UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad de Ciencias y Tecnologías Computacionales

Centro Representación y Análisis de Datos

Módulo Nominaciones para la gestión de los procesos electorales en la Suite decidimOS

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autor(es): Roberto Valladares Martín

Tutor(es): Ing. Lester Collado Rolo

Ing. Leonel Eduardo Perdomo Roig

La Habana, 14 noviembre de 2023

"Año 65 de la Revolución

DECLARACIÓN DE AUTORÍA

Declaro por este medio que yo Roberto Valladares Martín, soy el autor del Trabajo de Diploma Módulo Nominaciones para la gestión de los procesos electorales en la Suite decidimOS y que autorizo a la Universidad de las Ciencias Informáticas a hacer uso de mismo en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Y para que así conste, firmo la presente declaración de autoría en La Habana a los 14 días del mes de noviembre del año 2023.

_____

Roberto Valladares Martín

Autor

_____

Lester Collado Rolo

Tutor

_____

Leonel Eduardo Perdomo Roig

Tutor

“Si una persona es perseverante, aunque sea dura de entendimiento, se hará inteligente; y aunque sea débil se transformará en fuerte”.

Leonardo Da Vinci

DEDICATORIA

Dedicado a mi familia y en especial a aquellos que me mostraron su apoyo incondicional, y que lamentablemente no pueden ver como por fin me gradúo.

AGRADECIMIENTOS

Agradecimiento a mi familia, mis amigos, a mis tutores y a todos aquellos que he conocido a lo largo de estos años y que me han ayudado a ser la persona que soy hoy.

RESUMEN

La transparencia y el control de los procesos de elecciones son claves en la sociedad y tienen alto impacto en la vida política de la nación. En este empeño ha surgido una relación de trabajo entre la Universidad de las Ciencias Informáticas y el hoy Consejo Electoral Nacional que data de alrededor de 8 años. Implementando varias soluciones de software para el desarrollo de los principales procesos electorales: elecciones municipales, de gobernadores y vicegobernadores, nacionales y referendos. Un conjunto de herramientas informáticas conforma el ecosistema de software para la gestión de los procesos electorales basado en una arquitectura orientada a componentes, desacoplada y con buenas prácticas tecnológicas y de seguridad para garantizar la escalabilidad y la funcionalidad de las soluciones. La informatización de los procesos electorales establecidos en la Ley Electoral vigente está completamente cubierta, lo que ha permitido un alto nivel de inmediatez de la información gestionada entre todas las estructuras electorales. El presente trabajo aborda el desarrollo del módulo de Nominaciones para el ecosistema decidimOS contribuyendo al completamiento funcional del ecosistema. Los resultados fueron integrados y probados cumpliendo las diferentes actividades definidas en el proceso de desarrollo.

Palabras clave: Ecosistema de Software, Elecciones, Nominaciones, Procesos Electorales.

Abstract

Transparency and control of the electoral process are of key importance in society and have a great impact on the political life of the nation. In this endeavor, a working relationship between the Universidad de las Ciencias Informáticas and the National Electoral Council has existed for about 8 years. Implementation of several software solutions for the development of the main electoral processes: municipal elections, governors and deputy governors, national elections and referendums. A set of IT tools make up the software ecosystem for the management of electoral processes, based on a component-oriented architecture, decoupled and with good technological and security practices to ensure the scalability and functionality of the solutions. The computerization of electoral processes established in the current electoral law is fully covered, which has allowed a high level of immediacy of the information managed among all electoral structures. The present work deals with the development of the Nominations module for the decidimOS ecosystem, contributing to the functional completion of the ecosystem. The results have been integrated and tested by fulfilling the different activities defined in the development process.

