

Herramienta de gestión en los procesos administrativos del GAD - La Maná

Management tool in the administrative processes of the GAD - La Maná

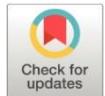
- ¹ Klever Xavier Almachi Cajas
Universidad Técnica de Cotopaxi
klever.almachi5@utc.edu.ec
- ² Jorge Enrique Cañar Tercero
Universidad Técnica de Cotopaxi
jorge.canar@utc.edu.ec
- ³ Johnny Xavier Vergara Brito
Universidad Técnica de Cotopaxi
johnny.vergara8@utc.edu.ec
- ⁴ Tamara Victoria Reyes Moreira
Universidad Técnica de Cotopaxi
tamara.reyes1@utc.edu.ec

 <https://orcid.org/0000-0001-6153-4637>

 <https://orcid.org/0000-0002-4591-2041>

 <https://orcid.org/0000-0001-9674-8141>

 <https://orcid.org/0000-0002-6079-0062>



Artículo de Investigación Científica y Tecnológica

Enviado: 14/12/2021

Revisado: 29/12/2021

Aceptado: 03/01/2022

Publicado: 18/02/2022

DOI: <https://doi.org/10.33262/ap.v4i1.2.185>

Cítese:

Almachi Cajas, K. X., Cañar Tercero, J. E., Vergara Brito, J. X., & Reyes Moreira, T. V. (2022). Herramienta de gestión en los procesos administrativos del GAD - La Maná . AlfaPublicaciones, 4(1.2), 63–77. <https://doi.org/10.33262/ap.v4i1.2.185>



ALFA PUBLICACIONES, es una Revista Multidisciplinar, **Trimestral**, que se publicará en soporte electrónico tiene como **misión** contribuir a la formación de profesionales competentes con visión humanística y crítica que sean capaces de exponer sus resultados investigativos y científicos en la misma medida que se promueva mediante su intervención cambios positivos en la sociedad. <https://alfapublicaciones.com>

La revista es editada por la Editorial Ciencia Digital (Editorial de prestigio registrada en la Cámara Ecuatoriana de Libro con No de Afiliación 663) www.celibro.org.ec

Esta revista está protegida bajo una licencia Creative Commons Attribution Non Commercial No Derivatives 4.0 International. Copia de la licencia: <http://creativecommons.org/licenses/by-nc-nd/4.0/>

Palabras claves:

herramientas,
gestión,
almacenamiento,
datos, interfaz
grafica

Keywords:

tools,
management,
storage, data,
graphical interface

Resumen

Introducción: es notorio que los avances tecnológicos cada vez son más frecuentes, esto se evidencia con el desarrollo de nuevas aplicaciones que generan bienestar y contribuyen de manera positiva en las actividades que se desarrollan en las organizaciones a nivel mundial, en este contexto la presente investigación se centra como objetivo principal el desarrollar herramientas de gestión de procesos administrativos, La investigación de la desarrolló mediante un enfoque cuantitativo, y cualitativo de nivel descriptivo, como técnicas se aplicaron encuestas y entrevistas validadas por expertos en las áreas afines. La población objeto de estudio correspondió al personal administrativo del GAD-La Maná y a los habitantes del mismo cantón. Entre los resultados destacados se puede mencionar que fue necesario la implementación de esta herramienta de gestión de procesos ya que anteriormente se lo desarrollaba manualmente generando pérdida de recursos como tiempo, dinero e información, el sistema posee una base de datos segura y confiable que permite reducir los tiempos de respuesta, reemplazar el trabajo manual y generar documentos de respaldo, gracias a las prestaciones de búsqueda, el sistema otorga una medida eficiente de almacenamiento, distribución por bloques, ingresar y actualizar datos, consultar y localizar todo tipo de información, además cuenta con una interfaz gráfica amigable con el usuario.

Abstract

Introduction: it is well known that technological advances are becoming more frequent, this is evidenced by the development of new applications that generate well-being and well-being in a positive way in the activities that are developed in organizations worldwide, in this context this research focuses As the main objective to develop management tools for administrative processes, the research developed a quantitative and qualitative approach at a descriptive level, as techniques were applied surveys and interviews validated by experts in related areas. The population under study corresponds to the administrative staff of GAD - La Maná and the inhabitants of the same canton. Among the outstanding results it can be mentioned that it was necessary to implement this process management tool since previously it

was developed manually, generating loss of resources such as time, money and information, the system has a safe and reliable database that allows reducing response times, replace manual work and generate supporting documents, thanks to the search features, the system provides an efficient measure of storage, distribution by blocks, entering and updating data, consulting and locating all kinds of information, it also has with a user-friendly graphical interface.

