

Universidad de Matanzas



Facultad de Ciencias Técnicas



Departamento de Informática

Trabajo de Diploma en opción al título de Ingeniero Informático

Título: Software de apoyo a la gestión de la información en el Organismo Político del MININT, Matanzas.

Autor:

Geykel Alberto Hernández Cruz

Tutor(es):

MCs. Mayli Estopiñán Lantigua

Ing. Ernesto Henríquez Rivero

Consultor(es):

María Caridad Díaz Mesa

Matanzas, Cuba

Junio, 2018

Pensamiento

“La única forma de hacer un buen trabajo es amando lo que haces”



Steven Jobs

“Los obstáculos son esas cosas que vez cuando apartas los ojos de la meta”



Henry Ford

Dedicatoria

Dedico esta investigación de manera muy especial a toda mi familia y esas personas que de una forma u otra me han apoyado incondicionalmente a lo largo de todos estos años, por haber confiado siempre en mí.

Agradecimientos

A mi madre, por siempre estar a mi lado en pie de guerra, por enseñarme que nunca se abandonan los sueños y que la voluntad de un hombre forja su futuro.

A mi familia por su ayuda durante todos estos años de estudio lejos de casa, por haberme apoyado y haber creído en mí.

A mi padre y a su esposa por sus consejos, por su confianza y comprensión.

A mis abuelos por ser las personas que más admiro en el mundo, por ser mi guía en la vida y por preocuparse tanto.

A Ismany que ha sido un padre para mí y por siempre estar pendiente de todo en mi vida.

A mis tutores: Mayli Estopiñán Lantigua y Ernesto Henríquez Rivero por toda su entrega y dedicación, por ser siempre tan positivos, darme seguridad, brindar lo mejor de sí y apoyarme en todo momento.

A mi Ada Madrina: Wendy que nunca me falla y siempre está ahí cuando más la necesito.

A mis cotutores Ángel Luis y Ariel por siempre verle el lado agradable a las cosas y encontrar las soluciones más prácticas a los problemas más difíciles.

A María Caridad Díaz Mesa por ser cliente, consultora, profesora, por siempre tener tiempo para atenderme y por brindarme sus conocimientos cuando los necesité.

A mis amigos por siempre estar ahí para mí en las buenas y en las malas.

A todo el piquete de la universidad (Reniel, Jorge, Ediel, Marlom, Roberto, Wuilliam, Zuliam, Heidy, Lianet y Yeni), estos 5 años con ustedes han sido maravillosos y nos hemos convertido en una gran familia.

A los profesores Omar Rivero Muñiz y Walfredo González Hernández por sus enseñanzas y apoyo en estos años.

A los profesores de la Universidad y del Departamento de Informática que durante esta etapa me formaron como profesional.

A los trabajadores del órgano de informática y comunicaciones del MININT que me ayudaron y estuvieron pendiente en esta recta final.

A todos, de corazón, muchas gracias.

Declaración de autoría

Yo, Geykel Alberto Hernández Cruz declaro que soy el único autor de esta investigación, confeccionada para optar por el título de Ingeniero Informático, realizada en la Delegación Provincial del Ministerio del Interior de Matanzas, por lo que autorizo a que los resultados de este estudio sean utilizados por la institución con la finalidad que se estime conveniente.

Y para que así conste firmo la presente a los ____ días del mes de _____ del 2018.

Firma del Autor

Geykel Alberto Hernández Cruz

Firma del Tutor

Mayli Estopiñán Lantigua

Firma del Tutor

Ernesto Henríquez Rivero

Resumen

El presente trabajo de diploma hace un estudio cuidadoso y detallado de la situación existente en el Organismo Político del Ministerio del Interior Provincial de Matanzas con respecto a gestión de la información referente a los procesos de “preservación de la memoria histórica, preparación política y realización de eventos”, debido a que en la actualidad existen dificultades en la forma de almacenar y procesar los datos necesarios para su planificación y control, ya que se realizan con el apoyo de modelos basados en Microsoft Excel, Microsoft Word y en formato papel, por diferentes personas con diversos modos de organización, provocando que se ejecuten de forma engorrosa y lenta, lo que propicia la ocurrencia sistemática de errores y la alta pérdida de información, por ello se desarrolló un software que contribuye y apoya a que estos procesos se lleven a cabo de forma más organizada, accesible, rápida, integral y segura. Para su descripción y desarrollo se utilizaron métodos teóricos, empíricos y herramientas argumentadas internacionalmente; fue utilizada la metodología de desarrollo de software Programación Extrema (XP), como servidor web Internet Information Services (IIS), como marco de trabajo Entity Framework, lenguaje de programación C# y como gestor de bases de datos Oracle 11g. Como resultado de la investigación se obtuvo el software SAGIOP que incorpora diversas funcionalidades que responden a las necesidades y exigencias del cliente en su totalidad, a dicho software se le realizaron diversas pruebas que permitieron argumentar su utilidad, interactividad y seguridad.

Summary

The present diploma work makes a careful and detailed study of the existing situation in the Political Organism of the Provincial Ministry of the Interior of Matanzas regarding information management regarding the processes of "preservation of historical memory, political preparation and realization of events ", due to the fact that currently there are difficulties in the way of storing and processing the necessary data for planning and control, since they are carried out with the support of models based on Microsoft Excel, Microsoft Word and in paper format, for different people with diverse organizational modes, causing them to be executed in a slow and cumbersome manner, which favors the systematic occurrence of errors and the high loss of information, for this reason software was developed that contributes and supports these processes to be carried out in a more organized, accessible, fast, comprehensive and secure way. For its description and development, theoretical, empirical methods and internationally argued tools were used; the software development methodology Extreme Programming (XP) was used, as an Internet Information Services (IIS) web server, as an Entity Framework framework, a C # programming language and as an Oracle 11g database manager. As a result of the research SAGIOP software was obtained that incorporates various functionalities that respond to the needs and requirements of the client in its entirety. The software was subjected to various tests that allowed arguing its usefulness, interactivity and security.

Índice

Introducción.....	1
Capítulo 1: Marco teórico referencial.....	6
1.1 Introducción.....	6
1.2 Objeto de Estudio.....	6
1.2.1 Descripción de los elementos asociados al objeto de estudio	6
1.2.2 Flujo actual de los procesos involucrados en el campo de acción.....	7
1.2.3 Análisis crítico de cómo se ejecutan los procesos	9
1.3 Antecedentes de la investigación.....	9
1.4 Métodos de la investigación.....	11
1.4.1 Métodos teóricos empleados.....	11
1.4.2 Métodos empíricos empleados.....	11
1.5 Herramientas, tecnologías y metodologías de desarrollo	12
1.5.1 Metodologías de desarrollo	12
1.5.2 Tendencias tecnológicas a considerar.....	15
1.5.3 Tecnologías y Herramientas.....	18
1.6 Conclusiones del Capítulo.....	27
Capítulo 2: Análisis, diseño y desarrollo de la solución propuesta.....	28
2.1 Introducción.....	28
2.2 Descripción de la solución	28
2.3 Etapa de planificación.....	29
2.3.1 Equipo de trabajo y roles	29
2.3.3 Planificación de iteraciones	36
2.3.4 Reuniones	37
2.4 Etapa de diseño	37
2.4.1 Prototipo de interfaz de usuario.....	38
2.4.2 Tareas a desarrollar.....	38
2.4.3 Tarjetas de Clase, Responsabilidad y Colaboración.....	43
2.4.4 Modelo físico de la base de datos	46
2.5 Conclusiones del Capítulo.....	47
Capítulo 3: Validación de la solución propuesta.....	49
3.1 Introducción.....	49

3.2 Pruebas al software	49
3.2.1 Plan de pruebas	49
3.2.2 Pruebas de aceptación	52
3.2.3 Pruebas de seguridad	56
3.3 Análisis de los resultados obtenidos	58
3.4 Análisis de los Costos	59
3.5 Conclusiones del Capítulo	60
Conclusiones	61
Bibliografía	63
Anexos	67
Anexo 1: Informe de la actividad de Preparación Política	67

Índice de Tablas

Tabla 1: Comparación entre metodologías ágiles y no ágiles. (Muñiz, 2015).....	13
Tabla 2: Equipo de trabajo y roles.....	29
Tabla 3: Historias de Usuarios Iniciales	32
Tabla 4 HU 1: Diseño y Creación de la base de datos	32
Tabla 5 HU 4: Gestionar usuario	33
Tabla 6 HU 6: Gestionar recurso	33
Tabla 7 HU 7: Gestionar álbum	34
Tabla 8 HU 10: Gestionar evento	34
Tabla 9 HU 11: Gestionar preparación política	35
Tabla 10 HU 14: Generar reporte de asistencia	35
Tabla 11 HU 18: Generar reporte de almacenamiento.....	36
Tabla 12: Tareas de Ingeniería	40
Tabla 13 TI 2: Creación de la base de datos	40
Tabla 14 TI 6: Insertar usuario.....	41
Tabla 15 TI 14: Insertar recurso	41
Tabla 16 TI 18: Insertar álbum	42
Tabla 17 TI 36: Insertar evento	42
Tabla 18 TI 44: Insertar preparación política	43
Tabla 19 TI 57: Generar reporte de asistencia	43
Tabla 20 Tarjeta CRC: Usuario	44
Tabla 21 Tarjeta CRC: Recurso	45
Tabla 22 Tarjeta CRC: Álbum.....	45
Tabla 23 Tarjeta CRC: Evento	46
Tabla 24 Tarjeta CRC: Preparación Política	46
Tabla 25: Plan de Pruebas	52
Tabla 26: PA 3: Test autenticarse.....	52
Tabla 27: PA 4: Test insertar usuario.....	53
Tabla 28: PA 12: Test insertar recurso	54
Tabla 29: PA 16: Test insertar álbum	54
Tabla 30: PA 34: Test insertar evento	55
Tabla 31: PA 42: Test insertar preparación política	56

Índice de Figuras

Figura 1: Arquitectura Modelo-Vista-Controlador.....	18
Figura 2: Planificación de las Iteraciones	37
Figura 3: Página principal del software	38
Figura 4: Modelo físico de la base de datos.....	47
Figura 5: Injection Modules.....	57
Figura 6: Response Processing Modules.....	57
Figura 7: Resultados generales obtenidos por el Vega	58

Introducción

Hoy en día las Tecnologías de la Información y la Comunicaciones (TICs) ocupan un lugar esencial en el paradigma económico emergente, constituyen un elemento facilitador para el soporte de la eficacia y la eficiencia incidiendo en casi todos los aspectos de nuestra vida. Su uso representa una variación notable en la sociedad y a la larga un cambio en la educación, en las relaciones interpersonales y en la forma de difundir y generar conocimientos. (José, 2016)

En Cuba, continúa incrementándose la velocidad de procesamiento de la información y su almacenamiento a través de largos períodos de tiempo, estos avances sostienen la diversificación de los sistemas de gestión que expanden su acción hasta cualquier área empresarial u organismo. De este contexto surgen nuevos software, aplicaciones web y sistemas informáticos que permiten realizar análisis multidimensionales al brindar como resultado información útil y actualizada, facilitando su uso en todas las esferas de la sociedad.

El Ministerio del Interior (MININT) en la Provincia de Matanzas no está ajeno a las transformaciones descritas anteriormente promoviendo el empleo de las nuevas tecnologías para desempeñar más eficientemente el trabajo de sus oficiales en el territorio.

Dentro del MININT se encuentra el Organismo político, el cual tiene como **objeto social** la necesidad de fortalecer la labor político-ideológica y educativa del personal; rescatar y preservar la memoria histórica de la institución; además de convocar a la movilización política de las tropas para el cumplimiento exitoso de las nuevas misiones y tareas que se asignen, lo que no sería posible sin un sistema coherente que integre todos los esfuerzos para asegurar un mejor empleo de los recursos humanos y financieros. (MININT, Misiones del Organismo Político, 2017)

El Organismo Político del MININT Provincial de Matanzas cuenta con muchísima información almacenada en diversos formatos ya que en el mismo se gestionan procesos diferentes que hoy aún no se encuentran informatizados, ejemplo de estos son la “Preservación de la Memoria Histórica” proceso que incluye el manejo y la

recopilación de toda la información histórica del MININT desde sus inicios hasta la actualidad, biografías de personalidades determinadas, efemérides sobre hechos históricos y la propia historia de la provincia, encontrándose en diversas formas de almacenamiento, propensas a su pérdida y deterioro eminente por causa de que una parte son documentos antiguos que se encuentran en formato papel que se quieren preservar y poder consultar de forma digital sin tener que manipular el documento original evitando así su deterioro, así como otros en soporte digital. Este proceso en ocasiones resulta engorroso y deficiente puesto que si se quisiera consultar una información determinada como por ejemplo una fotografía tomada un día (x), esta no se puede encontrar fácilmente ya que las mismas son guardadas por diferentes personas con distintos modos de organización, teniendo que emplear mucho tiempo para la revisión de las fuentes y poder encontrar lo que se necesita. Repitiéndose el problema así para los diferentes archivos o evidencias que registran en este organismo.

Otro de los procesos fundamentales que se desarrollan en el Organismo son las actividades referentes a la “Preparación Política”. Esta labor tiene como objetivo mantener a los oficiales actualizados política e ideológicamente para que sean ejemplo ante las masas y ejecuten el papel que les corresponde en el control político de lo establecido por la institución y el propio Partido Comunista de Cuba (PCC). Estas actividades se realizan varias veces al mes en todos los grupos de estudio de la provincia, se llevan a cabo por parte de los monitores quienes se encargan de organizarlas, teniendo obligatoriamente que trasladarse desde cada una de las unidades hacia la cabecera provincial para poder obtener la información requerida en el desarrollo de las mismas, evidenciando así que no se aprovechan al máximo vías más eficientes para ello o sea con el uso de las TICs. Como estas actividades se efectúan frecuentemente generan una gran cantidad de estadísticas, la cual hoy se gestiona a través de modelos en Excel, concluido el mes y cumplidas todas las actividades planificadas se entregan las actas de estas en la cabecera provincial, pero posteriormente a la hora de ejecutar controles y reportes sobre la respuesta de las tropas a las mismas, la asistencia, los planteamientos de los combatientes y los temas que se abordan ya sean de meses o sobre años anteriores, se necesita hacer

una revisión exhaustiva de todas las actas para poder extraer los datos deseados necesarios para la toma de decisiones y asegurar el control de estos procesos, trayendo consigo, que no se puedan consultar en un período de tiempo corto, además de que se incurre en un mayor esfuerzo por los trabajadores del órgano.

