar(30)	No	
descripcion	varchar(50)	Si	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 48. Diccionario de datos: tbcorrelacionbivariada.

Entidad: tbcorrelacionbivariada			
Columna	Tipo	Nulo	Descripción
idcorrelacion	int	Np	Clave primaria
nombrecorrelacion	varchar(70)	Si	
descripcion	varchar(70)	Si	
estado	varchar(15)	No	
idtipocorrelacion	int	No	Clave foránea
idproyectoinv	int	No	Clave foránea

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

Tabla 49. Diccionario de datos: tbconfiguracioncorrelacion.

Entidad: tbconfiguracioncorrelacion			
Columna	Tipo	Nulo	Descripción
idconfiguracioncorrelacion	int	No	Clave primaria
idcorrelacion	int	No	Clave foránea
idvariable	int	No	Clave foránea
descripcion	varchar(50)	Si	

FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

Tabla 50. Diccionario de datos: tbtipocorrelacion.

Entidad: tbtipocorrelacion			
Columna	Tipo	Nulo	Descripción
idtipocorrelacion	int	No	Clave primaria
nombretipocorrelacion	varchar(15)	No	
interpretacion	varchar(250)	Si	
descripcion	varchar(250)	Si	

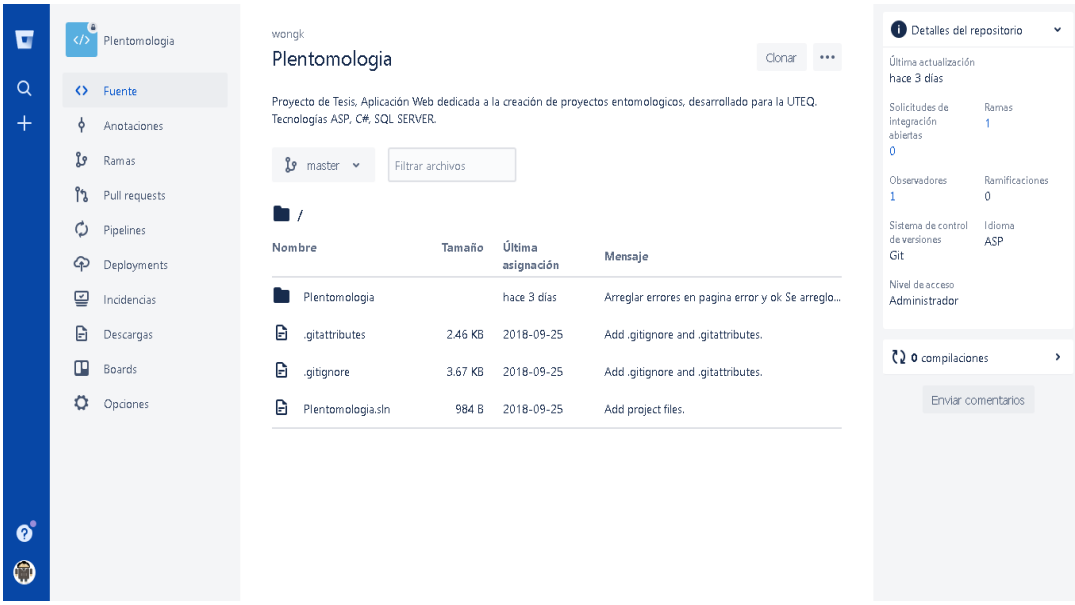
FUENTE: ANÁLISIS DEL PROYECTO DE INVESTIGACIÓN

ELABORADO POR: AUTOR

7.3. Anexo 3: Control de versiones

7.3.1. Control de versiones bitbucket

Ilustración 64.Repositorio bitbucket, para control de versiones de sistema PIentomología.



FUENTE: https://bitbucket.org/wongkevin/pruebafinalwong/src/master/

ELABORADO POR: AUTOR

7.4. Anexo 4: Registros de investigaciones entomológicas

7.4.1. Diferentes registros de datos en investigaciones entomológicas realizadas en la UTEQ

Investigación Fluctuación poblacional de premnobius cavipennis eichhoff (coleoptera; curculionidae; scolytinae) en plantaciones de balsa [ochroma pyramidale (cav. ex. lam) urb] en la zona centro del litoral ecuatoriano.

Tabla 51. Registro de datos de investigación fluctuación.

Fecha	Semana	Bloque	Altura	Trampa	*P. cavipemnis*
23/9/2014	1	1	1	1	0
	1	1	1	2	0
	1	1	1	3	0
	1	1	4	1	0
	1	1	4	2	0
	1	1	4	3	0
	1	1	8	1	0
	1	1	8	2	0
	1	1	8	3	0
	1	2	1	1	0
	1	2	1	2	0
	1	2	1	3	0
	1	2	4	1	0
	1	2	4	2	0
	1	2	4	3	0
	1	2	8	1	0
	1	2	8	2	0
	1	2	8	3	0

FUENTE: UTEQ

ELABORADO: JOSELYN AVEROS

Investigación diferencias en la diversidad de insectos coleóptera: curculionidae, scolytinae en la formación primaria y secundaria del bosque protector murocomba, cantón Valencia, provincia Los Ríos, año 2015.

Tabla 52. Registro de datos de investigación diversidad.

GENEROS INSECTOS	TRAMPA 1					
SCOLYTINAE	12 Junio	22 Junio	02 Julio	12 Julio	22 Julio	01 Agosto
Bostrichidae	0	0	0	0	0	0
Coptoborus spp.	0	0	0	0	0	0
Corthylus 1	4	0	5	2	0	1
Corthylus 2	0	1	3	0	0	0
Corthylus 3	0	0	0	0	0	0
Cryptocarenus heveae	0	0	0	0	0	0
Microcorthylus	0	0	0	0	0	0
Platypodidae	0	0	0	0	0	0
Premnobius cavipenis	0	0	0	0	0	0
Tricolus	0	0	0	0	0	0
X. bispinatulus	0	0	0	0	0	0
Xyleborinus bicornatulus	0	1	1	0	0	0
Xyleborus (g)	0	0	0	0	0	0
Xyleborus 1	0	0	0	0	0	0
Xyleborus affinis	1	2	2	3	0	0
Xyleborus ferrugenius	0	0	1	0	0	0
Xyleborus sp.	0	0	1	0	0	0
Xyleborus volvulos	0	0	0	0	0	0
Xylosandrus morigerus	0	0	1	1	0	0

FUENTE: UTEQ

ELABORADO: LEVIN ZAMBRANO

Investigación diversidad de insectos (coleoptera: curculionidae: scolytinae) en ecosistemas forestales.

Tabla 53. Registros de datos de investigación diversidad scolytinae.