Introducción

En el mundo todos los métodos, técnicas y procesos automatizados, se han convertido en elementos de gran importancia para las instituciones públicas o privadas, con el avance de los años y la evolución tan rápida que ha tenido la tecnología hemos sido testigos de las grandes innovaciones y desarrollos tecnológicos. La presente investigación se desarrolló en base a las necesidades identificadas en la Administración del Camposanto del Cantón La Maná, debido a que no existía un sistema para su gestión, los procesos se los realizaba manualmente. Luego de haber implementado la investigación desarrollada la misma aporta a la gestión de la información de manera sistematizada, se organizan los procesos individualmente, y digitaliza los documentos antes mencionados y adicionalmente el sistema viene implementado con el proceso de contribución por prestación de servicios estipulado legalmente en el **Art. 53 Coste de los servicios** de la ordenanza Municipal para los camposantos en el Cantón La Maná. Los beneficios de la implementación de un sistema de gestión, es que la información será procesada y guardada de manera segura para que los procesos de control de los usuarios sean de forma más rápida y confiable, el sistema obtendrá una respuesta positiva por parte de la comunidad, por la cual se sentirán seguros y confiados de que la información cuenta con un control de calidad en torno a la gestión del camposanto.

Métodos

Para el desarrollo de la investigación, se realizó el levantamiento de información en dos ámbitos, el primero en el GAD del cantón La Maná y segundo aplicado a los pobladores del mismo cantón. La investigación se la desarrollo bajo un enfoque cuantitativo y cualitativo, como técnica se aplicó la encuesta y entrevista cuyos instrumentos de medición se basó en el cuestionario que fue validado y sometido a la experticia de personal con conocimientos en la rama.

Como población objeto de estudio se consideró al personal administrativo que administra el camposanto y una muestra de los habitantes del cantón La Maná.

Tabla 1
Técnicas e instrumentos

N.º	Técnicas	Instrumentos
1	Encuestas	Cuestionario
2	Entrevistas	Test

Tabla 2
Población objeto de estudio

Agente y/o Tecnologías	Funciones	Técnicas, espacios de trabajo y difusión	Población	Muestra	Cantidad Total
Población	Objeto de estudio	Población	49,897	265	265
Profesionales	Ing. Medio Ambiente Ing. Sistemas	GAD Municipal del Cantón La Maná	2	2	2
Directivos	Colaboran en la ejecución del proyecto	Oficinas administrativas GAD Municipal del Cantón La Maná	3	3	3
				TOTAL	273

Ingeniería de Software

La Ingeniería de Software es aquella disciplina que se ocupa del desarrollo, la operación y el mantenimiento del software o programas informáticos. El proceso de desarrollo de un software se denomina formalmente como ciclo de vida del software, en tanto, se encuentra conformada por cuatro estadios: concepción, en esta se fijan los objetivos y se desarrolla el modelo elaboración (en este paso se establecen las características y cómo será la arquitectura del mismo y porqué), construcción implica el desarrollo del programa y transición (es el momento en el cual se transfiere el producto final al usuario) (DifinicionABC, 2013).

Ciclo de Vida del Software

Es el proceso que se sigue para construir, entregar y hacer evolucionar el software, desde la concepción de una idea hasta la entrega y retiro del sistema. Se definen las distintas fases intermedias que se requieren para validar el desarrollo de un software, es decir, para garantizar que el software cumpla los requisitos para la aplicación y verificación de los procedimientos de desarrollo, se asegura de que los métodos utilizados son apropiados Herramientas para el desarrollo (EcuRed, 2016)

SCRUM, metodología de desarrollo de software ágil

Dentro de los muchos *frameworks* para desarrollar software, Scrum se ha convertido en los últimos tiempos en un estándar que muchas grandes empresas de tecnología están utilizando para acortar sus tiempos de desarrollo, y entregar un producto de calidad.