La organización, divulgación y realización de “eventos” constituyen otro de los procesos esenciales que ahí se desarrollan, hoy muy puntuales para fomentar y cultivar la preparación política e ideológica, sobre todo de la fuerza joven. Entre las tareas que divulgan o planifican se encuentran: talleres, eventos científicos, conversatorios, concursos, etc. pero desprovistos de formas de divulgación masivas, así como de una forma específica para su almacenamiento y gestión como (sitio, repositorio o ftp).

A partir de lo cual, se identifica como **problema científico**:

¿Cómo contribuir al proceso de gestión de la información en el Organismo Político del MININT Provincial de Matanzas?

El **objeto de estudio** lo constituirá el proceso de gestión de la información en el Organismo Político del MININT Provincial de Matanzas.

El **campo de acción** de esta investigación lo constituye la automatización del proceso de gestión de la información en el Organismo Político del MININT Provincial de Matanzas.

Teniendo en cuenta el problema se propone trabajar en la siguiente **hipótesis**:

Si se desarrolla un software de apoyo a la gestión de la información en el Organismo Político del MININT Provincial de Matanzas se podrá contribuir a que este proceso se realice de una manera más eficiente.

De la hipótesis antes mencionada se deriva como **variable independiente** el desarrollo de un software de apoyo a la gestión de la información en el Organismo Político del MININT Provincial de Matanzas y como **variable dependiente** la gestión de la información en el Organismo Político del MININT Provincial de Matanzas

Para dar solución al problema anterior, se define como **objetivo general**: Desarrollar un software que sirva de apoyo al proceso de gestión de la información en el Organismo Político del MININT Provincial de Matanzas.

Basándose en el objetivo general se establecieron como **objetivos específicos** los siguientes:

- 1- Confeccionar el marco teórico referencial de la investigación.
- 2- Desarrollar un software de apoyo a la gestión de la información en el Organismo Político del MININT Provincial de Matanzas.
- 3- Validar el software de apoyo a la gestión de la información en el Organismo Político del MININT Provincial de Matanzas.

Como **resultados esperados** se desea que el Organismo Político cuente con un software capaz de contribuir con la gestión de la información referente a la memoria histórica, los eventos y la preparación política en el MININT Provincial de Matanzas y les permita realizar su trabajo con mayor facilidad, eficiencia y rapidez.

Para el desarrollo de la investigación se utilizaron diversos **métodos de investigación científica**.

Dentro de los **métodos teóricos** se utilizaron:

- ❖ Histórico-lógico.
- ❖ Análisis-síntesis.
- ❖ Inducción-deducción.
- ❖ Enfoque de sistema.

Como **métodos empíricos**, fueron analizados los siguientes:

- ❖ Observación.
- ❖ Entrevista.
- ❖ Análisis de documentos.

La investigación se estructuró en 3 capítulos, de la siguiente forma:

Capítulo 1: Marco teórico referencial

Recoge las definiciones fundamentales asociadas al dominio del problema. Se hace un estudio sobre las tendencias y tecnologías actuales que serán usadas. Se exponen las características fundamentales de los lenguajes de programación, los

sistemas de bases de datos y las características fundamentales de las metodologías de desarrollo de software ágiles.

Capítulo 2: Análisis, diseño y desarrollo de la solución propuesta

Presenta una propuesta de solución para el software, donde se describen las reglas y los elementos del negocio, una planificación inicial del proyecto con el empleo de la metodología de ingeniería del software *Extreme Programming* (XP). Se desarrolla la propuesta para darle solución a la situación problemática, presentando una planificación por iteraciones.

Capítulo 3: Validación de la solución propuesta

Se realizan las pruebas al software y un análisis de los resultados obtenidos con el objetivo de entregarle al cliente un producto totalmente funcional, cumpliendo con todos los requisitos demandados por el mismo y satisfaciendo sus necesidades. Además se presentan las conclusiones, recomendaciones, bibliografía y anexos.

Capítulo 1: Marco teórico referencial

1.1 Introducción

En este capítulo se analizan las bases teóricas que sustentan y fundamentan la investigación. Se abordan los conceptos esenciales con respecto al dominio del problema, el objeto de estudio, los antecedentes del trabajo y el análisis de algunas de sus características elementales con el objetivo de permitir una mejor orientación para su solución. Se definen las tecnologías, herramientas y metodologías que se utilizarán durante el desarrollo de la investigación.

1.2 Objeto de Estudio

El MININT es el organismo encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado y del Gobierno en cuanto a la organización, mantenimiento y defensa de la seguridad y el orden interno en el país. (Decreto Ley No.67, 1976)

La primera misión y gran tarea del MININT desde su creación ha sido defender a todo precio la seguridad nacional y del Estado Cubano, sin agredir ni causar daños a ninguna persona, además de mantener el orden interno, apoyar las labores de desarrollo del pueblo y participar en misiones internacionalistas.

Como parte del MININT el Organismo Político tiene como **misión** dirigir y controlar el trabajo del Partido Comunista de Cuba (PCC) y la Unión de Jóvenes Comunista de Cuba (UJC) en el MININT, apoyar a la jefatura en la instrumentación y control del trabajo político ideológico, en interés de la movilización de las fuerzas hacia el cumplimiento de sus misiones; proponer la proyección de divulgación de la labor e imagen de la Institución; coordinar y controlar la ejecución práctica de la política aprobada; influir y exigir por la calidad en la prestación de los servicios de atención a la población, así como en la implementación y cumplimiento de los Lineamientos del PCC y la 1era Conferencia del PCC. (MININT, Misiones del Organismo Político, 2017)

1.2.1 Descripción de los elementos asociados al objeto de estudio

Organismo Político MININT Matanzas: Organización que es rectora de instrumentar las ordenes que regulan el Sistema de Trabajo Político-Ideológico del MININT en la provincia de Matanzas.

Memoria Histórica: se refiere a la historia de la institución desde su fundación hasta la actualidad, la cual se encuentra registrada en documentos y materiales audiovisuales.

Programa para la Preparación Política: Programa de capacitación de la fuerza que se desarrolla en la institución con el objetivo de preparar a los combatientes y trabajadores civiles política e ideológicamente para el cumplimiento de sus misiones.

Indicaciones metodológicas: Materiales metodológicos de orientación para el desarrollo de la preparación política, es emitido por la Dirección Política y el Organismo Político.

Grupos de estudio: Lo conforman los combatientes y trabajadores civiles, estructurados por unidades y no deben exceder los 30 miembros.

Informe resumen de la actividad de preparación política: requerimiento informativo mensual que recoge datos estadísticos del desarrollo de la preparación política en la institución.

1.2.2 Flujo actual de los procesos involucrados en el campo de acción

En el Organismo Político de la Jefatura Provincial el ciclo de la información referente a la preservación de la memoria histórica se ejecuta a partir de la gestión humana y se almacena en forma (Manuscrita y Digital), casi toda esta información, ya existente, en formato papel un gran volumen de ella y otra parte en CD, DVD y HDD, se emplea para la realización de consultas bibliográficas, publicaciones en la intranet del MININT y documentación de cualquier investigación que necesite de información histórica.

La preparación política tiene sus propias particularidades, se realiza a partir del Plan para el Programa Audiovisual “Hasta la Victoria Siempre” donde se emite por parte del órgano superior (Organismo Político MININT Habana) un documento que contiene todos los temas de estudio a debatir en el año, denominado Programa para la Preparación Política y de manera mensual se envía a la Jefatura Provincial el material necesario para desarrollar estos temas, dígame indicaciones metodológicas y materiales audiovisuales.

La preparación política se efectúa dos veces en el mes (quincenalmente) en todos los grupos de estudio de la provincia. En la primera semana se realiza una actividad metodológica con el Jefe del MININT Provincial y los Jefes de Órgano donde se analizan los temas correspondientes a ese propio mes; fruto de esta actividad pueden o no sufrir cambios las indicaciones metodológicas que se van a tratar después en los grupos de estudio. También la primera semana se ejecuta en la provincia la actividad de preparación y capacitación de todos los monitores de cada grupo.

En la segunda semana estos realizan en su grupo de estudio la actividad correspondiente a la primera quincena, donde se analiza el primer tema del mes sobre el que se genera un debate entre sus miembros, concluida, se conforma el informe resumen de la actividad de preparación política como se muestra en el **Anexo 1**.

En la tercera semana se hace la actividad correspondiente a la segunda quincena analizándose el otro tema de estudio, concluida la actividad se conforma el informe resumen de la actividad de preparación política. Una vez concluidos los dos temas el monitor entrega los informes en la provincia la primera semana del mes siguiente, específicamente el día correspondiente a su preparación y capacitación.

Otro lugar importante en el trabajo del Organismo Político lo ocupan la organización, divulgación y realización de eventos, estas actividades se llevan a cabo a partir de las orientaciones emitidas por el órgano superior (Organismo Político MININT Habana) descritas en el Plan de Trabajo Mensual y en el Plan de Trabajo Anual. Se realizan en todas las unidades de la provincia durante todo el año, al ser muy diversas cada una tiene comportamientos diferentes.

En ellas siempre participan **corresponsales**, trabajadores del Órgano quienes tienen la misión de ir coleccionando todos los reportajes y los archivos multimedia, o sea fotos, videos, audios y documentos vinculados a la temática en cuestión. Luego toda esta información le es entregada al **editor** el cual selecciona la que tenga mejor calidad para archivarla y utilizarla como apoyo para la realización de

otros eventos en el futuro y para la preservación de la memoria histórica de la institución.

1.2.3 Análisis crítico de cómo se ejecutan los procesos

Según entrevistas realizadas al personal que interviene en estos procesos del Organismo Político en la Jefatura Provincial, se pudo conocer que no explotaban las facilidades que brindan las TICs para agilizar los procesos necesarios.

El hecho de que los procesos de preservación de la memoria histórica, preparación política y la realización de eventos se almacenen fundamentalmente en formato papel y en documentos digitales, por diferentes personas, con distintos modos de organización, implica que sea extremadamente trabajoso la búsqueda oportuna, se incurra en errores humanos y en algunas ocasiones, en la pérdida de tiempo por parte de los trabajadores empleando un mayor esfuerzo a la hora de llevar a cabo cualquier actividad. Además, no se cuenta con una forma estándar para la entrega de reportes con estadísticas o información que se solicita en ocasiones por los jefes superiores y la información que se almacena en formato papel, en algunos casos se encuentra un tanto deteriorada, por lo que se hace necesario digitalizarla.

Dentro de las dificultades que aún persisten en el órgano se puede mencionar también que, no están disponibles de una manera factible, los materiales audiovisuales sobre la preparación política para que el personal se mantenga al día sobre estos temas vitales en la formación de los oficiales del MININT.

1.3 Antecedentes de la investigación

Para realizar esta investigación se tuvo en cuenta experiencias similares efectuando búsquedas en páginas web a nivel nacional en todas las instituciones que tributan al MININT con el objetivo de indagar sobre la existencia de algún software que aborde el tema de la gestión de información referente a la preservación de la memoria histórica, la realización de eventos y la preparación política, aunque no se encontró ninguno que guardase relaciones directas con la presente investigación facilitaron su comprensión y generaron nuevas ideas a poner en práctica.

Se encontró un sitio a nivel nacional para el apoyo del trabajo político-ideológico del Organismo Político denominado **“La luz de la tropa”**, pero este no cumple con las expectativas del software que se necesita ya que solo se emplea para publicar noticias sobre el quehacer diario a nivel de país y almacenar documentos de forma temporal. (MININT, La luz de la tropa, 2018)

En la intranet provincial del MININT de Matanzas también existe un sitio en apoyo al trabajo político-ideológico con la misma estructura llamado **“Organismo Político Matanzas”**, este solo contiene vínculos a otros repositorios de documentos estáticos y a algunos de sus departamentos, donde no existe una retroalimentación entre la nueva información que se va generando del trabajo diario y el sitio, puesto que este es imposible de actualizar. En el único aspecto que se mantiene actualizado es en lo informativo sobre el acontecer nacional e internacional nutriéndose para ello de **Cuba Debate** y de **La luz de la tropa**. (MININT, Organismo Político Matanzas, 2018)

Dentro del propio MININT Provincial también existe una aplicación de escritorio que en estos momentos ya no se utiliza denominada **“Repositorio de videos”** que al ser una solución de escritorio no se adapta al contexto actual de la institución en el empleo de sistemas multiplataforma y alternativas basadas en la web, además que solo contiene las ubicaciones donde se encuentran archivados los videos y no los archivos físicamente.

Los anteriores software, no cumplen con los requisitos no funcionales referidos a las tecnologías y herramientas que demanda el MININT en este momento; además, es una línea de trabajo que la institución produzca sus propios software para garantizar la compartimentación necesaria, a partir de la información sensible que se maneja, evitando la vulnerabilidad que refiere la implementación de software creados por expertos ajenos a la entidad capaces de alterar información secreta, por lo cual no se tuvieron en cuenta en la revisión de antecedentes software a nivel internacional. Después de analizados los software anteriores y expuestas sus funcionalidades podemos resumir que actualmente el Organismo Político del MININT de Matanzas no cuenta con una herramienta web para la gestión de

información referente a la preservación de la memoria histórica, la realización de eventos y la preparación política ya que ninguno de los software analizados divulga ni promueve la labor político-ideológica que desarrolla el MININT de Matanzas.

1.4 Métodos de la investigación

1.4.1 Métodos teóricos empleados

- ❖ **Análisis Histórico-lógico:** permitió esclarecer las etapas de desarrollo histórico del problema a solucionar y cómo funciona la gestión de información en el organismo político, su definición, características, evolución y desarrollo.
- ❖ **Análisis-síntesis:** fue precisado durante la revisión bibliográfica y el análisis de resultados, permitiendo descomponer lo complejo en sus partes y cualidades, la división mental del todo en sus múltiples relaciones para luego unir las partes analizadas, descubrir las relaciones y características generales entre ellas.
- ❖ **Inducción-deducción:** su uso fue necesario tanto en la revisión bibliográfica, como en el análisis de los resultados, permitiendo arribar a conclusiones que se infirieron a partir de propiedades y relaciones existentes entre los elementos que conforman el fenómeno objeto de estudio.
- ❖ **Enfoque de sistema:** proporcionó la orientación general del estudio como una realidad integral formada por elementos que interaccionan unos con otros.