Tribus	Especie	Balsa	Teca	Caucho	Melina
Xyleborini	C.ochromactonus	1	0	0	0
Xyleborini	X. affinis	1	0	1	1
Xyleborini	X. spinulosus	1	1	1	0
Xyleborini	X. ferrugineus	0	1	1	1
Xyleborini	Xyleborinus bicornatulus	1	1	1	1
Xyleborini	Xylosandrus morigerus	1	1	1	1
Ipini	Premnobius cavipennis	1	1	1	0
Cryphalini	Cryptocarenus heveae	1	1	1	1
Corthylini	Corthylus sp	1	1	1	1
Corthylini		1	1	0	1
Corthylini	Theoborus sp	1	0	0	0

FUENTE: UTEQ

ELABORADO: GABRIEL IBARRA

7.5. Anexo 5: Códigos fuente PIentomología

Ilustración 65. Código fuente creación de objetos de SSAS.

```
using Microsoft.AnalysisServices;
using System;
using System.Collections.Generic;
using System.Data.OleDb;
using System.Linq;
using System.Web;

namespace PIentomologia.Models.MineriaDatos
{
    /// <summary> LAS GESTION DEL SERVIDOR Y LOS OBJETOS SSAS SE REALIZARA MEDIANTE LA .DLL Microsoft.AnalysisServices QUE G ...
    5 references | kevin, 16 days ago | 1 author, 4 changes
    public class ProveedorDatosAMO
    {
        propiedades
        1 reference | kevin, 20 days ago | 1 author, 1 change
        public ProveedorDatosAMO() ...
        1 reference | kevin, 20 days ago | 1 author, 1 change
        public ProveedorDatosAMO(String nombreServidorSSAS)...
        #region Base de Datos SSAS
        3 references | kevin, 19 days ago | 1 author, 2 changes
        public bool Conectar()...
        1 reference | kevin, 20 days ago | 1 author, 1 change
        public void Desconectar()...
        1 reference | kevin, 20 days ago | 1 author, 1 change
        public Boolean crearBD(String nombreBDSSAS)
        {
            try{
                baseDatos = new Database();
                baseDatos.Name = BASEDATOS + nombreBDSSAS;
                baseDatos.ID = BASEDATOS + nombreBDSSAS;
                //Agregar al Servidor
                servidor.Databases.Add(baseDatos);
                baseDatos.Update();
                return true;
            }catch{
                return false;
                throw;
            }
        }
        /// <summary> Busca mediante el nombre una base da datos SSAS
        2 references | kevin, 20 days ago | 1 author, 1 change
        public Boolean buscarBDxNombre(String nombreBDSSAS)...
        1 reference | kevin, 20 days ago | 1 author, 1 change
        public Boolean removerBD(String nombreBDSSAS)...
        #endregion
        Data Source
        Data Source View
        Mining Structure
        Mining Model
        Procesar Modelo mineria

    }
}
```

Ilustración 66. Script SQL Server organizar datos a tablas.

```
ALTER PROCEDURE [dbo].[pcdobtenerdatosproyectoinv]
@idproyecto INT
AS
BEGIN
/*PARAMETROS*/...
DECLARE @idobservacion INT
DECLARE @valor VARCHAR(MAX)
DECLARE @nombrevariable VARCHAR(40)

DECLARE @contadorIdoObservacion INT
DECLARE @contadorDatosVariable INT
SET @contadorIdoObservacion = 0

DECLARE @informacionDatoVariable VARCHAR(MAX)
DECLARE @informacionTabla VARCHAR(MAX)

DECLARE cursorIdobservacion CURSOR SCROLL FOR
(SELECT tbdatosvariable.idobservacion FROM tbobservacion...

OPEN cursorIdobservacion

FETCH FIRST FROM cursorIdobservacion INTO @idobservacion
WHILE @@FETCH_STATUS = 0
BEGIN
    DECLARE cursorDatos CURSOR SCROLL FOR
    (SELECT tbvariable.nombrevariable, tbdatosvariable.valor...

    SET @informacionDatoVariable =''
    SET @contadorDatosVariable =0

    OPEN cursorDatos
    FETCH FIRST FROM cursorDatos INTO @nombrevariable,@valor
    WHILE @@FETCH_STATUS = 0
    BEGIN
        SET @contadorDatosVariable += 1
        IF @contadorDatosVariable = 1
            SET @informacionDatoVariable = @informacionDatoVariable + 'SELECT OBSERVACION='
            +CONVERT(varchar,@idobservacion)+','
            +@nombrevariable+'='''+CONVERT(varchar(MAX),@valor)+''''
        ELSE
            SET @informacionDatoVariable = @informacionDatoVariable + ','
            +@nombrevariable+'='''+CONVERT(varchar(MAX),@valor)+''''

        FETCH NEXT FROM cursorDatos INTO @nombrevariable,@valor
    END
    CLOSE cursorDatos

    DEALLOCATE cursorDatos

    SET @contadorIdoObservacion +=1
    IF @contadorIdoObservacion =1
        SET @informacionTabla = @informacionDatoVariable
    ELSE
        SET @informacionTabla = @informacionTabla + ' UNION ' + @informacionDatoVariable
    FETCH NEXT FROM cursorIdobservacion INTO @idobservacion
END
EXEC(@informacionTabla)
CLOSE cursorIdobservacion
DEALLOCATE cursorIdobservacion
END
GO
```

Ilustración 67. Código fuente de clase POCO.

```
namespace PIentomologia.Models
{
    using ...

    [Table("tbproyectoinv")]
    47 references | kevin, 42 days ago | 1 author, 1 change
    public partial class tbproyectoinv
    {
        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
        3 references | kevin, 42 days ago | 1 author, 1 change
        public tbproyectoinv()
        {
            tbcorrelacionbivariada = new HashSet<tbcorrelacionbivariada>();
            tbmodelomineria = new HashSet<tbmodelomineria>();
            tbobservacion = new HashSet<tbobservacion>();
            tbsumarizacion = new HashSet<tbsumarizacion>();
            tbvariable = new HashSet<tbvariable>();
        }

        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        7 references | kevin, 42 days ago | 1 author, 1 change
        public int idproyectoinv { get; set; }

        [Required]
        [StringLength(100)]
        6 references | kevin, 42 days ago | 1 author, 1 change
        public string nombreproyecto { get; set; }

        [Required]
        [StringLength(40)]
        3 references | kevin, 42 days ago | 1 author, 1 change
        public string responsable { get; set; }

        3 references | kevin, 42 days ago | 1 author, 1 change
        public int idusuario { get; set; }

        [StringLength(40)]
        3 references | kevin, 42 days ago | 1 author, 1 change
        public string estado { get; set; }

        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]

        2 references | kevin, 42 days ago | 1 author, 1 change
        public virtual ICollection<tbcorrelacionbivariada> tbcorrelacionbivariada { get; set; }

        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
        1 reference | kevin, 42 days ago | 1 author, 1 change
        public virtual ICollection<tbmodelomineria> tbmodelomineria { get; set; }

        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
        2 references | kevin, 42 days ago | 1 author, 1 change
        public virtual ICollection<tbobservacion> tbobservacion { get; set; }

        1 reference | kevin, 42 days ago | 1 author, 1 change
        public virtual tbusuario tbusuario { get; set; }

        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
        2 references | kevin, 42 days ago | 1 author, 1 change
        public virtual ICollection<tbsumarizacion> tbsumarizacion { get; set; }

        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
        14 references | kevin, 42 days ago | 1 author, 1 change
        public virtual ICollection<tbvariable> tbvariable { get; set; }
    }
}
```