Aproximadamente en 1986 Hirotaka Takeuchi e Ikujiro Nonaka describieron una nueva forma para el desarrollo productos comerciales, que incrementaba la rapidez y la flexibilidad en el proceso. Ellos comparan este nuevo método, en la cual las fases se traslapan de manera intensa y el proceso completo es realizado por un equipo con funciones diversas, como en el rugby, donde el equipo entero actúa como “un solo hombre para intentar llegar al otro lado del campo, pasando el balón de uno a otro”. Estos casos de estudio se originan de las industrias automovilísticas, así como de fabricación de máquinas fotográficas, computadoras e impresoras.

En 1991 Peter DeGrace y Leslie Stahl en su libro *Wicked Problems, Righteous Solutions*, se refirieron a esta aproximación como Scrum, un término propio del rugby mencionado en el artículo por Takeuchi y Nonaka (Agil, 2017).

Qué es el software propietario

El software privativo hace referencia a aquel programa en el que los usuarios tienen limitadas las posibilidades de uso, análisis, modificación o distribución mediante copias. En otras palabras: existe una persona o entidad que posee derechos sobre el programa y que limita el libre uso, la posibilidad de analizarlo, de incorporar mejoras, de publicar los resultados del análisis o de distribuirlo libremente.

Software propietario

Por norma general las empresas proveedoras de software propietario cuentan con soporte técnico para atender a sus clientes. De esta forma podremos realizar labores de mantenimiento y reparación a un coste por lo general inferior al que tendríamos que asumir en el caso de usar software libre. Conviene recordar en este punto que un modelo de negocio bastante lucrativo consistía en distribuir libremente el software y generar ingresos con la prestación de servicios adicionales como la programación, la instalación, el mantenimiento etc.

Además, por lo general se trata de empresas dedicadas exclusivamente al desarrollo y la mejora de productos. Cabe esperar que su personal esté más capacitado y conozca mejor su producto que nosotros para solucionar problemas en caso de avería.

Podemos deducir de las ventajas anteriores que este soporte técnico reduce considerablemente la necesidad de contar con personal técnico preparado para hacer frente a posibles averías o a las labores de mantenimiento rutinario.

Las empresas proveedoras de software privativo suelen tener un tamaño considerable que les permite dedicar un importante volumen de recursos a labores de investigación y desarrollo (IEBS, 2017).

Lenguaje de Modelamiento Unificado (UML)

Todo lenguaje (formal o natural) es el mapa de una “Realidad “. Es capaz de expresar hechos a partir de la combinación de objetos y eventos. UML define ambos conceptos y facilita la trazabilidad de sus interacciones para acotar los escenarios de una organización. Con UML podemos organizar nuestro conocimiento utilizando tres coordenadas.

Diagrama de Casos de Uso

Un Diagrama de Casos de Uso muestra la relación entre los actores y los casos de uso del sistema. Son los principales medios para capturar la funcionalidad del software a implementar, representan la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa, describen acciones y reacciones al comportamiento de un sistema desde el punto de vista del usuario. Están basados en lenguaje natural (Campderrich, 2017).

Modelos de Power Designer

Logical Data Model

Un modelo de datos lógicos se describen los datos en tantos detalles como sea posible, sin tener en cuenta cómo van a ser física implementados en la base de datos.

Características de un modelo de datos lógicos incluyen:

- Incluye todas las entidades y relaciones entre ellos.
- Todos los atributos de cada entidad se especifican.
- Se especifica la clave principal de cada entidad.
- Las claves externas (claves que identifican la relación entre las diferentes entidades) se especifican.
- La normalización se produce a este nivel.

Physical Data Model

Modelo físico de datos representa cómo el modelo será construido en la base de datos, un modelo de base de datos física muestra todas las estructuras de la tabla, incluido el nombre de columna, tipo de la columna, las restricciones de columna, clave primaria, clave externa, y las relaciones entre las tablas.

Características de un modelo de datos físicos incluyen:

- Especificación de todas las tablas y columnas.
- Las claves externas se utilizan para identificar las relaciones entre las tablas.
- Desnormalización puede producirse sobre la base de las necesidades del usuario.
- Las consideraciones físicas pueden hacer que el modelo de datos físico a ser bastante diferente del modelo de datos lógicos.

Conceptual Data Model

Un modelo conceptual de datos identifica las relaciones de más alto nivel entre las diferentes entidades.

Características del modelo conceptual de datos incluyen:

- Incluye las entidades importantes y las relaciones entre ellos.
- No se especifica ningún atributo.
- No se especifica ninguna clave primaria (1keydata, 2016).