1.4.2 Métodos empíricos empleados

- ❖ **La observación:** acompañó la investigación desde los primeros momentos, a través de la cual se conoció el estado y proceder de todo el proceso de gestión de la información en el Organismo Político del MININT, y se obtuvo la información primaria acerca de los objetos investigados.
- ❖ **La entrevista:** aportó datos esenciales a la investigación. Permitted conocer cuáles eran las inquietudes y sugerencias por parte del personal especializado en el tema y qué hacer para facilitar y agilizar su trabajo. Fue útil en distintos momentos de la investigación; fundamentalmente al inicio, cuando se realizó el levantamiento de requisitos para efectuar una exploración preliminar del problema a investigar.

- ❖ **Análisis de documentos:** incluye la revisión de documentos como son reglamentos y disposiciones vigentes relacionadas con el tema. También se consultaron libros y diversos artículos de bancos de información de la web a nivel nacional e internacional relacionados con el tema a investigar.

1.5 Herramientas, tecnologías y metodologías de desarrollo

Para el desarrollo de esta investigación, se hace necesario el estudio y selección de herramientas, tecnologías y metodologías para el desarrollo de aplicaciones web, con arquitectura cliente-servidor y la arquitectura modelo-vista-controlador dadas las grandes posibilidades que brindan, con el propósito de darle cumplimiento al objetivo general. A continuación, se describen aspectos a tener en cuenta.

1.5.1 Metodologías de desarrollo

Las metodologías de desarrollo de **software** son un conjunto de pasos y procedimientos que deben seguirse para llevar a cabo el desarrollo de **software** con calidad. Estas brindan un conjunto de detalles organizativos añadiendo rigurosidad y normas permitiendo que los integrantes de un equipo de desarrollo puedan seguir un criterio común a la hora de realizar las tareas durante el desarrollo de un **software**. La constante innovación tecnológica hace que cada vez sea más necesaria la aplicación de nuevas metodologías adaptadas a los nuevos tiempos. (Armas, 2015)

Actualmente existen dos grandes grupos de metodologías de desarrollo, las **metodologías tradicionales** y las **metodologías ágiles**. Las primeras se centran en el uso exhaustivo de documentación durante todo el ciclo de vida del proyecto, mientras que las segundas dan mayor importancia a la capacidad de respuesta a los cambios. A continuación, se presenta una breve comparación entre ellas.

Metodologías ágiles dan mayor valor a la colaboración con el cliente y al desarrollo incremental del **software** con iteraciones muy cortas. Se basa en la filosofía de que es más importante desarrollar **software** que funcionen, que conseguir una buena documentación y es más importante responder ante un cambio, que seguir estrictamente un plan.

Metodologías tradicionales llevan un control estricto del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, y

las herramientas y notaciones que se usarán. Centran su atención en llevar una documentación exhaustiva de todo el proyecto y en cumplir con un plan de proyecto. En la siguiente tabla se muestra una comparación entre metodologías ágiles y tradicionales. (Muñiz, 2015)

Metodología Ágil	Metodología Tradicional
Pocos artefactos	Más artefactos
Pocos roles, más genéricos y flexibles	Más roles, más específicos
No existe un contrato tradicional, debe ser bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (< 10 integrantes) y trabajando en el mismo sitio	Aplicables a proyectos de cualquier tamaño, pero suelen ser especialmente efectivas/usadas en proyectos grandes y con equipos posiblemente dispersos
La arquitectura se va definiendo y mejorando a lo largo del proyecto	Se promueve que la arquitectura se defina tempranamente en el proyecto
Se hace énfasis en los aspectos humanos: el individuo y el trabajo en equipo	Se hace énfasis en la definición del proceso: roles, actividades y artefactos
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparadas para cambios durante el proyecto	Cierta resistencia a los cambios

Tabla 1: Comparación entre metodologías ágiles y no ágiles. (Muñiz, 2015)

Después de haber realizado una comparación entre las metodologías ágiles y las tradicionales (**Tabla 1**) se observa que las ágiles se adecuan más para guiar el desarrollo del software propuesto, principalmente porque el proyecto no es

altamente complejo, se cuenta con pocos roles y no se dispone de mucho tiempo para su puesta en marcha.

Dentro de las metodologías ágiles más conocidas se encuentran: *Crystal Methodologies*, *Scrum* y *Extreme Programming (XP)*.

Fundamentación de la metodología de desarrollo de software utilizada: XP (en inglés *Extreme Programming*)

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo del software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en la realimentación continua entre el cliente y el equipo de desarrollo, la comunicación fluida entre todos los participantes, la simplicidad en las soluciones implementadas y el coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. (Amaro, 2015)

Las principales características de la programación extrema son las siguientes: (Amaro, 2015)

- ❖ Desarrollo iterativo e incremental
- ❖ Los requerimientos se expresan como escenarios (también conocidos como historias de usuarios), los cuales se dividen e implementan como una serie de tareas
- ❖ Se escriben pruebas unitarias para cada tarea. No se considerará una unidad de código como correcto hasta que no pase todos los test escritos. Se recomienda escribir las pruebas antes que el código de la tarea
- ❖ Los desarrolladores trabajan en parejas. Se da prioridad a un código de mayor calidad, discutido y revisado desde dos puntos de vista distintos, que a una mayor productividad

- ❖ Integración del equipo de programación con el cliente. Se recomienda que un representante del cliente trabaje de forma conjunta con el equipo de desarrollo
- ❖ Refactorización del código, es decir, reescribir parte del código para aumentar su legibilidad, mantenibilidad y capacidad de reutilización
- ❖ Propiedad compartida del código, es decir, que la responsabilidad del desarrollo de cada módulo se divide entre varios grupos de trabajo, de forma que cualquier persona del equipo pueda modificar e introducir mejoras en cualquier parte; debido a esto la mayoría de los errores se acaban detectando
- ❖ Comunicación. Es una premisa de la filosofía de desarrollo ágil, una buena comunicación entre los integrantes de un equipo de desarrollo, esto evita malentendidos que puedan afectar posteriormente al desarrollo del proyecto

Justificación de su uso en el proyecto

En el desarrollo de este proyecto se ha decidido utilizar como metodología de desarrollo de software XP para darle solución a la propuesta, ya que al ser una metodología ágil, se caracteriza por su sencillez tanto en su aprendizaje, como en su aplicación. Es adecuada para cualquier tipo de proyecto y para equipos de desarrollo de cualquier tamaño. XP es de todas las metodologías ágiles citadas, con la que más se ha trabajado y con la que más experiencia se ha adquirido por parte del autor de este trabajo.

1.5.2 Tendencias tecnológicas a considerar

Arquitectura de Software

La arquitectura de software es el diseño de más alto nivel en la estructura de un sistema (Medina, 2017); este va más allá de los algoritmos y estructuras de datos de la computación. Las arquitecturas para la construcción de la propuesta de solución son descritas en este acápite.

Arquitectura Cliente-Servidor

La arquitectura cliente servidor es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. (Dueñas, 2014)

Es la tecnología que proporciona al usuario final el acceso transparente a las aplicaciones, datos, servicios de cómputo o cualquier otro recurso del grupo de trabajo y/o, a través de la organización, en múltiples plataformas. El modelo soporta un medio ambiente distribuido en el cual los requerimientos de servicios hechos por estaciones de trabajo inteligente o clientes, resultan en un trabajo realizado por otros computadores llamados servidores. (Pérez F. F., 2015)

Entre las principales características de la arquitectura Cliente-Servidor, se pueden destacar las siguiente (Aballi, 2016)

- ❖ El servidor presenta a todos sus clientes una interface única y bien definida.
- ❖ El cliente no necesita conocer la lógica del servidor, sólo su interface externa.
- ❖ El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.

Aplicación Web

En la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación de software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador. Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs, tiendas en línea y la propia Wikipedia que son ejemplos bastante conocidos de aplicaciones web. (Perera, 2017)

Patrones de diseño

Los patrones de diseño son un conjunto de reglas que describen como afrontar tareas y solucionar problemas que surgen durante el desarrollo del software. Estos identifican y especifican abstracciones que van más allá del simple ámbito de clases e instancias, o componentes. Para que una solución sea considerada un patrón debe poseer ciertas características como son efectividad habiendo resuelto problemas similares en ocasiones anteriores y reusabilidad permitiendo su aplicación a diferentes problemas de diseño en distintas circunstancias. (García D. C., Aplicación informática para gestión de reservas y estancias Hostal Azul Matanzas., 2016)

En este proyecto el autor decidió utilizar el patrón de diseño Modelo-Vista-Controlador el cual se explica con detalle a continuación.

Modelo-Vista-Controlador (MVC)

Es un patrón de arquitectura de software utilizado para implementar sistemas donde se requiere el uso de interfaces de usuario. Surge de la necesidad de crear un software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos. Su fundamento es la separación del código en tres capas diferentes, acotadas por su responsabilidad, en lo que se llaman Modelos, Vistas y Controladores. (Álvarez M. Á., Desarrolloweb.com, 2014)

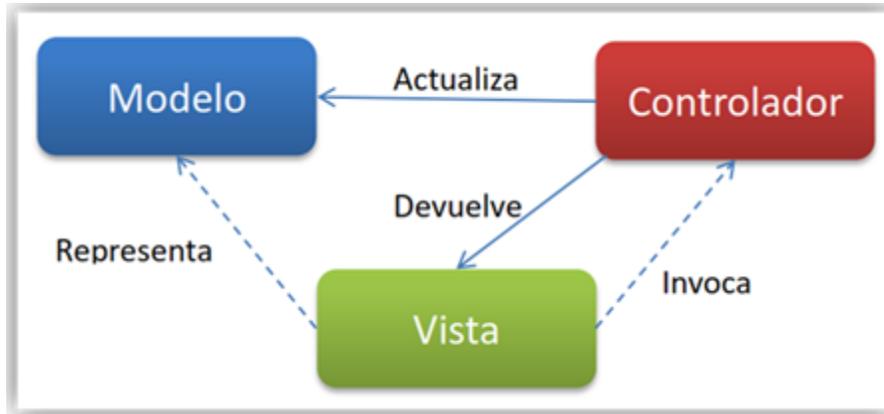


Figura 1: Arquitectura Modelo-Vista-Controlador

El **Modelo**: es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones del software (lógica de negocio). Envía a la vista aquella parte de la información que en cada momento se le solicita para que sea mostrada (típicamente a un usuario). Las peticiones de acceso o manipulación de información llegan al modelo a través del controlador.

La **Vista**: contiene el código del software que va a producir la visualización de las interfaces de usuario, o sea, el código que permite mostrar los estados del software en HTML.

El **Controlador**: es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestro software. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo.

1.5.3 Tecnologías y Herramientas

Lenguajes de Programación utilizados

Un lenguaje es un sistema de comunicación estructurado para el que existe un contexto de uso y ciertos principios combinatorios formales. Sus símbolos primitivos, y reglas para unir esos símbolos, están formalmente especificados, y consta de una semántica formal que puede interpretar y dar significado a las fórmulas bien formadas de este.

Del lado del cliente:

Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento, en el proyecto se emplearon las siguientes tecnologías:

HTML (*Hypertext Markup Language*)

Es el lenguaje de marcado más utilizado para crear páginas web (Campos, 2015). Define una estructura básica y un código, denominado código HTML, para la definición de contenido de la página, como textos, imágenes, sonidos, etc. No es un lenguaje de programación, es un lenguaje de marcado que incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación. Los documentos HTML se encuentran estrictamente organizados. Cada parte del mismo está diferenciado por etiquetas específicas.

CSS Hojas de Estilo en Cascada (*Cascading Style Sheets*)

Este lenguaje es un componente desarrollado para superar las limitaciones y reducir la complejidad de HTML. CSS ha crecido y ganado importancia, pero siempre desarrollado en paralelo, enfocado en las necesidades de los diseñadores y apartado del proceso de evolución de HTML. La versión 3 de CSS sigue el mismo camino, pero esta vez con un mayor compromiso. La especificación de HTML5 fue desarrollada considerando CSS a cargo del diseño. Debido a esta consideración, la integración entre HTML y CSS es ahora vital para el desarrollo web y esta es la razón por la que cada vez que se menciona HTML5 también se está haciendo referencia a CSS3, aunque oficialmente se trate de dos tecnologías completamente separadas. (Fernández A. G., 2017)

JavaScript

Lenguaje de programación interpretado, es decir, que no requiere compilación. Es utilizado principalmente en la creación de páginas web. Este lenguaje orientado a objetos es imperativo, débilmente tipado y dinámico. Actualmente existen numerosas bibliotecas de JavaScript como *jQuery*, *ExtJS*, *Midori*, etc. Según Campos, *Javascript* tiene todo el poder necesario para proveer dinamismo y construir aplicaciones web completamente funcionales, además de considerarlo la mejor opción para la web. (Campos, 2015)

Ajax

Ajax son las siglas de *Asynchronous Java Script And XML*. Es una técnica de desarrollo web para lograr aplicaciones más interactivas, permitiendo actualizar parte de una página con información que se encuentra en el servidor sin tener que refrescar completamente la página. De modo similar podemos enviar información al servidor. Algunas de sus ventajas son: utiliza tecnologías ya existentes; soportada por la mayoría de los navegadores modernos; interactividad ya que el usuario no tiene que esperar hasta que lleguen los datos del servidor; portabilidad y mayor velocidad debido a que no hay que recargar toda la página nuevamente. (Romero, 2014)

Del lado del servidor:

C#

Aunque es posible utilizar muchos lenguajes de programación en .Net, C# es el único diseñado específicamente para ser utilizado en esta plataforma, por lo que se define como el lenguaje nativo de .Net. C# toma los mejores aspectos de los lenguajes preexistentes, tales como C++, *Java* o *Visual Basic* y los combina, creando uno solo. Algunas de las principales características del lenguaje, que también tienen repercusión en la plataforma .NET se enumeran a continuación (Santiago, 2017):

- ❖ Sencillez: Elimina los elementos de otros lenguajes que son innecesarios en .Net.
- ❖ Modernidad: Incorpora elementos útiles que no tienen otros lenguajes preexistentes.
- ❖ Orientación a objetos: Uso de objetos y sus interacciones para el diseño de aplicaciones.
- ❖ Orientación a componentes: Permite definir propiedades, eventos y atributos.
- ❖ Eficiente: Incluye numerosas restricciones en todo el código para asegurar su seguridad.

Justificación para su uso en el proyecto

El autor determinó para el desarrollo de la solución propuesta, el lenguaje de programación **C#** por ser multiplataforma y debido a sus características se hace posible que el cliente interactúe con una página rápida, eficiente y segura; capaz de mostrar y procesar información de forma más óptima.