Ilustración 68. Código fuente de Clase DAO.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace PIentomologia.Models
{
    8 references | kevin, 41 days ago | 1 author, 2 changes
    public class DAOProyectoInv
    {
        private BaseDatosContexto accesoDatos;
        5 references | kevin, 42 days ago | 1 author, 1 change
        public DAOProyectoInv()
        {
            accesoDatos = new BaseDatosContexto();
        }
        1 reference | kevin, 42 days ago | 1 author, 1 change
        public List<tbproyectoinv> obtenerListaProyectosInvXUsuario(int idUsuario)
        {
            List<tbproyectoinv> listaProyectos = null;
            try{
                listaProyectos = accesoDatos.tbproyectoinv.Where(p => p.idusuario == idUsuario).ToList<tbproyectoinv>();
                return listaProyectos;
            }catch (InvalidOperationException detalleError){
                return listaProyectos;
                throw;
            }
        }

        2 references | kevin, 42 days ago | 1 author, 1 change
        public tbproyectoinv buscarxId(int idProyecto) ...

        1 reference | kevin, 41 days ago | 1 author, 2 changes
        public Boolean eliminar(tbproyectoinv proyecto) ...

        2 references | kevin, 42 days ago | 1 author, 1 change
        public int obtenerUltimoRegistro() {
            try {
                int ultimoRegistro = accesoDatos.tbproyectoinv
                    .OrderByDescending(p => p.idproyectoinv)
                    .FirstOrDefault().idproyectoinv;
                return ultimoRegistro;
            } catch(Exception detalleError){ return -1; throw; }
        }
        2 references | kevin, 42 days ago | 1 author, 1 change
        public Boolean guardar(tbproyectoinv nuevoProyecto) ...
    }
}
```

Ilustración 69. Código fuente clase Controlador.

```
using PIentomologia.Models;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace PIentomologia.Controller
{
    16 references | kevin, 42 days ago | 1 author, 1 change
    public class ProyectoInvController
    {
        14 references | kevin, 42 days ago | 1 author, 1 change
        public ProyectoInvController() {

        }

        9 references | kevin, 42 days ago | 1 author, 1 change
        public tbproyectoinv buscarProyectoXid(int idProyectoInv) {
            return new DAOProyectoInv().buscarxId(idProyectoInv);
        }

        3 references | kevin, 42 days ago | 1 author, 1 change
        public List<tbproyectoinv> listaProyectoInvxUsu(int idUsuario) {
            return new DAOProyectoInv().obtenerListaProyectosInvXUsuario(idUsuario);
        }

        1 reference | kevin, 42 days ago | 1 author, 1 change
        public Boolean eliminar(int idProyectoInv) ...

        0 references | kevin, 42 days ago | 1 author, 1 change
        public Boolean guardar(int idUsuario,string nombreProyecto, string responsable,string estado) {
            tbproyectoinv proyecto = new tbproyectoinv();
            DAOProyectoInv accesoDatos = new DAOProyectoInv();
            proyecto.idproyectoinv = accesoDatos.obtenerUltimoRegistro()+1;
            proyecto.idusuario = idUsuario;
            proyecto.nombreproyecto = nombreProyecto;
            proyecto.responsable = responsable;
            proyecto.estado = estado;
            return accesoDatos.guardar(proyecto);
        }

        1 reference | kevin, 42 days ago | 1 author, 1 change
        public int guardarId(int idUsuario, string nombreProyecto,
                             string responsable, string estado)...

    }
}
```

Ilustración 70. Código fuente página ASP.NET.

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="Gestionar.aspx.cs" Inherits="PIentomologia.ProyectoInv.Gestionar" %>
<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">
    <div class="row" style="padding:10px">
        <h3>Lista de Proyectos Investigación</h3>
        <hr/>
        <div class="table-responsive">
          <asp:GridView ID="Proyectos" DataKeyNames="idproyectoinv"
              runat="server" GridLines="Vertical"
              AutoGenerateColumns="false" CssClass="table table-bordered"
              ItemType="PIentomologia.Models.tbproyectoinv"
              SelectMethod="listaProyectos">
              <Columns>
                  <asp:BoundField DataField="idproyectoinv" HeaderText="ID" SortExpression="ID" HeaderStyle-BackColor="LightBlue"/>
                  <asp:BoundField DataField="nombreproyecto" HeaderText="NOMBRE" HeaderStyle-BackColor="LightBlue"/>
                  <asp:BoundField DataField="responsable" headerText="RESPONSABLE" HeaderStyle-BackColor="LightBlue"/>
                  <asp:BoundField DataField="estado" HeaderText="ESTADO" HeaderStyle-BackColor="LightBlue"/>
                  <asp:TemplateField HeaderText="OPCIONES" HeaderStyle-BackColor="LightBlue" >
                      <ItemTemplate>
                          <asp:LinkButton ID="lnkBtnActualizar" Text="Actualizar" runat="server" OnClick="onclickActualizar"/>
                          |
                          <asp:LinkButton ID="lnkBtnEliminar" Text="Eliminar" runat="server"
                              OnClientClick="return confirm('Seguro de Eliminar Proyecto')" OnClick="onclikEliminar"/>
                          |
                          <asp:LinkButton ID="lnkBtnAgregarVariable" Text="Agregrar Variables" runat="server" OnClick="onclickAgregarVariable" />

                          <asp:HiddenField ID="hiddenFielProyectoid" runat="server" Value='<%# Eval("idproyectoinv") %>' />
                      </ItemTemplate>
                  </asp:TemplateField>
              </Columns>
          </asp:GridView>
          </div>
          <asp:PlaceHolder runat="server" ID="Mensaje" Visible="false">
              <p class="text-danger">
                  <asp:Literal  runat="server" ID="textoMensaje"/>
              </p>
          </asp:PlaceHolder>
          <div class="form-group">
                  <asp:Button runat="server" OnClick="onclickCrearProyecto"
                   Text="Nuevo Proyecto" CssClass="btn btn-primary first-in-line" />
          </div>
    </div>
</asp:Content>
```