Como base principal para la consecución del desarrollo del sistema de gestión de cementerios si hicieron uso de herramientas que se van detallando a continuación, cada una de ellas cumple su función específica.

Base de Datos Oracle 11 G

Alta disponibilidad con Oracle Database 11g

Incorporar un sistema básico de tolerancia a fallas en una infraestructura de TI es una tarea sencilla. Es posible agregar algunos componentes redundantes y obtener tolerancia a fallas o alta disponibilidad. Si la estructura de TI presenta alguna falla, se puede conmutar a un componente redundante disponible. Teniendo en cuenta este principio básico, algunos clientes han creado un marco de alta disponibilidad que consiste en:

- Un modelo activo-pasivo de agrupación en clústeres de servidores N+1 (por ejemplo, clústeres integrados en el sistema operativo);
- La duplicación de los bits en la matriz de almacenamiento en alguna otra matriz remota de almacenamiento;
- Un producto de *backup* en cinta que garantiza la realización de *backups* periódicos y su almacenamiento fuera del centro;
- Un producto de administración de volumen independiente que facilita la administración del almacenamiento subyacente. Este tipo de configuración funciona, aunque con importantes limitaciones, del siguiente modo:

En general, las soluciones mencionadas corresponden a diferentes proveedores. La combinación y la administración de estas soluciones dispares exigen mucho esfuerzo. Aunque las soluciones de alta disponibilidad basadas en hardware (por ejemplo, la duplicación) constituyen métodos simples de protección de datos, su enfoque por bytes dificulta la creación de capacidades de aplicación optimizada (Availability, 2010).

Developer 6i

Developer 6i, posee su propio lenguaje de red, que se asienta de igual manera sobre casi cualquier protocolo; este es Net8 (antiguo Sql-Net). Este protocolo permite la configuración, e implementación sobre otros protocolos debido a su versatilidad, es decir, se adapta a los tamaños

de tramas de red, y resulta una solución de envío y recepción de datos en cualquier red a nivel LAN. Si se desea realizar un desarrollo y web tenemos dos alternativas, mediante *Oracle forms y Reports* los cuales generan su propio código, al momento de llevarlo a web este código se convierte en applets de java y si solo funciona con Oracle IAs. La segunda opción es desarrollar todo mediante *servlets* a través de *Developer* que también está incluido en el IDS, este ya genera código más portable que permite ser llevado a distintas plataformas que soporten *servlets*. El proceso de desarrollo de aplicaciones abarca el ciclo de vida de desarrollo, una serie de productos, documentos de proyecto y un conjunto de herramientas que funcionan conjuntamente en un entorno de desarrollo. El ciclo de vida de desarrollo es un proceso estándar a través del cual se construyen las aplicaciones. Los documentos de proyectos es un producto del ciclo de vida que describe algunos aspectos de las aplicaciones que se van a desarrollar, como los requisitos, los planes de prueba y del proyecto. Este sistema desarrollado en *Oracle Developer 6i* es multicompañía, y permite llevar un control de las compras, facturación, apartados e inventario. Maneja clientes, proveedores, bodegas y artículos, paquetes de artículos, tipos de precios y fotografías. Es especial para tiendas, ferreterías, farmacias, supermercados, mediante la definición de estilos y a partir de un modelo de base de datos del sistema a desarrollar, este software, genera pantallas funcionales con *triggers*, procedimientos, listas de valores, y diseño gráfico estándar (Bryan, 2010).

TOAD SQL

Toad es la herramienta líder en la industria para el desarrollo y el manejo de bases de datos que aumentan la productividad del usuario y la calidad de código de aplicaciones. Esto le ofrece al usuario una comunidad interactiva de soporte para mejorar aún más la experiencia. *Toad* es una poderosa herramienta que hace el desarrollo de bases de datos y aplicaciones mucho más fácil y rápido, mientras que simplifica las tareas cotidianas de administración. Así usted desarrolle aplicaciones, bases de datos o analice negocios, *Toad* ofrece características específicas que le volverán mucho más productivo. Al contar con

la capacidad de ofrecer liderazgo en varias plataformas, *Toad* me permite extraer mayor valor de su ambiente heterogéneo de bases de datos. TOAD fue originalmente diseñada por un desarrollador en prácticas de Oracle, Jim McDaniel.