Unido a lo anterior, se puede mencionar que es muy sencillo y legible, y cuenta con una amplia librería de funcionalidades, que permiten hacer cualquier tipo de operación, como trabajo con archivos y carpetas, procesamiento de formularios, paginación de resultados y muchas otras opciones. Esta amplia librería concede la posibilidad de realizar un producto de acuerdo a las necesidades del usuario.

Además, **C#** es, de todos los Lenguajes de Programación estudiados en la carrera, con uno de los que más se ha trabajado y experiencia se ha adquirido, igualmente aclarar que es el lenguaje de programación que se utiliza en el MININT.

IDE de desarrollo: Visual Studio 2012

Microsoft Visual Studio 2012 es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación, tales como *Visual C++*, *Visual C#*, *Visual J#*, y *Visual Basic .NET*, al igual que entornos de desarrollo web como *ASP.NET*, aunque

actualmente se han desarrollado las extensiones necesarias para muchos otros. (Microsoft Corporation, 2015)

Visual Studio permite a los desarrolladores crear sistemas, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión .NET 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles.

En este proyecto se utiliza Microsoft Visual Studio 2012 debido a su amplia documentación y facilidad en el trabajo con la base de datos.

Framework

Un framework constituye una estructura de soporte definida (o plataforma de desarrollo enfocada al entorno web) en la cual otro proyecto de software puede ser organizado y desarrollado. (Alcalde A. , 2017)

Frameworks utilizados

Lado cliente

Bootstrap

Bootstrap, es un *framework* originalmente creado por *Twitter*, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Esta técnica de diseño y desarrollo se conoce como *responsive design* (en inglés) o diseño adaptativo. (García L. C., 2015)

El beneficio de usar *responsive design* en un sitio Web, es principalmente que el sitio Web se adapta automáticamente al dispositivo desde donde se acceda. Lo que se usa con más frecuencia es el uso de *Media Queries* (consultas), que es un módulo de CSS3 que permite la representación de contenido para adaptarse a condiciones como la resolución de la pantalla y si trabajas las dimensiones de tu contenido en porcentajes, puedes tener una Web muy fluida capaz de adaptarse a casi cualquier tamaño de forma automática. (Bermúdez, 2017)

Lado servidor

Entity Framework

Entity Framework (EF) está basado sobre la plataforma de ADO.NET mediante la cual genera un conjunto de objetos que están directamente ligados a una Base de Datos (BD). Posibilita el manejo de dichos objetos en lugar de utilizar Lenguaje de Consulta Estructurado (SQL) contra la BD (Acosta, 2015). Esto representa la gran ventaja de EF, hacer transparente la BD con la que se trabaja. El modelo de este framework genera las sentencias SQL nativas requeridas para cada Sistema Gestor de Bases de Datos (SGBD). Permite además intercambiar un SGBD por otro de ser necesario.

Existen tres formas de trabajar los datos con **EF**: *Database First* (Primero la Base de Datos), *Model First* (Primero el Modelo) y *Code First* (Primero el Código) (Medina, 2017):

- ❖ **Database First.** desde una BD creada se genera un modelo de datos que consiste en clases y propiedades que corresponden a los objetos de la BD.
- ❖ **Model First.** se crea un modelo de datos y se genera un Lenguaje de Definición de Datos (DDL) para crear una BD.
- ❖ **Code First.** se puede tener una BD creada o no. Se crean clases donde sus propiedades son las columnas de una tabla es una BD.

Mapeo Objeto-Relacional (ORM):

El mapeo objeto-relacional es una técnica de programación para convertir datos del sistema de tipos utilizado en un lenguaje de programación orientado a objetos al utilizado en una BD relacional. En la práctica esto crea una BD virtual orientada a objetos sobre la BD relacional. Esto posibilita el uso de las características propias de la orientación a objetos (esencialmente la herencia y el polimorfismo).

Las BD relacionales solo permiten guardar tipos de datos primitivos (enteros, cadenas de texto, etc.) por lo que no se pueden guardar de forma directa los objetos del software en las tablas, sino que estos se deben de convertir antes en registros, que por lo general afectan a varias tablas. En el momento de volver a recuperar los datos, hay que hacer el proceso contrario, se deben convertir los registros en objetos. Es entonces cuando ORM cobra importancia, ya que se encarga de forma automática de convertir los objetos en registros y viceversa, simulando así tener una BD orientada a objetos. (Guardado, 2013)

Entre las ventajas que ofrecen los ORM se encuentran: rapidez en el desarrollo, abstracción de la BD, reutilización, seguridad, mantenimiento del código, lenguaje propio para realizar las consultas (Yanes Enriquez & Gracia del Busto, 2013).

Gestor de base de datos: Oracle 11g

Oracle es una herramienta cliente/servidor para la gestión de BD la cual se identifica por el empleo de la administración gráfica intuitiva. Se considera uno de los sistemas gestores de BD más completos y de alto rendimiento utilizados en la actualidad, destacando su soporte de transacciones, su estabilidad, escalabilidad y su completa seguridad en el entorno de producción, de pruebas, el acceso a los sistemas y aplicaciones, así como la gestión de copias de seguridad.

Este ha sido diseñado para que se pueda adaptar a las necesidades cambiantes del negocio donde las organizaciones puedan controlar y gestionar grandes volúmenes de contenidos no estructurados en un único repositorio con el objetivo de reducir los costes y los riesgos asociados a la pérdida de información. Esta herramienta hace que el proceso de recopilación de datos sea más completo, factible y sencillo. Utiliza estándares de la industria de código abierto para ayudar a que las funcionalidades estén creadas específicamente para la realización de aplicaciones y sistemas. (Arthur, 2015)

Justificación para su uso en el proyecto

Se eligió Oracle 11g, porque es uno de los SGBD más utilizado a nivel mundial, por su robustez y su seguridad. Es un sistema de manejo de BD con un gran nivel de

facilidad de desarrollo que se integra cómodamente con el lenguaje de programación C#. Dispone, además, de una arquitectura que lo hace extremadamente rápido y fácil de personalizar; además se adecúa perfectamente a las exigencias del cliente.

Lenguaje de Consulta: LinQ. Expresiones Lambda.

LINQ o *Language Integrated Query* es un **lenguaje de consultas** creado para facilitar la explotación de los datos sin importar el tipo de fuente de datos utilizada. Permite consultar información en tecnologías tan diferentes como ficheros XML, BD relacionales o colecciones fuertemente tipadas. Extiende el lenguaje a través de las llamadas expresiones de consulta, que son parecidas a las sentencias SQL. LINQ además define un conjunto de nombres de métodos (llamados operadores de consulta estándar), y un conjunto de reglas de traducción, que son usadas por el compilador para traducir las expresiones de consulta en expresiones normales del lenguaje (Torres & Paguay Soxo, dspace, 2014).

Expresiones lambda: son un conjunto de instrucciones capaces de retornar un valor partiendo de los parámetros que se les suministra, aunque en determinados casos es posible que no reciba ningún parámetro, o que realicen una acción sin retornar nada. Igual que una función tradicional. Con estas expresiones podemos llamar directamente a funciones *where*, *join*, *select*, directamente desde el objeto. (Torres & Paguay Soxo, dspace, 2014)

Las expresiones lambda pueden ser usadas en escenarios tales como:

- ❖ Argumentos de otras funciones
- ❖ Tipo de retorno de una función
- ❖ Expresiones LINQ
- ❖ Asignación a instancias de delegados genéricos

Expresiones Regulares

Una expresión regular, es un patrón que describe una cadena de caracteres, esta descripción no enumera a ningún elemento. Es una manera de representar a los lenguajes regulares utilizando caracteres del alfabeto sobre el cual se define el lenguaje. Específicamente, las expresiones regulares se construyen utilizando los operadores unión y concatenación. Además cada expresión regular tiene un autómata finito asociado. Se emplean en muchos tipos comunes de software, incluyendo herramientas para la búsqueda de patrones, en textos o en nombres de archivo.

Por ejemplo, el grupo formado por las cadenas Handel, Händel y Haendel se describe con el patrón "H(a|ä|ae)ndel" (Hopcroft, Motwani, & Ullman, 2017).

Tienen diversas aplicaciones en problemas reales tales como:

- ❖ Facilitar la construcción de un compilador
- ❖ Validar la sintaxis de un programa
- ❖ Algunos editores de texto y programas similares permiten la sustitución de una cadena por otra cualquiera que cumpla con la expresión regular dada
- ❖ Como correctores ortográficos
- ❖ Para encontrar patrones en el genoma humano

1.6 Conclusiones del Capítulo

Luego de analizar el estado actual de la temática y sus antecedentes, así como las principales técnicas y recursos utilizados, se concluye que:

- ❖ El problema planteado debe ser resuelto con un adecuado uso de las TIC.
- ❖ En la bibliografía consultada no se encontró ningún software que dé solución al problema planteado.
- ❖ Se determinaron los elementos esenciales del proceso para la gestión de la información en el Organismo Político del MININT Provincial de Matanzas.
- ❖ Se determinó utilizar como marco de trabajo Entity framework, C# como lenguaje de programación, Oracle 11g como SGBD, XP como metodología de desarrollo de software, MVC como patrón de diseño y la arquitectura Cliente-Servidor.
- ❖ Se demostró la necesidad de realizar un software que permita la solución eficiente de la problemática existente.
- ❖ En sentido general se ha favorecido a una mejor comprensión del objeto de estudio y se han establecido las bases para las siguientes fases de la investigación.

Por lo expuesto en el análisis comparativo de los sistemas existentes y por el estudio de esta problemática en otras divisiones de la Jefatura Provincial del MININT, queda demostrada la falta de un software de apoyo a la gestión de la información en el Organismo Político del MININT Provincial de Matanzas.

Capítulo 2: Análisis, diseño y desarrollo de la solución propuesta

2.1 Introducción

El desarrollo de aplicaciones web y de software conlleva una ardua labor de análisis y diseño para lograr cumplir las expectativas del cliente de forma tal que el producto obtenido tenga la calidad requerida.

La metodología de desarrollo de software que se emplea en esta investigación es la Programación Extrema (XP), en el capítulo se exponen los elementos necesarios para la descripción de la solución propuesta. A través de las Historias de Usuarios (HU) que acumulan las necesidades existentes definidas por el cliente, es llevado a cabo el análisis de los requerimientos, además apoyándose en esta metodología también se realiza la planificación inicial del proyecto.

2.2 Descripción de la solución

Se propone el desarrollo de un software que sirva de apoyo a la gestión de la información en el Organismo Político del MININT Provincial de Matanzas.

Los usuarios que interactúan con el software tendrán los siguientes privilegios:

- ❖ **Administrador:** tiene la posibilidad de asignar permisos, gestionar los usuarios que pueden acceder al software, así como editar las contraseñas y crear sus roles.
- ❖ **Editor:** se encarga de gestionar la información que se publicará. Entiéndase gestionar como revisar, insertar, editar, eliminar y publicar.
- ❖ **Corresponsal:** tiene la posibilidad de insertar información al sitio.
- ❖ **Activista o Monitor:** tiene la posibilidad de insertar la información referente al proceso de Preparación Política luego de concluidas las actividades de preparación política en cada grupo de estudio.
- ❖ **Invitado:** tiene la posibilidad de consultar, descargar información y realizar comentarios.

Para garantizar la seguridad y la confiabilidad en la información que se procesa es muy importante implementar un software que permita la autenticación para identificar los usuarios que utilizan el programa con roles de Administrador, Editor, Corresponsal y Activista; los Usuarios Funcionales sólo tendrán que loguearse por

motivos de seguridad a la hora de realizar comentarios y al inscribirse en los eventos.

2.3 Etapa de planificación

La planificación se realiza con el objetivo de lograr una eficiente organización del prototipo inicial del problema y proporcionar así un buen comienzo a una solución eficaz. Con este objetivo y según las ideas del cliente sobre el software se desarrollarán las HU, mediante las cuales se obtendrá un punto de partida para el resto de la planificación del proyecto. Igualmente se realizará un estimado de cada una de las entregas del proyecto y del tiempo, basándose en que la planificación inicial se podría afectar debido a cambios que pudiesen sufrir estos aspectos durante el desarrollo del proyecto.

2.3.1 Equipo de trabajo y roles

En la metodología XP se plantean una serie de principios que se ponen en práctica a lo largo del ciclo de desarrollo, para ello se evidencia el uso de una especie de “equipo de desarrollo” que debe cumplir algunos roles ya establecidos en el proceso. A continuación, se definen en la **Tabla 2** los roles del equipo de trabajo para el desarrollo del software, y los miembros involucrados en el desarrollo del mismo:

Miembros	Roles XP
María Caridad Díaz Mesa	Cliente, Consultor
Geykel Alberto Hernández Cruz	Programador, Encargado de pruebas
Ernesto Henríquez Rivero	Jefe del Proyecto

Tabla 2: Equipo de trabajo y roles

2.3.2 Historias de usuarios iniciales

Las HU son la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el software debe poseer, sean requisitos funcionales o no funcionales.

Son utilizadas para estimar tiempos de desarrollo de la parte del software que describen. También se utilizan en la fase de pruebas, para verificar si el software cumple con lo que especifica la historia de usuario. Las HU ayudan en la

comunicación entre el cliente y los desarrolladores y pueden ir cambiando a medida que avanza el proyecto y que el cliente vea nuevas posibilidades y soluciones. Cuando llega la hora de implementar una historia de usuario, el cliente y los desarrolladores se reúnen para concretar y detallar lo que tiene que hacer dicha historia.

Según (Fuentes, 2015) para el establecimiento de las historias se utilizan dos escalas nominales que exponen tres categorías alta, media y baja las cuales significan el riesgo y la prioridad en la escala de riesgo y prioridad respectivamente.

A continuación se muestran las escalas equivalentes a la prioridad en el negocio:

Alta: asignada a las HU que corresponden a funcionalidades esenciales en el desarrollo del proyecto, a las que el cliente define como primordiales.

Media: dada a las HU que resultan para el cliente como funcionalidades a tener en cuenta, sin que estas tengan una afectación directa sobre el proyecto que se esté desarrollando.

Baja: se le otorga a las HU que constituyen funcionalidades que sirven de ayuda al control de elementos asociados al equipo de desarrollo, a la estructura y no tienen nada que ver con el proyecto en desarrollo.

Escala Nominal de Riesgo en Desarrollo:

Alta: cuando para la implementación de la HU se considera la posible existencia de errores que lleven a inoperatividad del código.

Media: cuando pueden aparecer errores en la implementación de la HU que puedan retrasar la entrega de la versión.

Baja: cuando pueden aparecer errores que serán tratados con relativa facilidad sin que traigan perjuicios para el desarrollo del proyecto.