Ilustración 71. Código para ingresar datos de observaciones.

```
try
{
    List<Observacion> listaObservaciones = new List<Observacion>();
    using (System.IO.TextReader lector = System.IO.File
                .OpenText(@"C:/Users/Kevin Onofre/Documents/TESIS/Tesis/Datos/DATOS_VERGEL.csv"))
    {
        CsvHelper.CsvReader archivoCsv = new CsvHelper.CsvReader(lector);
        archivoCsv.Configuration.Delimiter = ";";
        archivoCsv.Configuration.MissingFieldFound = null;
        while (archivoCsv.Read()){
            Observacion observacion = archivoCsv.GetRecord<Observacion>();
            listaObservaciones.Add(observacion);
        }
        int cantidadObjetos = listaObservaciones.Count;
        //Organizar los datos a guardar
        for (int contadorObservacion = 0; contadorObservacion < listaObservaciones.Count; contadorObservacion++)
        {
            //Crear Observacion
            tbobservacion observacionAGuardar = new tbobservacion();
            observacionAGuardar.idusuario = 6;
            observacionAGuardar.idproyectoinv = 200;
            observacionAGuardar.fechaobservacion = Convert.ToDateTime(listaObservaciones[contadorObservacion].fecha);
            observacionAGuardar.zona = listaObservaciones[contadorObservacion].zona;
            observacionAGuardar.georeferenciamovil = listaObservaciones[contadorObservacion].georeferencia;
            //Guardar observacion
            int idObservacion = new ObservacionController().guardarId(observacionAGuardar.fechaobservacion
                                                                      , observacionAGuardar.zona
                                                                      , observacionAGuardar.idproyectoinv
                                                                      , observacionAGuardar.idusuario
                                                                      , false);
            List<tbdatosvariable> listaMasCampos = new List<tbdatosvariable>();
            //Crar campos adicionales
            //fecha
            tbdatosvariable datosVariableFecha = new Models.tbdatosvariable();
            datosVariableFecha.valor =
                Convert.ToDateTime(listaObservaciones[contadorObservacion].fecha).ToString("yyyy-MM-dd");
            datosVariableFecha.fechaingreso = Convert.ToDateTime("2015-04-25");
            datosVariableFecha.idvariable = 111;
            datosVariableFecha.idobservacion = idObservacion;

            datosVariableTrampa.idobservacion = idObservacion;

            //cantidad
            tbdatosvariable datosVariableCantidad = new Models.tbdatosvariable();
            datosVariableCantidad.valor = listaObservaciones[contadorObservacion].cantidad.ToString();
            datosVariableCantidad.fechaingreso = Convert.ToDateTime("2015-04-25");
            datosVariableCantidad.idvariable = 107;
            datosVariableCantidad.idobservacion = idObservacion;

            //nombreespecie
            tbdatosvariable datosVariableNombre = new Models.tbdatosvariable();
            datosVariableNombre.valor = listaObservaciones[contadorObservacion].nombreespecie;
            datosVariableNombre.fechaingreso = Convert.ToDateTime("2015-04-25");
            datosVariableNombre.idvariable = 99;
            datosVariableNombre.idobservacion = idObservacion;

            listaMasCampos.Add(datosVariableFecha);
            listaMasCampos.Add(datosVariableSemana);
            listaMasCampos.Add(datosVariableBloque);
            listaMasCampos.Add(datosVariableAltura);
            listaMasCampos.Add(datosVariableTrampa);
            listaMasCampos.Add(datosVariableCantidad);
            listaMasCampos.Add(datosVariableNombre);
            int contadorRegistrosIngresados = 0, contadorRegistrosNoIngresados = 0;
            ////Guardar mas campos
            foreach (Models.tbdatosvariable itemDatosVariable in listaMasCampos)
            {
                if (new DatosVariableController().guardar(itemDatosVariable.valor
                                                          , itemDatosVariable.fechaingreso
                                                          , itemDatosVariable.idvariable
                                                          , itemDatosVariable.idobservacion))
                    contadorRegistrosIngresados += 1;
                else
                    contadorRegistrosIngresados += 1;
            }
        }
    }
}
catch (Exception detalleError){
    string error = detalleError.ToString();
}
```

7.6. Anexo 6: Documentación PIentomología

7.6.1. Manuales de Usuario

Introducción a PIentomología

En la UTEQ se han desarrollado diversos estudios de investigación sobre la diversidad y caracterización de especies biológicas de distintos grupos taxonómicos entre ellos de insectos en diferentes ecosistemas forestales. La importancia de estos estudios es motivada por el conocimiento de que los insectos pueden llegar a atacar plantaciones forestales enteras provocando daños económicos. También se conoce que existen dificultades en el almacenamiento, procesamiento y obtención de resultados de esta información.

PIentomología tiene como principal obetivo gestionar los datos entomológicos recolectados por la UTEQ para análisis estadístico. El sistema se divide en dos herramientas móvil y web que permiten la recolección de datos. Mediante la aplicación móvil se captarán datos como la ubicación geográfica, fotografías y la información que se establesca en al crear el proyecto de estudio. Y luego ya en los laboratorios se realizará el análisis taxonómico correspondiente a la muestra para poder completar la información que será almacenada en la aplicación web. La aplicación web cuenta con diversas caracteristicas como son la creación de proyectos de estudio, el registro de observaciones y el análisis del proyecto de estudio donde se tienen tecnicas como la correlación, sumarización.

Aplicación Web: Manual de usuario para crear y configurar proyecto por parte del investigador

Tipo de usuario: Investigador. **Función:** Crear proyecto y configurar las variables del proyecto.

Debe ingresar al sistema mendiante el navegador web, de preferencia Google Chrome o Firefox. Se muestra la página de inicio de sesión, donde debera ingresar el usuario y la contraseña previamente proporcionadas por el administrador.

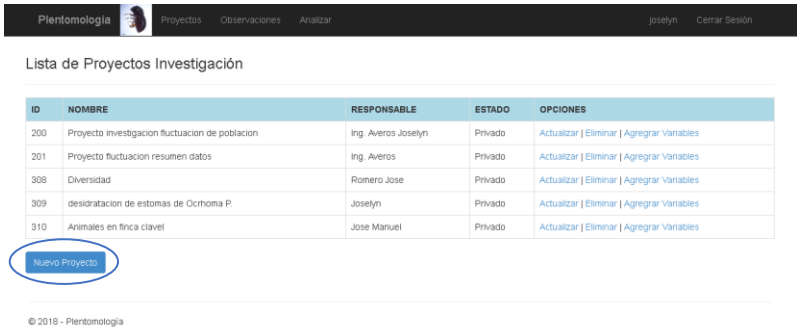
Ilustración 72. Página inicio de sesión.

Al ingresar correctamente los datos aparecera la página principal y un menú en la parte superior con la pestaña de Proyectos. Elegir esa opción.

Ilustración 73. Página principal.