TOADman, para hacer su propia vida más fácil. El llamó a esto "Herramienta para Desarrolladores de Aplicación Oracle", acrónimo en inglés TOAD (SoftwareShop, 2017).

Especificación y Análisis de Requerimientos

Este proceso permite conocer el ámbito y flujo de la información en general; el objetivo principal en esta etapa es determinar la naturaleza de la herramienta informática o software, necesidades a cubrir, capacidades y funciones requeridas. Los requerimientos identificados se deben especificar de manera clara y precisa, sin lugar a ambigüedades, y cubriendo el alcance necesario, para así lograr los objetivos propuestos y obtener resultados válidos. Es necesario, además, evaluar y definir la criticidad y complejidad del software para así poder determinar su nivel de riesgo. Para el caso de software comercial, contar con una adecuada especificación de requerimientos permite realizar una correcta selección del software a adquirir. Para el caso de los tipos modificable y personalizado, la especificación de requerimientos constituirá la base para la definición de objetivo final y alcance del software.

La importancia de este proceso se centra en que la calidad del producto se basa en el cumplimiento de los requerimientos planteados y, por consiguiente, de las necesidades y expectativas de los implicados.

Metodología de desarrollo SCRUM

Las metodologías que se han propuesto para conseguir mejoras de trabajo en equipo frente al individual aplican dos formas básicas de trabajo: 1) dividir el problema entre los integrantes del equipo y luego unir las soluciones parciales que hayan elaborado, o 2) resolver todo el mismo problema y luego mezclar diferentes soluciones en una sola que aproveche todas las fortalezas que aporten las soluciones parciales. Existen numerosas metodologías de trabajo en equipo, alguna que permita adaptarse a desarrollos dinámicos de forma compatible con el ciclo de mejora continua.

Una clase de ellas incluye las denominadas metodologías ágiles, de las cuales Scrum es uno de los mejores exponentes y cuyo origen está orientado hacia la producción de software. Sin embargo, nosotros loaremos su filosofía general para adaptarla a cualquier desarrollo de trabajo en equipos en Ingeniería de modo que faciliten la planificación, coordinación, seguimiento, integración y evaluación del trabajo realizado.

En Scrum, la adaptación a los cambios del mercado, o de los requisitos del cliente, y la rapidez y calidad de los resultados se consiguen con la noción de *Sprint*. Una metodología

ágil del tipo Scrum proporciona resultados en muy poco tiempo mediante *sprints* muy cortos, con participación

propietario del producto en el proceso de evaluación de cada resultado parcial y en la determinación de las acciones a desarrollar en el próximo Sprint.

Algunas de las características de esta metodología son las siguientes:

- Cada ciclo se denomina Sprint.
- En el desarrollo del producto o servicio deben considerarse a todos los agentes interesados en el mismo. En particular se distinguen el Propietario del Producto (EP), el Propietario de la Empresa de Desarrollo, el Scrum-Máster y el resto de miembros del equipo.
- Al final e inicio de cada ciclo se mantiene una reunión con el Propietario del Producto para evaluar el producto y determinar las características a desarrollar en su próxima versión (López & Vaquerizo, 2013).

Resultados

Existen requisitos mínimos para el arranque del sistema, y estos se presentan en las tablas 3 y 4.