En la siguiente tabla se muestra un resumen de todas las HU que se plantearon inicialmente.

No	Historia de Usuario	Prioridad	Riesgo	Puntos estimados	Iteración	Entrega
1	Diseño y creación de la base de datos	Alta	Alto	1	1	1
2	Diseño de la interfaz de usuario	Alta	Medio	1	1	1
3	Autenticarse	Alta	Alto	1	1	1
4	Gestionar usuario	Alta	Alto	2	1	1
5	Gestionar rol	Media	Medio	1	1	1
6	Gestionar recurso	Alta	Alto	2	1	1
7	Gestionar álbum	Alta	Alto	2	2	2
8	Gestionar efeméride	Alta	Alto	2	2	2
9	Gestionar biografía	Alta	Alto	2	2	2
10	Gestionar evento	Alta	Alto	2	2	2
11	Gestionar preparación política	Alta	Alto	2	3	3
12	Gestionar informe preparación política	Alta	Alto	2	3	3
13	Gestionar grupo de estudio	Alta	Alto	1	3	3
14	Generar reporte de asistencia	Alta	Medio	1	3	3
15	Generar gráficos de asistencia	Alta	Medio	1	3	3
16	Generar reporte de almacenamiento	Media	Medio	1	3	3
17	Generar búsqueda	Alta	Alto	2	4	4

18	Gestionar comentario	Bajo	Bajo	1	4	4
19	Trazas	Media	Medio	1	4	4
Total				28	4	4

Tabla 3: Historias de Usuarios Iniciales

A continuación se muestra la descripción de las HU de esta investigación.

En la **Tabla 4** se muestra la HU 1: Diseño y Creación de la base de datos

Historia de Usuario	
Número: 1	Usuario: Administrador
Nombre historia: Diseño y creación de la base de datos	
Prioridad en negocio: alta	Riesgo en desarrollo: alto
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Se diseña e implementa la base de datos en el gestor Oracle 11g.	
Observaciones:	
<ul style="list-style-type: none"> ❖ Verificar si a la base de datos se le estableció una contraseña. ❖ Se debe verificar la integridad de los datos. 	

Tabla 4 HU 1: Diseño y Creación de la base de datos

En la **Tabla 5** se muestra la HU 4: Gestionar usuario

Historia de Usuario	
Número: 4	Usuario: Administrador
Nombre historia: Gestionar usuario	
Prioridad en negocio: alta	Riesgo en desarrollo: alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Geykel Alberto Hernández Cruz	

<p>Descripción: Permite al administrador listar, insertar, editar y eliminar usuarios que puedan acceder al software, así como editar las contraseñas y sus roles.</p>
<p>Observaciones:</p> <ul style="list-style-type: none"> ❖ Estas operaciones solo las podrá realizar aquel usuario que tenga el rol de administrador. ❖ Cuando se va a insertar un nuevo usuario se comprueba que este exista a través del servicio web SUIN. ❖ Se debe verificar la integridad de los datos.

Tabla 5 HU 4: Gestionar usuario

En la **Tabla 6** se muestra la HU 6: Gestionar recurso

Historia de Usuario	
Número: 6	Usuario: Administrador
Nombre historia: Gestionar recurso	
Prioridad en negocio: alta	Riesgo en desarrollo: alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Insertar, mover, eliminar y ver recursos (imágenes, videos, audios y documentos).	
Observaciones: Solo usuarios con privilegios correspondientes a estas funcionalidades las pueden ejecutar y se debe verificar la integridad de los datos.	

Tabla 6 HU 6: Gestionar recurso

En la **Tabla 7** se muestra la HU 7: Gestionar álbum

Historia de Usuario	
Número: 7	Usuario: Administrador
Nombre historia: Gestionar álbum	

Prioridad en negocio: alta	Riesgo en desarrollo: alto
Puntos estimados: 2	Iteración asignada: 2
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Listar, insertar, editar, eliminar, ver y publicar álbumes.	
Observaciones: Solo usuarios con privilegios correspondientes a estas funcionalidades las pueden ejecutar y se debe verificar la integridad de los datos.	

Tabla 7 HU 7: Gestionar álbum

En la **Tabla 8** se muestra la HU 10: Gestionar evento

Historia de Usuario	
Número: 10	Usuario: Administrador
Nombre historia: Gestionar evento	
Prioridad en negocio: alta	Riesgo en desarrollo: alto
Puntos estimados: 2	Iteración asignada: 2
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Listar, insertar, editar, eliminar, ver, publicar, cerrar eventos y subir bases del evento.	
Observaciones: Solo usuarios con privilegios correspondientes a estas funcionalidades las pueden ejecutar y se debe verificar la integridad de los datos.	

Tabla 8 HU 10: Gestionar evento

En la **Tabla 9** se muestra la HU 11: Gestionar preparación política

Historia de Usuario	
Número: 11	Usuario: Administrador
Nombre historia: Gestionar preparación política	

Prioridad en negocio: alta	Riesgo en desarrollo: alto
Puntos estimados: 2	Iteración asignada: 3
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Listar, insertar, editar, eliminar, ver y publicar los materiales para el desarrollo de las actividades de preparación política.	
Observaciones: Solo usuarios con privilegios correspondientes a estas funcionalidades las pueden ejecutar y se debe verificar la integridad de los datos.	

Tabla 9 HU 11: Gestionar preparación política

En la **Tabla 10** se muestra la HU 14: Generar reporte de asistencia

Historia de Usuario	
Número: 14	Usuario: Administrador
Nombre historia: Generar reporte de asistencia	
Prioridad en negocio: alta	Riesgo en desarrollo: medio
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Se generará a partir de los datos insertados en los informes de cada actividad de preparación política una relación con la asistencia a estas actividades.	
Observaciones: Solo usuarios con privilegios correspondientes a estas funcionalidades las pueden ejecutar y se debe verificar la integridad de los datos.	

Tabla 10 HU 14: Generar reporte de asistencia

En la **Tabla 11** se muestra la HU 16: Generar reporte de almacenamiento

Historia de Usuario	
Número: 16	Usuario: Administrador
Nombre historia: Generar reporte de almacenamiento	

Prioridad en negocio: media	Riesgo en desarrollo: medio
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Se generarán gráficas a partir de la cantidad de archivos, el tamaño que ocupan físicamente en el software y en que partición que se encuentran distribuidos.	
Observaciones: Solo usuarios con privilegios correspondientes a estas funcionalidades las pueden ejecutar y se debe verificar la integridad de los datos.	

Tabla 11 HU 18: Generar reporte de almacenamiento

2.3.3 Planificación de iteraciones

Teniendo ya definidas las HU del software se pasa a realizar la planificación de las iteraciones. Atendiendo a lo mencionado con anterioridad se decide realizar cuatro iteraciones durante el desarrollo del software, en la **Figura 2** se detallan las mismas.



Figura 2: Planificación de las Iteraciones

2.3.4 Reuniones

El planeamiento es esencial para cualquier tipo de metodología, es por ello que XP requiere de una revisión continua del plan de trabajo. A pesar de ser una metodología que evita la documentación exagerada, es muy estricta en la organización del trabajo.

Quincenalmente se realizará una reunión general con todo el equipo de trabajo con el objetivo de ver el progreso de cada iteración y en caso de algún inconveniente se deberá planificar una reunión aparte que deberá ser dentro de dicha quincena.

2.4 Etapa de diseño

En XP solo se diseñan aquellas HU que el cliente ha seleccionado para la iteración actual por dos motivos: por un lado se considera que no es posible tener un diseño completo del sistema y sin errores desde el principio. El segundo motivo es que

dada la naturaleza cambiante del proyecto, el hacer un diseño muy extenso en las fases iniciales del proyecto para luego modificarlo, se considera un desperdicio de tiempo.

Es importante resaltar que esta tarea es permanente durante la vida del proyecto partiendo de un diseño inicial que va siendo corregido y mejorado en el transcurso del proyecto.

2.4.1 Prototipo de interfaz de usuario

En la **Figura 4** se muestra la página principal del software



Figura 3: Página principal del software

2.4.2 Tareas a desarrollar

Cada HU se transformará en Tareas de Ingeniería (TI), por esta razón solo se documentarán las más significativas:

No	Nombre de la Historia de Usuario	Tareas de Ingeniería	Iteración
1	Diseño y creación de la base de datos	1. Diseño de la base de datos 2. Creación de la base de datos	1
2	Diseño de la interfaz de usuario	3. Diseño de la interfaz principal 4. Diseño de otras interfaces	1

3	Autenticarse	5. Autenticarse	1
4	Gestionar usuario	6. Insertar usuario 7. Listar usuario 8. Editar usuario 9. Eliminar usuario	1
5	Gestionar rol	10. Insertar rol 11. Listar rol 12. Eliminar rol 13. Asignar permisos	1
6	Gestionar recurso	14. Insertar recurso 15. Mover recurso 16. Eliminar recurso 17. Ver recurso	1
7	Gestionar álbum	18. Insertar álbum 19. Listar álbum 20. Editar álbum 21. Eliminar álbum 22. Ver álbum 23. Publicar álbum	2
8	Gestionar efeméride	24. Insertar efeméride 25. Listar efeméride 26. Editar efeméride 27. Eliminar efeméride 28. Ver efeméride 29. Publicar efeméride	2
9	Gestionar biografía	30. Insertar biografía 31. Listar biografía 32. Editar biografía 33. Eliminar biografía 34. Ver biografía 35. Publicar biografía	2
10	Gestionar evento	36. Insertar evento 37. Listar evento 38. Editar evento 39. Eliminar evento 40. Ver evento 41. Publicar evento 42. Subir bases 43. Cerrar evento	2
11	Gestionar preparación política	44. Insertar preparación política 45. Listar preparación política 46. Editar preparación política 47. Eliminar preparación política 48. Ver preparación política 49. Publicar preparación política	3

12	Gestionar informe preparación política	50. Insertar informe preparación política 51. Listar informe preparación política 52. Ver informe preparación política 53. Eliminar informe preparación política	3
13	Gestionar grupo de estudio	54. Insertar grupo de estudio 55. Listar grupo de estudio política 56. Eliminar grupo de estudio	3
14	Generar reporte de asistencia	57. Generar reporte de asistencia	3
15	Generar gráficos de asistencia	58. Generar gráficos de asistencia	3
16	Generar reporte de almacenamiento	59. Generar reporte de almacenamiento	3
17	Generar búsqueda	60. Generar búsqueda	4
18	Gestionar comentario	61. Insertar comentario 62. Listar comentario 63. Eliminar comentario 64. Publicar comentario	4
19	Trazas	65. Listar trazas 66. Buscar trazas	4

Tabla 12: Tareas de Ingeniería

A continuación se muestran las TI más esenciales, el resto se podrán encontrar en la documentación del sistema.

En la **Tabla 13** se muestra la TI 2: Creación de la base de datos

Tarea	
Número tarea: 2	Número historia: 1
Nombre tarea: Creación de la base de datos	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Crear en Oracle 11g la base de datos con su respectivas relaciones y la integridad correspondiente entre las tablas.	
Observaciones:	

Tabla 13 TI 2: Creación de la base de datos

En la **Tabla 14** se muestra la TI 6: Insertar usuario

Tarea	
Número tarea: 6	Número historia: 4
Nombre tarea: Insertar usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Se mostrará un formulario donde el usuario con los rol de administrador podrá insertar un nuevo usuario. Estará compuesto por datos como: nombre, nick, rol y contraseña. Los campos del formulario no pueden quedar vacíos, en caso contrario se notificará un mensaje de error.	
Observaciones: Cuando se inserta un nuevo usuario se comprueba que este exista a través del servicio web SUIN.	

Tabla 14 TI 6: Insertar usuario

En la **Tabla 15** se muestra la TI 14: Insertar recurso

Tarea	
Número tarea: 14	Número historia: 6
Nombre tarea: Insertar recurso	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Insertar recurso será una opción que estará presente en la vista de los usuarios con privilegios correspondientes a esta funcionalidad. Al seleccionar la opción (Subir recursos al álbum) aparecerá un formulario en el cual habrá un botón que permitirá adicionar nuevos recursos al álbum seleccionado.	
Observaciones:	

Tabla 15 TI 14: Insertar recurso

En la **Tabla 16** se muestra la TI 18: Insertar álbum

Tarea	
Número tarea: 18	Número historia: 7
Nombre tarea: Insertar álbum	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Insertar álbum será una opción que estará presente en la vista de los usuarios con privilegios correspondientes a esta funcionalidad. Al marcar la opción (Crear álbum) aparecerá un formulario en el cual se podrá adicionar un nuevo álbum, permitiendo insertar una serie de información relacionada con este.	
Observaciones:	

Tabla 16 TI 18: Insertar álbum

En la **Tabla 17** se muestra la TI 36: Insertar evento

Tarea	
Número tarea: 36	Número historia: 10
Nombre tarea: Insertar evento	
Tipo de tarea: Desarrollo	Puntos estimados: 0.25
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Insertar evento será una opción que estará presente en la vista de los usuarios con privilegios correspondientes a esta funcionalidad. Al marcar la opción (Crear evento) aparecerá un formulario en el cual se podrá seleccionar el tipo de evento que se desea adicionar, permitiendo insertar una serie de información relacionada con este.	
Observaciones: Los eventos son temporales por lo que se definirá una fecha límite en la que el evento dejara de estar activo (fecha de admisión límite).	

Tabla 17 TI 36: Insertar evento

En la **Tabla 18** se muestra la TI 44: Insertar preparación política

Tarea	
Número tarea: 44	Número historia: 11
Nombre tarea: Insertar preparación política	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Insertar preparación política será una opción que estará presente en la vista de los usuarios con privilegios correspondientes a esta funcionalidad. Al seleccionar la opción (+Preparación Política) aparecerá un formulario en el cual se podrá adicionar una nueva preparación política, permitiendo insertar una serie de información relacionada con esta.	
Observaciones:	

Tabla 18 TI 44: Insertar preparación política

En la **Tabla 19** se muestra la TI 57: Generar reporte de asistencia

Tarea	
Número tarea: 57	Número historia: 14
Nombre tarea: Generar reporte de asistencia	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Geykel Alberto Hernández Cruz	
Descripción: Se generará a partir de los datos insertados en los informes de cada actividad de preparación política una relación con la asistencia a estas actividades.	
Observaciones:	

Tabla 19 TI 57: Generar reporte de asistencia

2.4.3 Tarjetas de Clase, Responsabilidad y Colaboración

La principal funcionalidad que tienen las tarjetas Clase-Responsabilidad-Colaboración (CRC) es ayudar a dejar el pensamiento procedimental para incorporarse al enfoque orientado a objetos y mostrar las colaboraciones que se establecen entre las clases que componen el software. Cada tarjeta representa una clase con su nombre en la parte superior, en la sección inferior izquierda están descritas las responsabilidades y a la derecha las clases que le sirven de soporte. A continuación se muestran algunas de las tarjetas que se elaboraron.