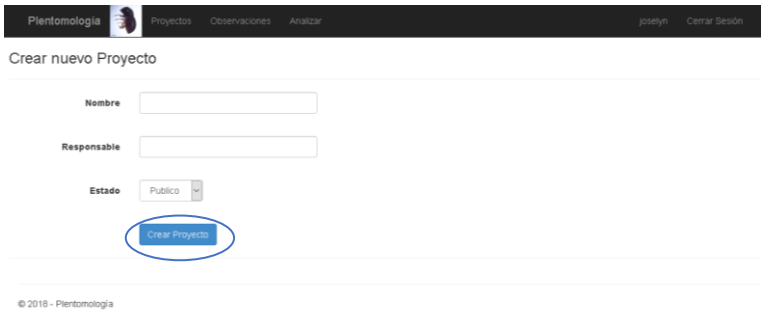
Al ingresar a la pestaña se presentará la lista con los diversos proyectos con los que cuenta el investigador, en caso de no tener ningun proyecto no se presentará nada, solo el botón de nuevo proyecto. Dar clic en nuevo proyecto.

Ilustración 74. Página lista de proyectos.



Al ingresar en nuevo proyecto se presentará un formulario para agregar los datos del proyecto nuevo. Ingresamos los datos solicitados. Luego damos clic en crear proyecto.

Ilustración 75. Página crear proyecto.



Al crearse el nuevo proyecto se prosigue con el ingreso de las variables que dispondrá el mismo. La página para agregar variables presenta un formulario donde encontramos parámetros como tipo de variable y tipo de fuente. Y las opciones de guardar, modificar, eliminar. A continuación se detalla el significado de los parámetros.

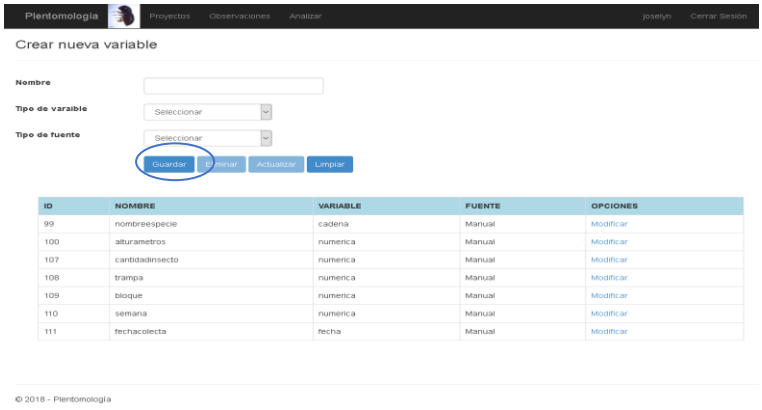
Tabla 54. Parámetros de variables.

Parámetro	Opciones soportadas
Tipo de variable: representa el tipo de contenido que se puede guardar en la variable.	Decimal: número que tiene un representación decimal finita en el sistema de numeración decimal. Cadena: texto de cadena de caracteres. Imagen: representación visual, de un objeto real.

	Numérica: número o cantidad sin coma decimal. Fecha: representa tiempo en día/mes/año
Tipo de fuente: representa el dispositivo con el que se captura el dato.	Manual: Dispositivo laptop. Móvil: Dispositivo celular.

Ingresamos los datos solicitados. Luego damos clic en Guardar. Al guardar se actualizará la lista con la nueva variable, que se encuentra en la parte de debajo de las opciones.

Ilustración 76. Página crear variable.



Como punto final puede seguir el mismo proceso antes mencionado para ingresar más variables o salir dando clic en la pestaña Proyectos.

Manual de usuario para registrar observación por parte del investigador

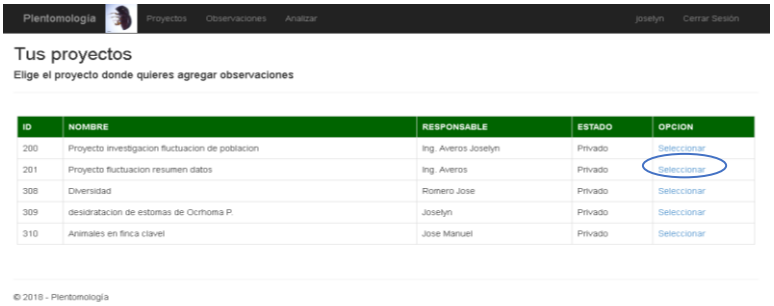
Tipo de usuario: Investigador. **Función:** Registrar observación en el proyecto.

Al ingresar correctamente los datos a la página de login, aparecera la página principal y un menú en la parte superior con la pestaña de Observaciones. Elegir esa opción.

Ilustración 77. Página principal opción observaciones.

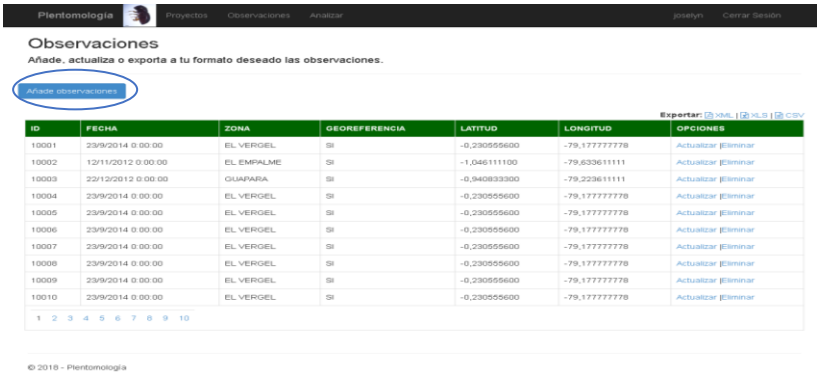
Al ingresar a la pestaña se presentará la lista con los diversos proyectos con los que cuenta el investigador, dar clic en el enlace del proyecto donde quiere ingresar observaciones.

Ilustración 78. Página lista de proyectos.



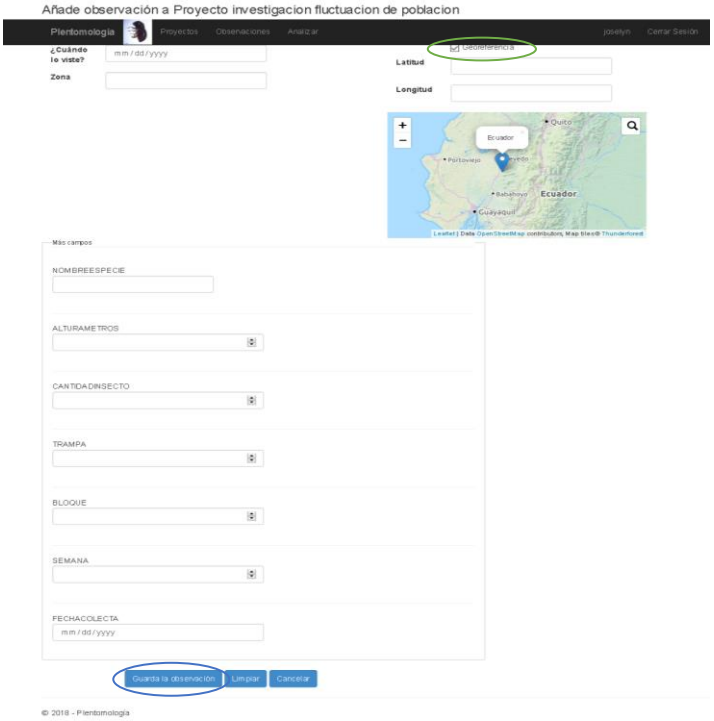
Aperece una pantalla donde se presenta la lista de registros de las observaciones del proyecto y el botón de añadir nueva observación. Dar clic en añadir nueva observación.