Tabla 3

Requisitos Arranque del Sistema Cliente

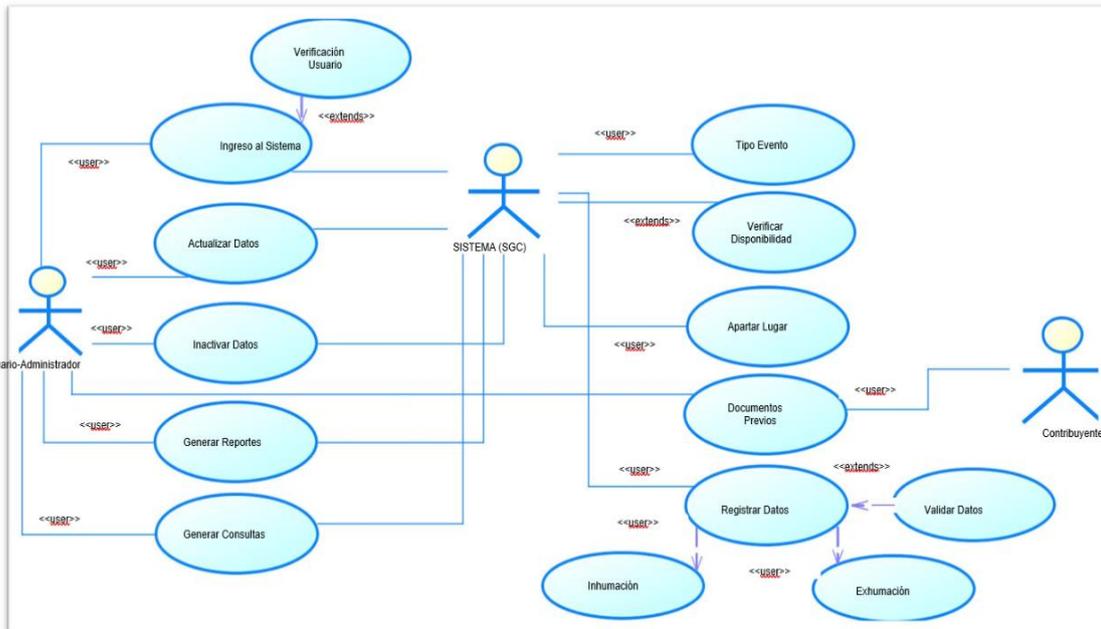
Nivel cliente	
Memoria mínima	1 GB
Espacio en disco mínimo	150 GB
Procesador	Pentium 4
Tipo de Sistema	32 bits a 64 bits
Sistema Operativo	Windows Xp a Windows 10

Tabla 4

Requisitos arranque sistema servidor

Nivel servidor	
Memoria mínima	4 GB
Espacio en disco mínimo	1 Terabyte
Procesador	Core 5
Sistema Operativo	Windows, Linux y Solaris

Figura 1
Diagrama general del sistema

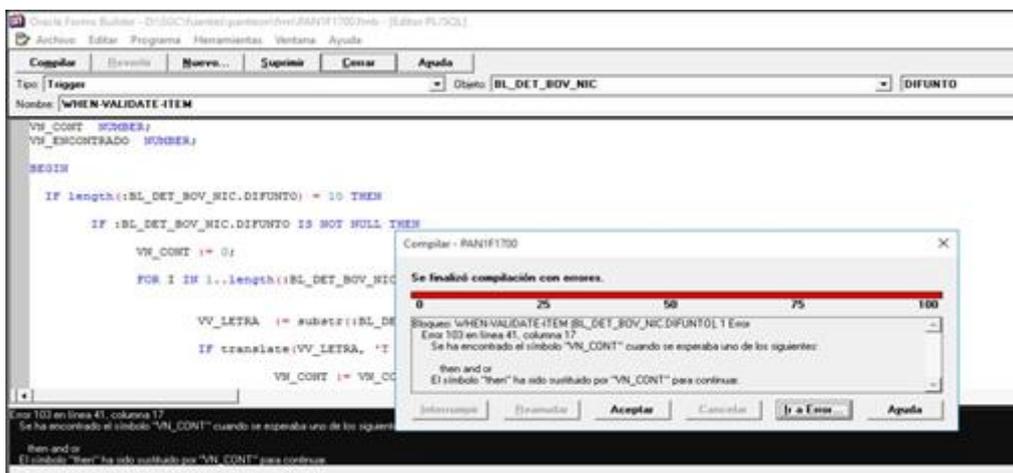


Nota: diagrama general en el que se muestra la relación lógica del sistema

Pruebas del Software Caja Blanca

Se procedió a realizar las pruebas de caja blanca al código fuente del sistema con su respectiva compilación para verificar ciertas fallas con el fin de corregir errores del sistema al momento de su ejecución y manejo.

Figura 2
Formulario error de programación



Nota: Formulario PAN1F1700 Error de Programación no se escribió correctamente la sentencia de control