En la **Tabla 20** se muestra la Tarjeta CRC para la clase: **Usuario**

Tarjeta CRC	
Clase: Usuario	
Superclase: Persona	
Subclase:	
Descripción: En esta clase se guardan los datos relacionados con los diferentes usuarios que interactúan con el software	
Atributos	
Id_usuario	INTEGER
CI	NVARCHAR2(11)
Password	NVARCHAR2(100)
Nick	NVARCHAR2(30)

Tabla 20 Tarjeta CRC: Usuario

En la **Tabla 21** se muestra la Tarjeta CRC para la clase: **Recurso**

Tarjeta CRC	
Clase: Recurso	
Superclase:	
Subclase:	
Descripción: En esta clase se guardan los datos relacionados con los recursos	

Atributos	
Id_recurso	INTEGER
URL	NVARCHAR2(500)
Id_tipo	INTEGER
Palabras_Claves	VARCHAR2(1000)

Tabla 21 Tarjeta CRC: Recurso

En la **Tabla 22** se muestra la Tarjeta CRC para la clase: **Álbum**

Tarjeta CRC	
Clase: Álbum	
Superclase:	
Subclase:	
Descripción: En esta clase se guardan los datos relacionados con los álbumes	
Atributos	
Id_album	INTEGER
Nombre	NVARCHAR2(100)
Fecha_creac	DATE
Descripción	CLOB
Palabras_Claves	VARCHAR2(1000)
Publicado	VARCHAR2(10)

Tabla 22 Tarjeta CRC: Álbum

En la **Tabla 23** se muestra la Tarjeta CRC para la clase: **Evento**

Tarjeta CRC	
Clase: Evento	
Superclase:	
Subclase:	

Descripción: En esta clase se guardan los datos relacionados con los eventos	
Atributos	
Id_evento	INTEGER
Tema	VARCHAR2(200)
Fecha_Realización	DATE
Estado	VARCHAR2(50)
Publicado	VARCHAR2(20)
Tipo	VARCHAR2(20)

Tabla 23 Tarjeta CRC: Evento

En la **Tabla 24** se muestra la Tarjeta CRC para la clase: **Preparación Política**

Tarjeta CRC	
Clase: Preparación Política	
Superclase:	
Subclase:	
Descripción: En esta clase se guardan los datos relacionados con preparaciones políticas	
Atributos	
Id_PP	INTEGER
Tema	NVARCHAR2(100)
Quincena	NVARCHAR2(100)
Mes	VARCHAR2(20)
Año	INTEGER
Publicado	NVARCHAR2(20)

Tabla 24 Tarjeta CRC: Preparación Política

2.4.4 Modelo físico de la base de datos

La **Figura 5** muestra el modelo físico de la base de datos, así como la dependencia entre sus clases.

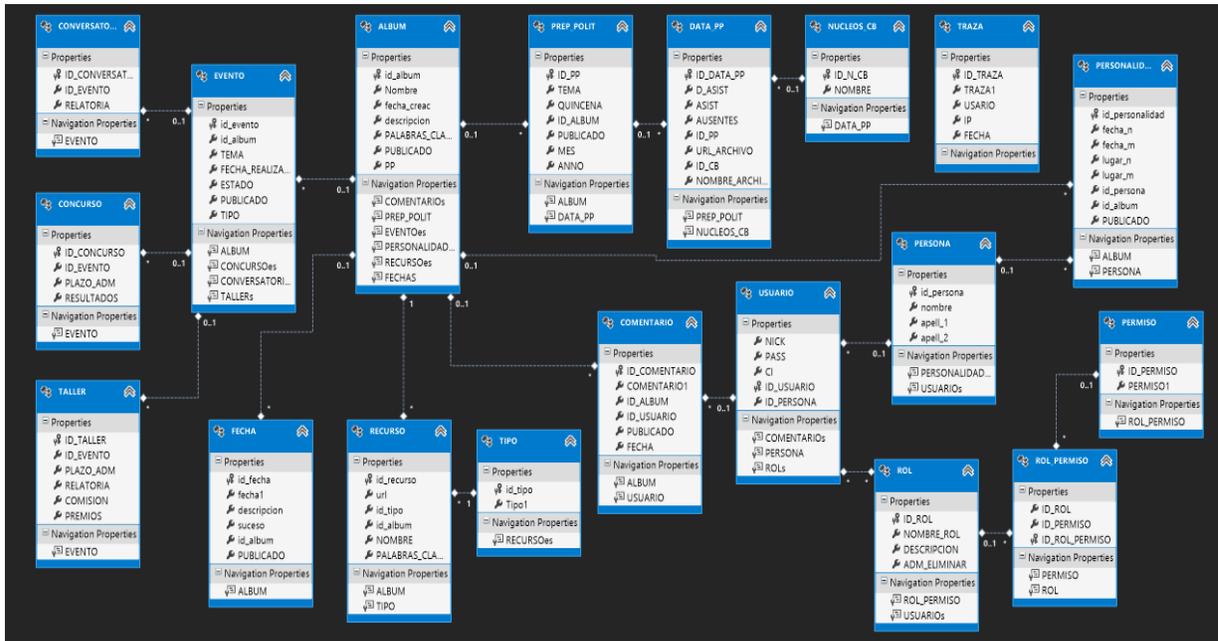


Figura 4: Modelo físico de la base de datos

2.5 Conclusiones del Capítulo

- ❖ En este capítulo se realiza la propuesta de solución según la metodología XP en su fase de planificación que plantea elementos importantes que facilitan una mejor comprensión del software a implementar según las necesidades del cliente.
- ❖ Se precisa el equipo de trabajo que a pesar de contar solamente con 1 integrante, cubre con todos los roles imprescindibles de un equipo XP.
- ❖ Se realiza una descripción de las HU que permitió guiar al equipo durante el proceso de desarrollo de la solución propuesta.
- ❖ Se define el plan inicial de iteraciones y el plan de entregas a seguir, siempre teniendo en cuenta que las mismas pueden sufrir modificaciones durante el proceso de implementación.

Los elementos tratados en este capítulo sirvieron para llegar a un acuerdo entre las partes interesadas en el diseño y la estructura del software a través de la

implementación de las funcionalidades que permitan realizar la gestión de la información en el Organismo Político del MININT, Matanzas.

Capítulo 3: Validación de la solución propuesta

3.1 Introducción

En este capítulo se realizan las pruebas al software que permiten comprobar la calidad de este producto, lo que constituye uno de los pasos más importantes en el desarrollo de un software. No debe existir ninguna característica en el programa que no haya sido probada con la intención de mostrar un error no descubierto hasta entonces y con el fin de verificar la fiabilidad y calidad del software como un todo.

3.2 Pruebas al software

Las pruebas son procesos de ejecución de un programa con la intención de descubrir un error y determinar el status de la calidad de un producto. Los errores pueden empezar a darse desde el primer momento del proceso en que los objetivos pueden estar especificados de forma errónea. Debido a que la comunicación y el trabajo del hombre no son perfectos, el desarrollo del software debe ir acompañado de una actividad que garantice la calidad. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el grado en que el software cumple con los requerimientos. En las pruebas se usan casos de prueba, especificados de forma estructurada mediante Técnicas de Prueba.

Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces. Los niveles de trabajo en los cuales se pueden realizar las pruebas son:

- ❖ Prueba unitarias
- ❖ Prueba de Aceptación
- ❖ Prueba de Seguridad

3.2.1 Plan de pruebas

Según (Loaiza & Zorro, 2010), el plan de pruebas de software se elabora con el fin de especificar qué elementos o componentes se van a probar para que el grupo de trabajo pueda realizar el proceso de validación y verificación de los requerimientos funcionales y no funcionales. Además, a través del plan de pruebas se puede continuar con la trazabilidad de los requerimientos, con lo cual el grupo de trabajo, identifica el porcentaje de avance que se ha logrado hasta cierto momento.

Al desarrollar el plan de pruebas, se puede obtener información sobre los errores, defectos o fallas que tiene el prototipo, así se realizan las correcciones pertinentes, según el caso y se asegura la calidad del producto que se está entregando al cliente.

A continuación se muestra en la **Tabla 25** el plan de pruebas

No	Nombre de la Historia de Usuario	Pruebas a realizar
1	Diseño y creación de la base de datos	1. Test base de datos
2	Diseño de la interfaz de usuario	2. Test diseño de la interfaz de usuario
3	Autenticarse	3. Test autenticarse
4	Gestionar usuario	4. Test insertar usuario 5. Test listar usuario 6. Test editar usuario 7. Test eliminar usuario
5	Gestionar rol	8. Test insertar rol 9. Test listar rol 10. Test eliminar rol 11. Test asignar permisos
6	Gestionar recurso	12. Test insertar recurso 13. Test mover recurso 14. Test eliminar recurso 15. Test ver recurso
7	Gestionar álbum	16. Test insertar álbum 17. Test listar álbum 18. Test editar álbum 19. Test eliminar álbum 20. Test ver álbum 21. Test publicar álbum
8	Gestionar efeméride	22. Test insertar efeméride 23. Test listar efeméride 24. Test editar efeméride 25. Test eliminar efeméride 26. Test ver efeméride 27. Test publicar efeméride

9	Gestionar biografía	<p>28. Test insertar biografía 29. Test listar biografía 30. Test editar biografía 31. Test eliminar biografía 32. Test ver biografía 33. Test publicar biografía</p>
10	Gestionar evento	<p>34. Test insertar evento 35. Test listar evento 36. Test editar evento 37. Test eliminar evento 38. Test ver evento 39. Test publicar evento 40. Test subir bases del evento 41. Test cerrar evento</p>
11	Gestionar preparación política	<p>42. Test insertar preparación política 43. Test listar preparación política 44. Test editar preparación política 45. Test eliminar preparación política 46. Test ver preparación política 47. Test publicar preparación política</p>
12	Gestionar informe preparación política	<p>48. Test insertar informe preparación política 49. Test listar informe preparación política 50. Test ver informe preparación política 51. Test eliminar informe preparación política</p>
13	Gestionar grupo de estudio	<p>52. Test insertar grupo de estudio 53. Test listar grupo de estudio 54. Test eliminar grupo de estudio</p>
14	Generar reporte de asistencia	<p>55. Test generar reporte de asistencia</p>
15	Generar gráficos de asistencia	<p>56. Test generar gráficos de asistencia</p>
16	Generar reporte de almacenamiento	<p>57. Test generar reporte de almacenamiento</p>
17	Generar búsqueda	<p>58. Test generar búsqueda</p>
18	Gestionar comentario	<p>59. Test insertar comentario 60. Test listar comentario 61. Test eliminar comentario 62. Test publicar comentario</p>

19	Gestionar trazas	63. Test listar trazas 64. Test buscar trazas
----	------------------	--

Tabla 25: Plan de Pruebas

3.2.2 Pruebas de aceptación

Las Pruebas de Aceptación (PA) son las realizadas por el cliente y usuarios finales del software. En estas serán probadas las funcionalidades determinadas por el cliente y descritas en las HU, además de los aspectos de seguridad requeridos. Luego de haber superado las pruebas de aceptación podrá considerarse que el software es apto para el uso y despliegue dentro del proyecto.

En la **Tabla 26** se muestra la PA 3: Test autenticarse

Pruebas de Aceptación	
Número Caso de Prueba: 3	No Historia: 3
Nombre Caso de Prueba: Test autenticarse	
Descripción: Se inserta el nombre de usuario y la contraseña para entrar al software en correspondencia con el rol que tiene asignado en la base de datos. Se insertarán de forma incorrecta, dejando campos en blanco para verificar la validación, se tratará de entrar al software poniendo una ruta destino para comprobar que no entre solo si se autentifica correctamente. Luego se insertarán los datos de manera correcta para comprobar que se autentifique un usuario correctamente en la aplicación.	
Condiciones de ejecución: Estar conectado a la Base de Datos	
Entradas/Pasos de ejecución: <ul style="list-style-type: none"> ❖ Presionar el botón Entrar con los campos en blanco. ❖ Fijar una ruta para acceder al software sin haberse autenticado. ❖ Insertar los datos correctamente y presionar el botón Entrar. 	
Resultado esperado: La aplicación debe alertar al usuario cuando se inserten datos erróneos o cuando se dejen campos en blanco. Se otorgan al usuario correspondiente los permisos que le corresponden a su rol.	
Evaluación: Prueba satisfactoria	

Tabla 26: PA 3: Test autenticarse

En la **Tabla 27** se muestra la PA 4: Test insertar usuario

Pruebas de Aceptación	
Número Caso de Prueba: 4	No Historia: 4
Nombre Caso de Prueba: Test insertar usuario	
Descripción: Se insertan los datos necesarios para crear usuarios. Se insertarán de forma incorrecta los datos, dejando campos en blanco, se tendrá en cuenta que el usuario no exista. Luego se insertarán de manera correcta para comprobar que los datos sean almacenados.	
Condiciones de ejecución: El usuario tendrá que tener privilegios de administrador para poder acceder a las funcionalidades correspondientes. Debe estar conectado a la base de datos.	
Entradas/Pasos de ejecución:	
<ul style="list-style-type: none"> ❖ Insertar los datos de manera incorrecta. ❖ Dejar campos en blanco. ❖ Insertar los datos correctamente. 	
Resultado esperado: Se gestionan adecuadamente los datos de los usuarios	
Evaluación: Prueba satisfactoria	

Tabla 27: PA 4: Test insertar usuario

En la **Tabla 28** se muestra la PA 12: Test insertar recurso

Pruebas de Aceptación	
Número Caso de Prueba: 12	No Historia: 6
Nombre Caso de Prueba: Test insertar recurso	
Descripción: Se selecciona la opción (Subir recursos al álbum), se insertan recursos que ya existan en el álbum con el objetivo de que no los inserte para verificar la validación, luego se insertan los recursos que se desee para comprobar que los datos sean almacenados y cargados.	
Condiciones de ejecución: El usuario tendrá que tener permisos suficientes para poder acceder a las funcionalidades correspondientes. Debe estar conectado a la base de datos.	
Entradas/Pasos de ejecución:	
<ul style="list-style-type: none"> ❖ Insertar los recursos que ya existan en el álbum. 	