Ilustración 79. Página lista de observaciones.



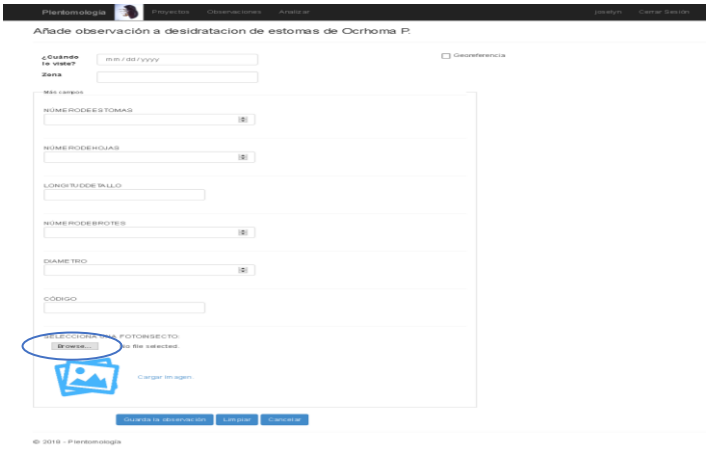
Al ingresar en añadir nueva observación se presentará un formulario para agregar los datos de la observación y una casilla de verificación para georeferenciar la observación (en color verde). Ingresamos los datos solicitados. Luego damos clic guardar observación.

Ilustración 80. Página agregar observaciones.



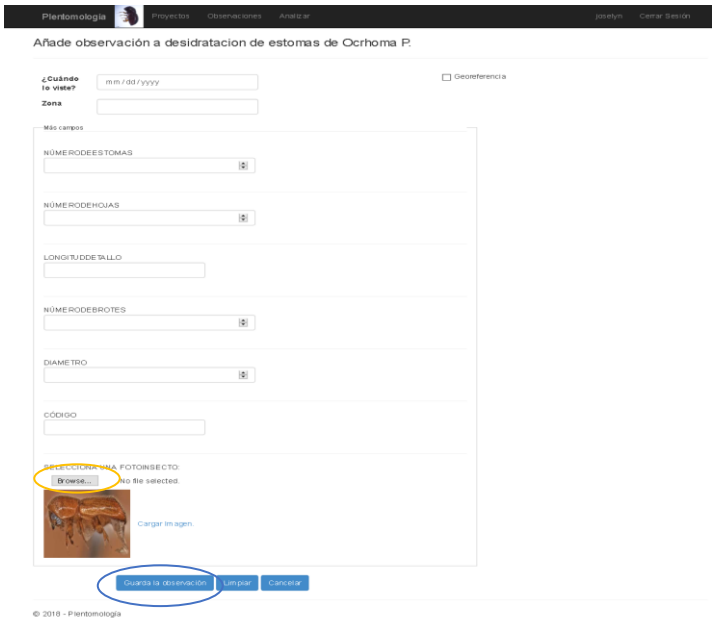
Las variables de cada proyecto pueden variar. Por lo tanto se presenta otra alternativa de variables, en la cual se tiene una variable de tipo imagen. En este caso para subir la imagen se debe dar clic en Browse, seleccionar el archivo que desea.

Ilustración 81. Página agregar observación con foto.



Dar clic en el enlace cargar imagen (en color amarillo), la imagen se presentará en pantalla si no existe problema. Luego damos clic guardar observación.

Ilustración 82. Página agregar observación, cargar foto.



Manual de usuario para realizar análisis de correlación por parte del investigador

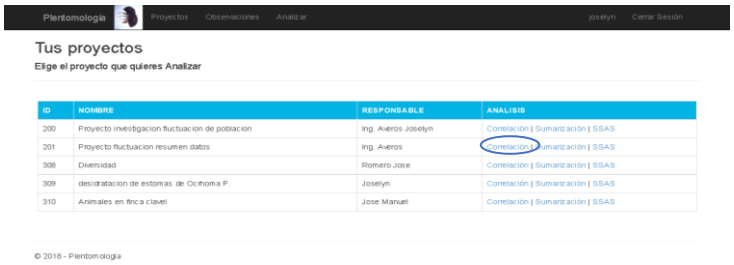
Tipo de usuario: Investigador. **Función:** Realizar análisis de correlación.

Al ingresar correctamente los datos a la página de login, aparecerá la página principal y un menú en la parte superior con la pestaña de Analizar. Elegir esa opción.

Ilustración 83. Página principal, opción analizar.

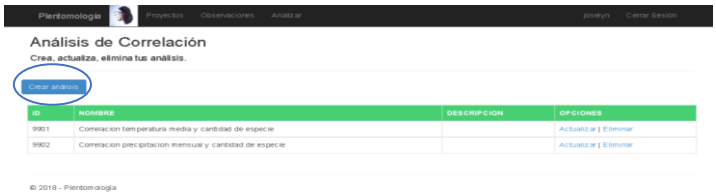
Al ingresar a la pestaña se presentará la lista con los diversos proyectos con los que cuenta el investigador, dar clic en el enlace del proyecto donde quiere ingresar nuevos análisis de correlación.

Ilustración 84. Página lista de proyecto, opción correlación.



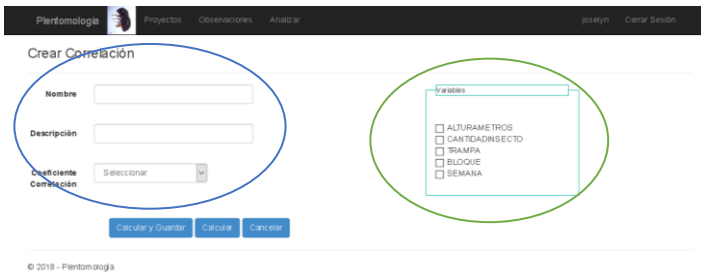
Aperece una pantalla donde se presenta la lista de análisis de correlación del proyecto y el botón de crear análisis. Dar clic en crear análisis.

Ilustración 85. Página crear lista de análisis de correlación.