Figura 3

Modelo físico de datos

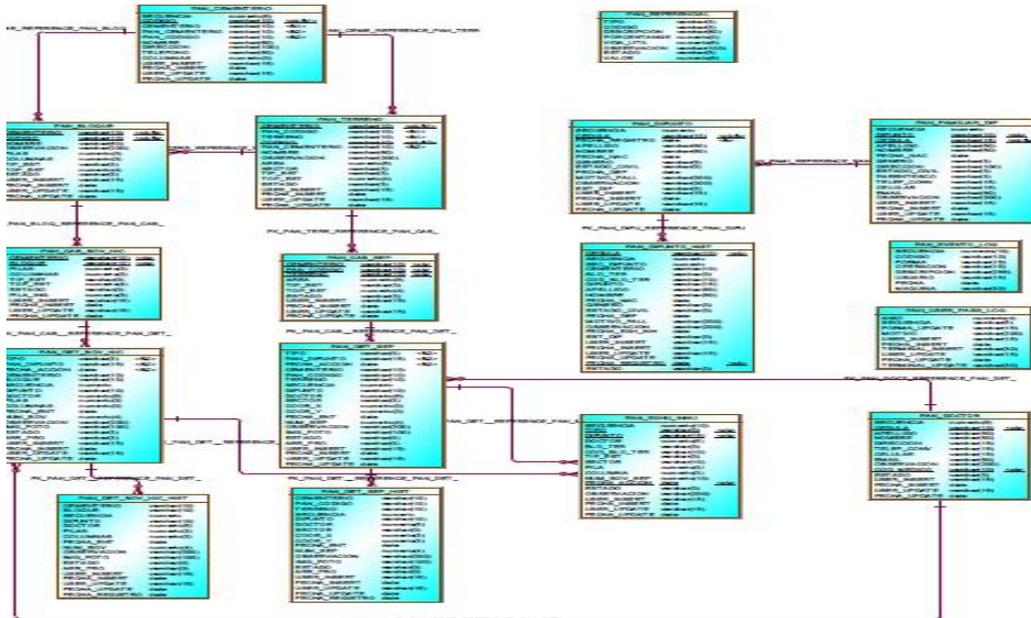


Figura 4

Ventana menú inicial del sistema



Figura 5

Ventana de Menús del Sistema



Figura 6

Ventana de registro de camposantos

Conclusiones

- Se estableció como punto de partida la necesidad de que el GAD Municipal del Cantón La Maná cuente con un sistema de gestión de camposantos.
- Se utilizó la metodología Scrum para el desarrollo de sistema de gestión de camposantos.

- La presente implementación se la puede tomar como una guía para futuras aplicaciones con las herramientas apropiadas.
- Para el desarrollo del diseño, la interfaz de la base de datos se lo realizó en base a la información recolectada.
- La implementación de un sistema de gestión de campos santos son soluciones completamente adaptables a todos los entornos tecnológicos.

Referencias bibliográficas

- 1keydata. (2016). *1keydata*. 1keydata: <https://www.1keydata.com/datawarehousing/conceptual-data-model.html>
- Agil, S. M. (2017). *http://blog.espol.edu.ec*. Obtenido de <http://blog.espol.edu.ec>: <http://blog.espol.edu.ec/elpoli/scrum-metodologia-de-desarrollo-de-software-agil/>
- Availability, O. D. (2010). *Alta disponibilidad con Oracle Database 11g versión 2*. Oracle Database High Availability: <http://www.oracle.com/technetwork/es/database/enterprise-edition/documentation/disponibilidad-oracle-database11gr2-2247600-esa.pdf>
- Bryan. (2010). *Oracle Developer 6i*. Oracle Developer 6i: <http://clienteservidor2.blogspot.com/2010/11/oracle-developer-6i.html>
- Campderrich, B. (2017). *Ingeniería de Software*. Barcelona: UOC.
- DifinicionABC. (2013). *DifinicionABC*. DifinicionABC: <https://www.definicionabc.com/tecnologia/ingenieria-de-software.php>
- EcuRed. (2016). *EcuRed*. EcuRed: https://www.ecured.cu/Ciclo_de_vida_del_software
- IEBS. (2017). *IEBS*. IEBS: <http://www.iebschool.com/blog/software-propietario-digital-business/>
- López, D. & Vaquerizo, B. (2013). *Jenui 2013*. Castellon de la Plana: IEEE.
- SoftwareShop. (2017). *SoftwareShop*. SoftwareShop: <https://www.software-shop.com/producto/toad>

El artículo que se publica es de exclusiva responsabilidad de los autores y no necesariamente reflejan el pensamiento de la **Revista Alfa Publicaciones**.



El artículo queda en propiedad de la revista y, por tanto, su publicación parcial y/o total en otro medio tiene que ser autorizado por el director de la **Revista Alfa Publicaciones**.



Indexaciones