❖ Insertar los recursos correctamente.
Resultado esperado: Se debe alertar al usuario cuando se inserten recursos repetidos. Cuando se inserten los recursos correctamente, la aplicación debe almacenarlos en la base de datos y mostrarlos.
Evaluación: Prueba satisfactoria

Tabla 28: PA 12: Test insertar recurso

En la **Tabla 29** se muestra la PA 16: Test insertar álbum

Pruebas de Aceptación	
Número Caso de Prueba: 16	No Historia: 7
Nombre Caso de Prueba: Test insertar álbum	
Descripción: Se insertan los datos necesarios para crear un álbum. Se insertarán de forma incorrecta los datos, dejando campos en blanco, se tendrá en cuenta que el álbum ya exista. Luego se insertarán los datos de manera correcta para comprobar que sean almacenados.	
Condiciones de ejecución: El usuario tendrá que tener permisos suficientes para poder acceder a las funcionalidades correspondientes. Debe estar conectado a la base de datos.	
Entradas/Pasos de ejecución: <ul style="list-style-type: none"> ❖ Insertar los datos de manera incorrecta. ❖ Dejar campos en blanco. ❖ Insertar los datos correctamente. 	
Resultado esperado: Debe alertar al usuario cuando se inserten datos erróneos y cuando se trate de insertar un álbum que ya exista. Cuando se inserten los datos correctamente, la aplicación debe almacenarlos en la base de datos y mostrarlos.	
Evaluación: Prueba satisfactoria	

Tabla 29: PA 16: Test insertar álbum

En la **Tabla 30** se muestra la PA 34: Test insertar evento

Pruebas de Aceptación	
Número Caso de Prueba: 34	No Historia: 10

Nombre Caso de Prueba: Test insertar evento
Descripción: Se insertan los datos necesarios para crear un evento. Se insertarán de forma incorrecta los datos, sin seleccionar el tipo de evento, dejando campos en blanco y también se tendrá en cuenta que el evento ya exista. Luego se insertarán los datos de manera correcta para comprobar que sean almacenados.
Condiciones de ejecución: El usuario tendrá que tener permisos suficientes para poder acceder a las funcionalidades correspondientes. Debe estar conectado a la base de datos.
Entradas/Pasos de ejecución: <ul style="list-style-type: none"> ❖ Insertar los datos de manera incorrecta. ❖ Dejar campos en blanco. ❖ Insertar los datos correctamente.
Resultado esperado: Debe alertar al usuario cuando se inserten datos erróneos y cuando se trate de insertar un evento que ya exista. Cuando se inserten los datos correctamente, la aplicación debe almacenarlos en la base de datos y mostrarlos.
Evaluación: Prueba satisfactoria

Tabla 30: PA 34: Test insertar evento

En la **Tabla 31** se muestra la PA 42: Test insertar preparación política

Pruebas de Aceptación	
Número Caso de Prueba: 42	No Historia: 11
Nombre Caso de Prueba: Test insertar preparación política	
Descripción: Se insertan los datos necesarios para crear un álbum de preparación política. Se insertarán de forma incorrecta los datos, dejando campos en blanco, se tendrá en cuenta que el álbum de preparación política ya exista. Luego se insertarán los datos de manera correcta para comprobar que sean almacenados.	
Condiciones de ejecución: El usuario tendrá que tener permisos suficientes para poder acceder a las funcionalidades correspondientes. Debe estar conectado a la base de datos.	
Entradas/Pasos de ejecución: <ul style="list-style-type: none"> ❖ Insertar los datos de manera incorrecta. 	

<ul style="list-style-type: none"> ❖ Dejar campos en blanco. ❖ Insertar los datos correctamente.
<p>Resultado esperado: Debe alertar al usuario cuando se inserten datos erróneos y cuando se trate de insertar un álbum de preparación política que ya exista. Cuando se inserten los datos correctamente, la aplicación debe almacenarlos en la base de datos y mostrarlos.</p>
<p>Evaluación: Prueba satisfactoria</p>

Tabla 31: PA 42: Test insertar preparación política

3.2.3 Pruebas de seguridad

Las pruebas de seguridad buscan medir la confidencialidad, integridad y disponibilidad de los datos, desde la perspectiva del aplicativo, es decir partiendo a identificar amenazas y riesgos desde el uso o interface de usuario final. Una vez ejecutadas las pruebas de seguridad es posible medir y cuantificar los riesgos a los cuales se ven expuestos los aplicativos tanto en la infraestructura interna como externa. (V&V Quality, 2016)

De acuerdo como se plantea en (Seguridad para todos, 2012), VEGA es una plataforma de software libre diseñada para realizar y ejecutar pruebas sobre la seguridad de las páginas web. Las principales características son:

- ❖ Capacidad de realización de análisis de vulnerabilidades y crawler automático.
- ❖ Ejecución de pruebas de accesibilidad UI.
- ❖ Capacidad de realizar un crawler (descargar copia) del website al completo.
- ❖ Función de manipulación manual de paquetes HTTP mediante interceptación mediante Proxy (similar a Paros Proxy, ZAP, etc.)
- ❖ Análisis del contenido
- ❖ Mensajes de notificación personalizables
- ❖ Modelo de datos propio
- ❖ API en Javascript personalizable para el desarrollo de complementos y extensiones personalizadas.

Para realizar las pruebas de seguridad se utilizó VEGA en su version 1.0, la cuál nos permite realizar un conjunto de pruebas que se dividen en 2 grandes módulos y se observan en las siguientes imágenes.

La **Figura 6** muestra Injection Modules

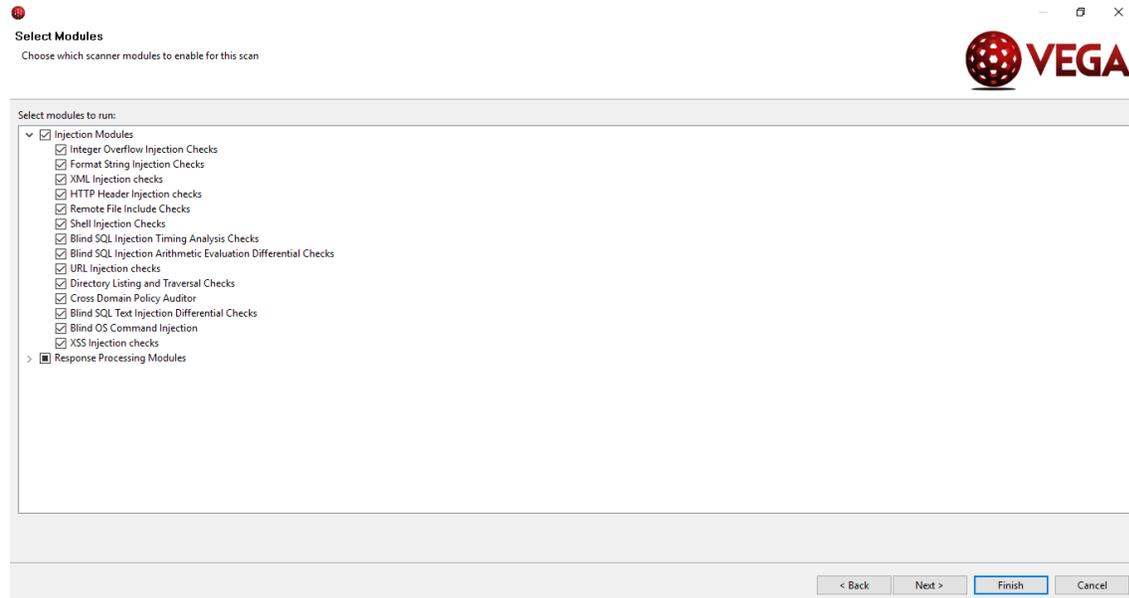


Figura 5: Injection Modules

La **Figura 7** muestra Response Processing Modules.

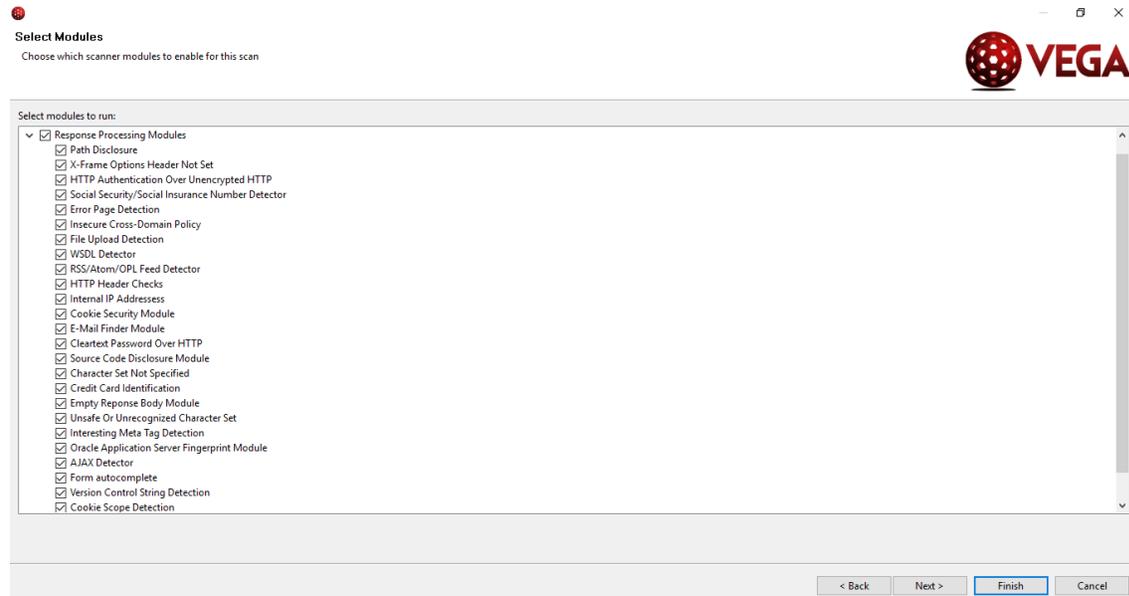


Figura 6: Response Processing Modules.

Seleccionando ambos módulos se le realizó al software una prueba en su totalidad obteniendo como resultado:

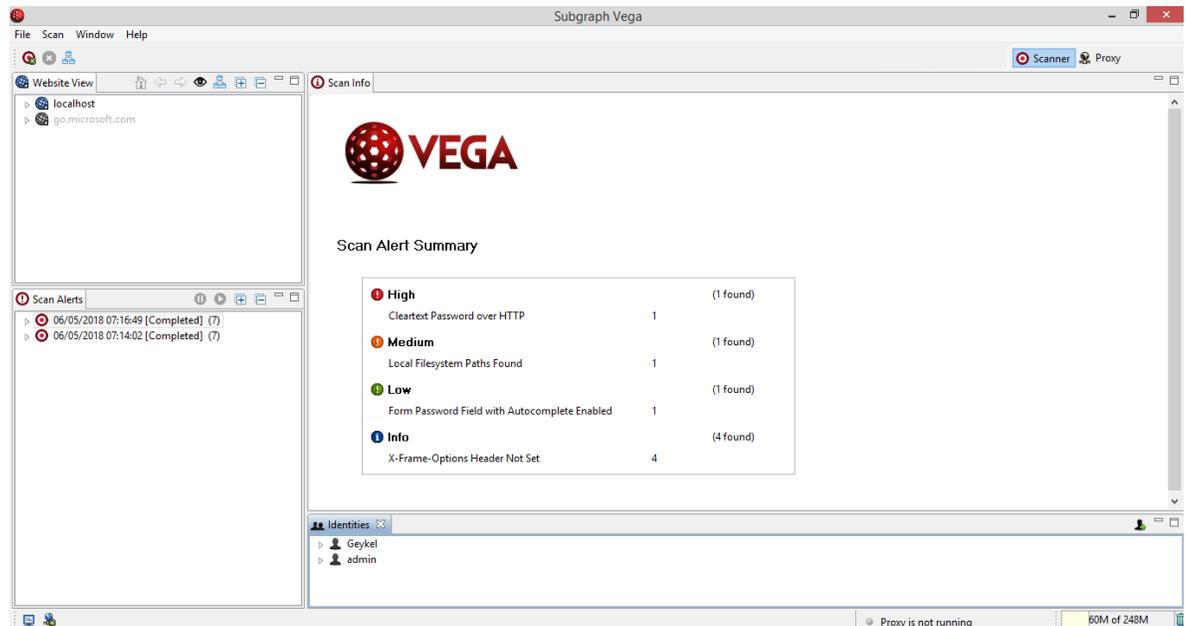


Figura 7: Resultados generales obtenidos por el Vega

3.3 Análisis de los resultados obtenidos

Después de desplegar el proceso de pruebas se lograron resultados satisfactorios, pues tras la detección de diferentes errores, obtenidos fundamentalmente con las pruebas realizadas, se solucionaron varios problemas que impedían el cumplimiento de los requisitos fundamentales del software en cuestión. Se logró obtener un software cuyas funciones se encuentran en correspondencia con las especificaciones acordadas con el cliente y que además cumple con los requerimientos de rendimiento.

Por tanto el desarrollo de este software cumple con las expectativas planteadas a inicios de la investigación y satisface al cliente en su totalidad ya que sirve de apoyo al Organismo Político del MININT Provincial de Matanzas en la gestión de toda la información relacionada con los procesos que se desarrollan, por lo que constituye una herramienta práctica, que permitirá la toma de decisiones certeras en el cumplimiento de su labor.

3.4 Análisis de los Costos

La estimación es el proceso de valoración o medición anticipada de la duración, esfuerzo y costos requeridos para la culminación exitosa de un proyecto. Es necesario tener en cuenta numerosos aspectos que afectan a la estimación como la complejidad, la estructuración, el tamaño, los recursos involucrados y los riesgos asociados al mismo.

Para la estimación de este proyecto se decide utilizar la fórmula de Bohem, puesto que se adecúa en general a los procesos ágiles y como se emplea la metodología de desarrollo de software XP es perfectamente utilizable. La misma plantea que el CT (Costo Total) = CH (Cantidad de Hombres) * TD (Tiempo de Desarrollo) * SM (Salario Mensual).

Durante la etapa de planificación fue necesario estimar el tiempo de implementación de cada HU, por lo que se obtiene al comenzar el proyecto, una estimación inicial de su duración. Esta se realizó basándose en los requerimientos funcionales resultantes de las sucesivas entrevistas con el cliente, la experiencia del programador en el trabajo con el lenguaje de programación, el entorno de desarrollo, el conocimiento sobre el tema de investigación y las técnicas de programación necesarias para resolver el problema.