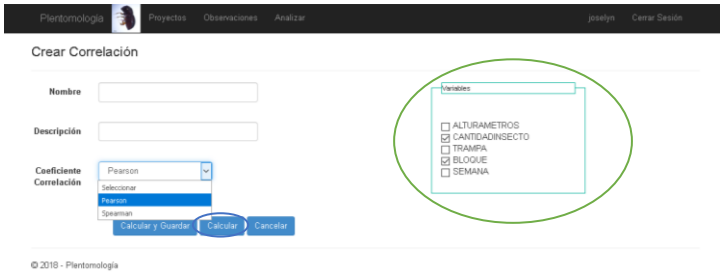
Al ingresar en crear análisis se presentará un formulario para agregar la configuración del análisis y un cuadro en la parte derecha, para seleccionar las variables que se utilizarán en el análisis (en color verde).

Ilustración 86. Página crear análisis de correlación.



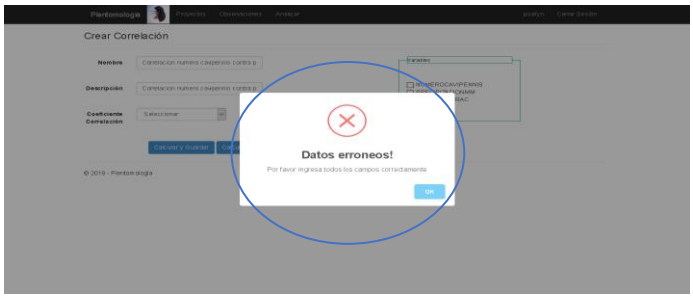
Ingresamos los datos solicitados y seleccionamos las variables a correlacionar (en color verde). Luego damos clic calcular (en color azul).

Ilustración 87. Página crear análisis de correlación.



En caso que los datos ingresados por el usuario son erroneos, se mostrará un cuadro con el detalle del error.

Ilustración 88. Notificación al configurar nueva correlación.



Al dar clic en calcular se presentará la notificación de correlación calculada ilustración 72 y la tabla de correlación ilustración 73 (en color verde).

Ilustración 89. Notificación de correlación creada.

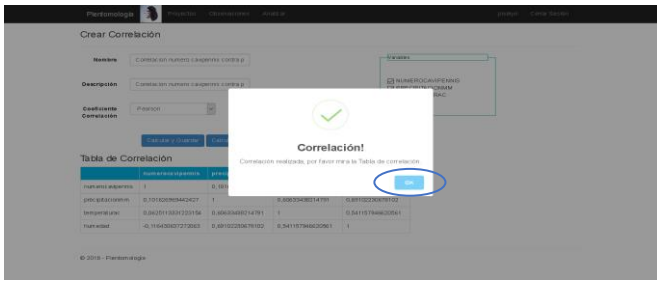
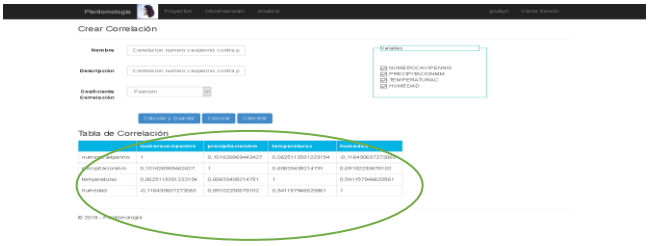


Ilustración 90. Tabla de correlación.



Manual de usuario para realizar sumarización por parte del investigador

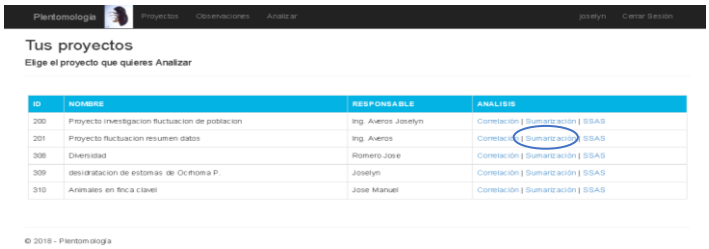
Tipo de usuario: Investigador. **Función:** Realizar análisis de sumarización.

Al ingresar correctamente los datos a la página de Inicio de sesión, aparecerá la página principal y un menú en la parte superior con la pestaña de analizar. Elegir esa opción.

Ilustración 91. Página principal, opción analizar de sumarización.

Al ingresar a la pestaña se presentará la lista con los diversos proyectos que cuenta el investigador, dar clic en el enlace del proyecto donde quiere ingresar nuevos análisis de sumarización.

Ilustración 92. Página lista de proyectos, opción sumarización.



Muestra una pantalla donde se presenta la lista de análisis de sumarización del proyecto, en caso de no tener ningun análisis solo se mostrará el botón de crear sumarización. Dar clic en crear sumarización.

Ilustración 93. Página lista de análisis de correlación.

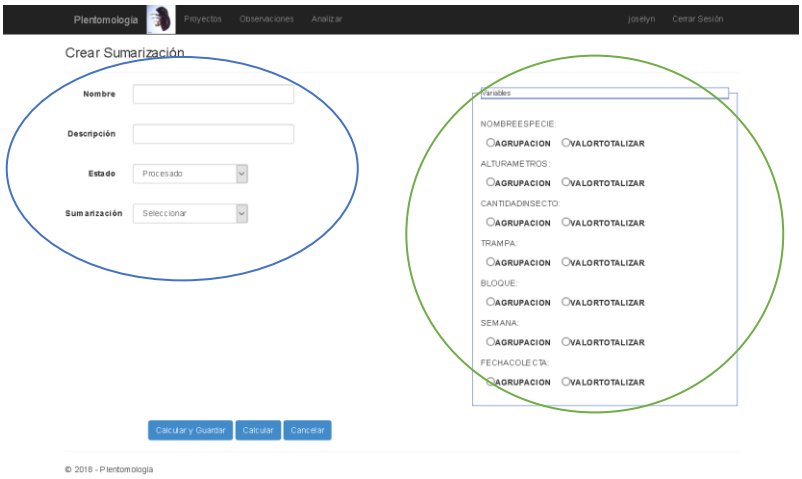
Al ingresar en crear sumarización se presentará un formulario para agregar la configuración del análisis y un cuadro en la parte derecha, para seleccionar las variables y el tipo de uso que se le dará en el análisis (en color verde). A continuación se detalla el significado del parámetro sumarización y el tipo de uso que soporta las variables.

Tabla 55. Detalle de parámetros de configuración sumarización.

Parámetro	Opciones soportadas
Sumarización: representa el tipo de cálculo que se va a realizar sobre el conjunto de datos.	Suma: suma todos los valores del conjunto de datos. Promedio: promedia los valores de un conjunto de datos. Varianza: calcula la varianza estadística de los valores de un conjunto de datos. Mínimo: valor mínimo del conjunto de datos. Máximo: valor máximo del conjunto de datos. Varianza estadística: calcula la varianza estadística de todos los valores de un conjunto de datos. Desviación estadística estándar: calcula la desviación estadística estándar para todos los valores de un conjunto de datos. Desviación típica estándar: calcula la desviación típica estadística de todos los valores de un conjunto.
Tipo de uso variables: representa como la variable se debe comportar en el cálculo.	Agrupación: se utiliza para producir un solo valor que cubra cada grupo. Valor a totalizar: representa el valor que se utilizará para realizar la operación.
Por ejemplo supongamos que se quiere obtener la cantidad total de insectos por nombre de especie. La sumarización seria de tipo suma porque se desea obtener la cantidad total	

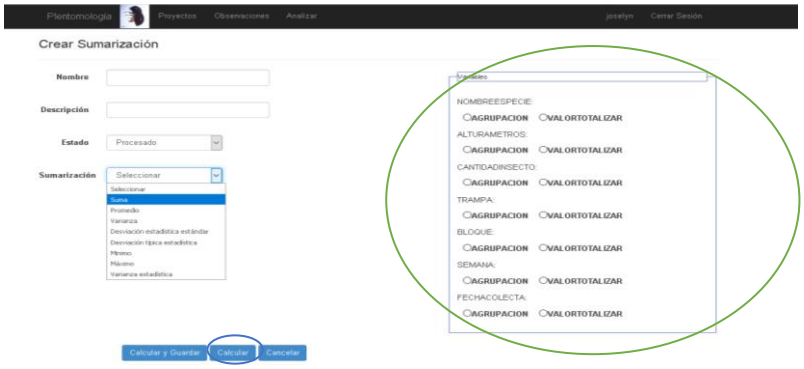
de insectos, lo cual seria la sumatoria de los insecto, pero se desea tener la cantidad por nombre de especie, lo que vendría a ser una agrupación.

Ilustración 94. Página crear sumarización, parámetros.



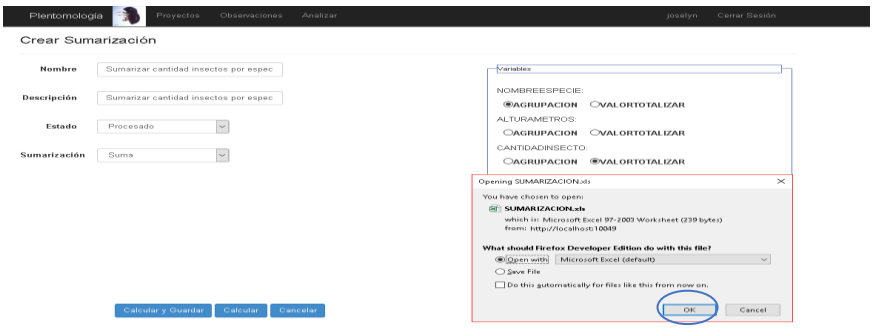
Ingresamos los datos solicitados y seleccionamos las variables a utilizar con su respectivo uso (en color verde). Luego damos clic calcular (en color azul).

Ilustración 95. Página crear sumarización, tipos de sumarización.



Los resultados del calculo se presentan en un archivo excel. Dar clic en ok para descargar.

Ilustración 96. Página sumarización, archivo Excel.



Aplicación móvil: **Manual de usuario registrar observación**

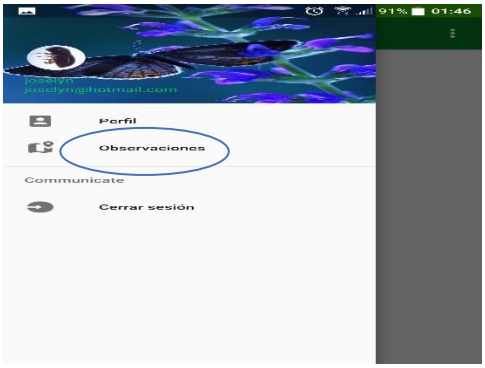
Tipo de usuario: Investigador. **Función:** Registrar observación.

Debe ingresar al sistema desde la aplicación móvil PIentomología. Al ejecutar la aplicación se mostrará el formulario de inicio de sesión, donde deberá ingresar el usuario y la contraseña previamente proporcionadas por el administrador.

Ilustración 97. Api PIentomología. Inicio de sesión.

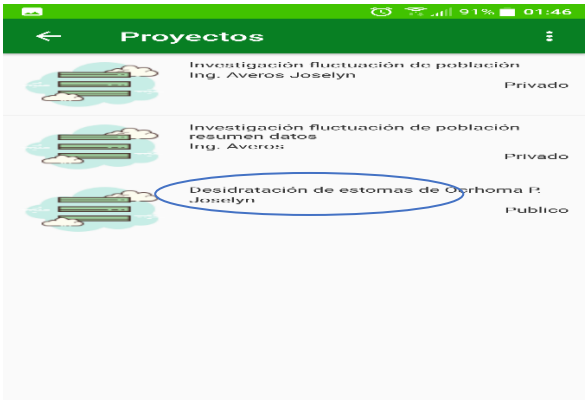
Al ingresar correctamente los datos aparecerá el menú del investigador. Elegir observaciones.

Ilustración 98. Menú de API PIentomología.



Se presentará lista con los diversos proyectos con los que cuenta el investigador, en caso de no tener ningun proyecto no se presentará nada. Pulsar en el proyecto donde quieres ingresar observaciones.

Ilustración 99. Api PIentomología, lista de proyectos.

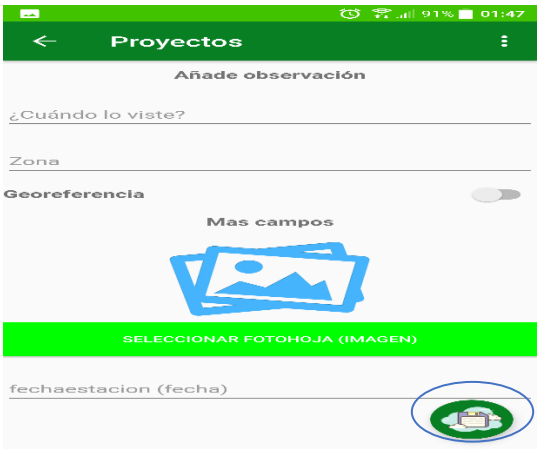


Se presentará la lista de las observaciones. Pulsar el botón flotante de la parte inferior derecha para registrar una nueva observación.

Ilustración 100. API PIentomología, lista de observaciones.

Se abrirá el formulario para ingresar los datos de la nueva observación. Ingresa los datos de la observación y pulsa guardar.

Ilustración 101. API PIentomología, formulario crear observación.



Requisitos mínimos aplicación web PIentomología:

- Compatible en navegadores Mozilla, Google Chrome.
- Compatible en computadores que corren sistema operativo Windows.

Requisitos mínimos aplicación móvil PIentomología:

- Compatible con móviles que corren sistema operativo Android mayores a 4.1.1.
- Cámara trasera.
- GPS.