Duración del proyecto en meses

28 semanas/4 semanas/1 mes= 7 meses

TD=7 meses y 1 mes para realizar pruebas, por tanto, son 8 meses.

Costo total del proyecto

Utilizando la fórmula de Bohem (1999):

Costo = CH * SM * TD

Costo = 1 * 625 * 8

Costo = 5000

El costo total del proyecto es de \$ 5000.00.

El software será puesto a disposición del MININT, para las funciones que necesite, por lo que no se recibirá ningún beneficio monetario.

3.5 Conclusiones del Capítulo

En este capítulo se mostraron los elementos de prueba de la herramienta, los casos de prueba que se le aplicaron y se explicaron las estrategias de prueba. Las pruebas realizadas utilizando las técnicas anteriormente planteadas fueron de gran importancia para demostrar el buen funcionamiento del software y el cumplimiento de los requerimientos del cliente. Como resultados finales se pudo obtener un software, con una apariencia agradable y fácil de usar. La planificación inicial se cumplió a un 100%, se utilizaron las herramientas más actuales para su desarrollo, y el plan de entrega fue cumplido con éxito.

Conclusiones

Como resultado de esta investigación quedaron cumplidos los objetivos trazados arribando a las siguientes conclusiones:

- ❖ Los softwares encontrados, vinculados al tema se salen del ámbito y no le dan solución al problema planteado por lo que no es factible su utilización.
- ❖ El estudio realizado sobre los antecedentes, el estado actual de la temática, la bibliografía y documentos relacionados con el objeto de estudio, permitió aportar los elementos necesarios para dar solución a la problemática planteada.
- ❖ El uso de la metodología de desarrollo de software XP permitió el diseño, implementación y prueba de un software a la medida de las necesidades del cliente y flexible ante los cambios en los requerimientos.
- ❖ Se determinaron como herramientas de software más factibles para la construcción de la solución framework para aplicaciones Web, *Entity Framework* empleando el lenguaje de programación C# y como gestor de base de datos Oracle 11g.
- ❖ El software implementado da solución a la situación problemática que lo originó y su explotación significará una mejora considerable en la calidad y eficiencia de los procesos que automatiza.

De forma general, se concluye que el software desarrollado es una herramienta confiable y fácil de utilizar pues eleva la calidad de la gestión de la información en el Organismo Político del MININT, Matanzas, reduce los errores a cometer, facilita y perfecciona el proceso. Por estas razones, queda respondida la pregunta científica planteada en la introducción.

Recomendaciones

Desde el punto de vista del alcance del presente trabajo y teniendo en cuenta el momento de desarrollo del mismo, se proponen las siguientes recomendaciones:

- ❖ Extender el software no solo en la provincia, sino desplegarlo al resto de los centros homólogos del país.
- ❖ Propiciar nuevos estudios e investigaciones en aras de lograr un mayor nivel de eficiencia y de informatización de los procesos.
- ❖ Aprovechar las posibilidades de información almacenadas en el software e implementar funcionalidades que respondan a nuevos intereses de la institución.

Bibliografía

- Aballi, J. L. (2016). *Herramienta web para la gestión de los procesos de la UJC municipal Unión de Reyes*. Matanzas.
- Achour, M. (2007). *Manual de PHP*.
- Acosta, L. (2015). *Herramienta web para la gestión de los procesos de los militantes en la UJC Municipal de Unión de Reyes*. Matanzas : Universidad de Matanzas "Camilo Cienfuegos" .: Editorial Universitaria.
- Alcalde, A. (2013). *El baúl del programador*. Obtenido de <http://elbauldelprogramador.com/los-10-mejores-frameworks-gratis-de-aplicaciones-web/>
- Alcalde, A. (2017). *El baúl del programador*. Obtenido de <http://elbauldelprogramador.com/los-10-mejores-frameworks-gratis-de-aplicaciones-web/>
- Álvarez, M. Á. (2007, 8 30). *DesarrolloWeb*. Obtenido de <http://desarrolloweb.com>
- Álvarez, M. Á. (2014, enero 2). *Desarrolloweb.com*. Obtenido de Que es MVC: <http://www.desarrolloweb.com/articulos/que-es-mvc.html>
- Álvarez, R. (2012). *Introducción al HTML*. Obtenido de <http://www.desarrolloweb.com/articulos/arquitectura-cliente-servidor.html>.
- Amaro, L. G. (2015). *Sistema Informático de apoyo al control del consumo de electricidad en el MININT, Matanzas*. Matanzas.
- Armas, R. M. (2015). *Sistema Informático para el control de asistencia y confección de la Pre-Nómina en la Empresa0 Integral de Servicios Automotores Matanzas*. Matanzas.
- Arocha, A. T. (2016). *Aplicación web para la gestión de equipos en depósito en la EMAE Matanzas*. Matanzas.
- Arthur, L. (2015). *ehow*. Obtenido de http://www.ehowenespanol.com/lista-ventajas-inconvenientes-oracle-database-11g-info_201876/
- Bermúdez, H. G. (2017). *Aplicación web para gestionar el proceso de renta y disponibilidad de autos en la Empresa CUBACAR*. Matanzas.
- Bowler, T., & Bancner, W. (2009). *Symfony 1.3 web application development*.
- Campos, S. G. (2015). *Programación Extrema: Prácticas, Aceptación y Controversia*.
- Chaffer, J. (2009). *Learning JQuery 1.3: Better Interaction and Web Development with 21 Simple JavaScript Techniques*.
- Cristian. (2008). *Ventajas de usar CSS*. Obtenido de <http://www.stardustxs.com/2008/03/05/ventajas-de-usar-css/>

- Decreto Ley No.67, 67 (1976).
- Dueñas, L. C. (2014). *Aplicación Web para la planificación del trabajo en el Telecentro TV Yumurí*. Matanzas.
- elGestor. (2016, 11 15). *elGestor Gestión de Vehículos*. Obtenido de <http://www.portalprogramas.com/gestion-de-vehiculos/>
- Falcón, C. (2006). *An Epic Review of PyCharm 3. Vim User's Perspective*.
- Fernández, A. G. (2017). *Aplicación web para la gestión de los procesos de la militancia en la Unión de Jóvenes Comunistas de la Universidad de Matanzas*. Matanzas.
- Fernández, P. (2011). *JetBrains anuncia PHP Storm*.
- Fernández, R. A. (2016, 04 29). Procedimiento para la atención de los equipos de la reserva estatal.
- Fuentes, J. (2015). *Desarrollo de Software Ágil. Extreme Programming y Scrum*.
- García, D. C. (2016). *Aplicación informática para gestión de reservas y estancias Hostal Azul Matanzas*. Matanzas.
- García, D. C. (2016). *Aplicación informática para gestión de reservas y estancias Hostal Azul Mtzas*. Matanzas.
- García, L. C. (2015). *Portal Web educativo para el desarrollo de habilidades comunicativas en el idioma inglés*. La Habana, Cuba: Universidad de Ciencias Informáticas.
- Garrett, J. (2005). *Ajax: A new approach to web applications*.
- Guardado, I. (2013). *Introducción a Object-Relational Mapping (ORM)*. Obtenido de <http://web.ontuts.com/tutoriales/introduccion-a-object-relational-mapping-orm/>
- Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2017). *Teoría de autómatas, lenguajes y computación*.
- Intercambios virtuales*. (2013). Obtenido de Intercambios virtuales: <http://www.intercambiosvirtuales.org/software/visual-studio-ultimate-2012-msdn-espanol>
- José, A. C. (2016). *INTRODUCCIÓN DE LAS TICs EN EL ISCED SUMBE, ANGOLA: LA GEOMETRIA PLANA EN EL CURSO DE MATEMÁTICA*. ANGOLA.
- Loaiza, V., & Zorro, L. (2010). *Plan de Pruebas de Software*.
- Luque, E. G. (2012). *Informatización del control de las reparaciones y mantenimientos de los vehículos automotores en la UEB Municipal de la Empresa Eléctrica de Matanzas*. Matanzas.
- Marcano, H. (2009). *Desarrollo de una aplicación educativa bajo ambiente web, como apoyo para la enseñanza de la asignatura bases de datos orientados a objetos, para la carrera de Licenciatura en Informática de la Universidad de Oriente*.

- Martínez, D. (2014). *PostgreSQL vs MySQL*. . Obtenido de [Http://danielpecos.com/documents/postgresql-vs-mysql/](http://danielpecos.com/documents/postgresql-vs-mysql/)
- Medina, C. S. (2017). *Software de gestión para la planificación y control del entrenamiento deportivo en el atletismo de pista*. Matanzas.
- Meléndez, K., Abraham, D., & Carla, B. (2005, 01 30). Estimación del Esfuerzo en la etapa Post-Arquitectura usando el modelo COCOMO II Versión 2.0.
- Microsoft Corporation. (2015). Obtenido de <https://www.microsoft.com/en-us/search/DownloadsDrillInResults.aspx?q=Visual%20Studio%202012>
- MININT. (2017). *Misiones del Organismo Político*. La Habana.
- MININT. (2018, enero 1). *La luz de la tropa*. Obtenido de Intranet Provincial MININT Matanzas: <http://wwwtp.webmin.int/politico/default.aspx>
- MININT. (2018, enero 1). *Organismo Político Matanzas*. Obtenido de Intranet Provincial MININT Matanzas: <http://wwwprov.webmin.int/matanzas/Sitios%20con%20bibliotecas/SeccionPolitica.aspx>
- Mondéjar, A. G. (2015). *Aplicación Web para el apoyo a los ajustes de planes de estudio en la Universidad de Matanzas*. Matanzas.
- Muñiz, O. R. (2015). *Sistema informático para la automatización del análisis de indicadores económicos por medio de redes neuronales*. Matanzas.
- Perera, J. M. (2017). *Aplicación web para la gestión de información del Jardín Botánico de Matanzas*. Matanzas.
- Pérez, F. F. (2015). *Sistema Informático para control de trabajo en taller de chapistería y pintura CUBACAR*. Matanzas.
- Pérez, F. F. (2015). *Sistema Informático para el control de trabajo en el taller de Chapistería y pintura CUBACAR*. Matanzas.
- Pérez, J. (2009). *Introducción a JavaScript*.
- Pichardo, J. T. (2010). *Aplicación para el control técnico de los vehículos de Ómnibus ASTRO*.
- Pressman. (2010). *Ingeniería del software. Un enfoque práctico*. Mexico.
- Quintero, E. (2012). *Sistema de gestión de la información para la secretaria general de la Universidad de Matanzas "Camilo Cienfuegos"*. Matanzas.
- Resolucion, 20 (04 11, 2013).
- Resolución 4 (SIME 2003).
- Reuelta, A. B. (2015). *Aplicación Informática para la Gestión de Equipos y Piezas de Repuesto de Importación de la CTE Antonio Guiteras*. Matanzas.
- Risueño, P. (2013). *Comenzando con Bootstrap, framework responsive*.

- Rivero, O. (2015). *Sistema Informático para la automatización del análisis de indicadores económicos por medio de redes neuronales*. Matanzas.
- RocketTheme. (2009). *Metodologías de desarrollo de software*.
- Rodríguez, J. A. (2014). *Sistema Informático para la gestión de estudiantes internos y control de capacidades en la Residencia Estudiantil de la*. Matanzas.
- Romero, L. F. (2014). *Herramienta web para la gestión de compra de materias primas en la Empresa de Recuperación de Materias Primas de Matanzas*. Matanzas.
- Sánchez. (2003). *Diferencias entre metodologías ágiles y no ágiles*.
- Sánchez, T. P., & Letelier. (2003). *Metodologías Ágiles en el desarrollo de Software*.
- Sandoval, G. A. (2014, octubre 20). *Implementando patrón repositorio- Repository pattern en C#*. Obtenido de El tavo: <http://www.eltavo.net/2014/10/patrones-implementando-patron.html>
- Santiago, F. D. (2017). *SOFTWARE PARA LA GESTIÓN DE LA INFORMACIÓN DEL CLIENTE INTERNO EN SISTEMAS HOTELEROS DESDE LA PERSPECTIVA DEL ENDOMARKETING*. Matanzas.
- Seco, J. A. (2014). *El lenguaje de programación C#*. Obtenido de Desarrollo web.com: <http://www.desarrolloweb.com/articulos/561.php>
- Seguridad para todos. (2012). *Seguridad para todos*. Obtenido de <http://www.seguridadparatodos.es/2012/02/vega-nueva-solucion-open-source-para.html>
- Senso, J. A. (2015). *Servidores WAMP para producción local*. Obtenido de <http://tecnologiasweb.jsenso.es/servidores-wamp/>
- Tamayo, S. A. (2016). *Mantenimiento perfectivo correctivo a la aplicación web Ajustes-UM*. Matanzas.
- Tedeschi, N. (2013). *¿Qué es un Patrón de Diseño?*
- Torres, R. B., & Paguay Soxo, P. (2014, 12 17). *dspace*. Obtenido de <http://dspace.epoch.edu.ec/handle/123456789/3633>
- Torres, R. B., & Paguay Soxo, P. X. (2014, 12 17). *dspace*. Obtenido de <http://dspace.epoch.edu.ec/bitstream/123456789/470/1/18T00357%20UDCTFIYE.pdf>
- Trey-SAT. (2016, 11 15). *Trey-SAT*. Obtenido de <http://http://www.portalprogramas.com/trey-sat/>
- V&V Quality. (2016). *V&V Quality*. Obtenido de <http://vyvquality.com/pruebas-seguridad/>
- Vehículos Pro. (2016, 11 10). *Vehículos Pro*. Obtenido de <http://vehiculospro.es/>
- Yanes Enriquez, O., & Gracia del Busto, H. (2013). *Revista Telem@tica. Vol. 10. No. 3*. Obtenido de <http://revistatelematica.cujae.edu.cu/index.php/tele/article/download/23/21/0>
- Zanotti, A. (2016). *El software libre su difusión en Argentina:mercado,estado,sociedad*.

Anexos

Anexo 1: Informe de la actividad de Preparación Política

"Ordinario"



REPUBLICA DE CUBA

Ministerio del Interior

Dirección Política

Informe de la Actividad De Preparación Política

Grupo de Estudio: _____

Unidad: _____ Órgano: _____

Tema: _____

Fecha: _____ Quincena: _____

Asistencia

Deben asistir: _____ Asistieron: _____ Ausentes: _____

% de asistencia: _____

Asistieron

No.	Nombre y Apellidos
_____	_____
_____	_____
_____	_____

Ausentes

No.	Nombre y Apellidos	Causas
_____	_____	_____
_____	_____	_____
_____	_____	_____

Resumen valorativo