Keywords: Electoral Processes, Elections, Nominations, Software Ecosystem

ÍNDICE

INTRODUCCIÓN 1

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA 5

1.1 Conceptos asociados al problema 5

1.2 Estudio del Estado del Arte 6

1.3 Tecnologías 6

1.4 Metodología de desarrollo de software AUP-UCI 9

1.5 Conclusiones del capítulo 9

CAPÍTULO 2: DISEÑO DE LA PROPUESTA DE SOLUCIÓN 10

2.1 Modelo Conceptual 10

2.1.1 Descripción de Conceptos 11

2.2 Especificación de Requisitos 11

2.2.1 Requisitos Funcionales 12

2.2.2 Requisitos No Funcionales 20

2.3 Historias de Usuario 22

2.4 Arquitectura de software 24

2.5 Patrones del diseño 26

2.6 Modelo Entidad Relación 28

2.7 Conclusiones del capítulo 29

CAPÍTULO 3: VALIDACIÓN DEL MÓDULO NOMINACIONES 31

3.1 Estrategia de pruebas 31

3.1.1 Prueba Unitaria 31

3.1.2 Prueba de Integración 32

3.1.3 Prueba de Rendimiento 34

3.2 Conclusiones del capítulo 35

CONCLUSIONES GENERALES 37

Recomendaciones ..... **Error! Bookmark not defined.**

Bibliografía 39
ANEXOS 43

ÍNDICE DE TABLAS

Tabla 1. Estudio de Aplicaciones Homólogas. ..... **Error! Bookmark not defined.**
Tabla 2. Requisitos Funcionales……..…….…….13
Tabla 3. Requisitos No Funcionales del Producto…….…….21
Tabla 4. Requisitos No Funcionales de la Organización…….…….22
Tabla 5. Requisitos No Funcionales Externos…….……..…….22
Tabla 6. Historia de Usuario RF4.....23
Tabla 7. Historia de Usuario RF26.....24

ÍNDICE DE FIGURAS

Figura 1. Modelo del Dominio 7
Figura 2. Patrón Arquitectónico Modelo Vista Controlador……..……26
Figura 3. Patrón Arquitectónico Modelo Vista Plantilla…….……27
Figura 4. Patrón Controlador.....28
Figura 5. Patrón Alta Cohesión.....28
Figura 6. Patrón Observador.....29
Figura 7. Patrón Decorador.....29
Figura 8. Modelo de Datos.....30
Figura 9. Prueba Unitaria Crear Nominados.....32
Figura 10. Prueba de Integración Listar Nominados.....33
Figura 11. Prueba de Integración Crear Nominados satisfactoria.....34
Figura 12. Prueba de Integración Crear Nominados fallida.....34
Figura 13. Prueba de Rendimiento Listar Nominados Reporte Resumen.....35
Figura 14. Prueba de Rendimiento Listar Nominados Gráfico de Resultados.....36

INTRODUCCIÓN

El mundo se encuentra inmerso actualmente en una era de avances tecnológicos que transforman la manera en que interactuamos con nuestro entorno. Uno de los ámbitos en los que estos han dejado una huella significativa es en el de las elecciones y el proceso de votación. A medida que la sociedad se vuelve más consciente de la importancia de garantizar la transparencia, la eficiencia y la equidad en los procesos electorales, surge la necesidad de desarrollar sistemas que ayuden a gestionar el proceso electoral, en este sentido existen muchos países y empresas que está experimentando con sistemas de votación electrónica y otras innovaciones tecnológicas para mejorar estos procesos (Simon, 2018).

Las aplicaciones tecnológicas para la gestión electoral no se limitan únicamente al voto electrónico. De hecho, existen soluciones tecnológicas diseñadas para cada una de las etapas del ciclo electoral, desde la preparación previa al proceso electoral hasta el escrutinio y la publicación de resultados. Lograr procesos electorales que reflejen la clara voluntad de la ciudadanía implica un esfuerzo que va más allá del día de la votación (European Comission, 2020).

En Cuba, las tecnologías son cada vez más utilizadas para la gestión de procesos, la resolución de problemas y el manejo de datos. Por ello, su empleo por parte del Consejo Electoral Nacional (CEN), órgano del Estado con la misión de “organizar, dirigir y supervisar las elecciones, consultas populares, plebiscitos y referendos” (Gaceta Oficial de la República de Cuba, 2019), resulta imprescindible para garantizar la rapidez, el flujo y procesamiento constante de la información, desde los colegios en los municipios, hasta los niveles provinciales y nacionales, logrando así unas elecciones transparentes y con un alto nivel de calidad (Ministerio de Comunicaciones, 2018).

Uno de los pilares fundamentales para el desarrollo de las tecnologías en el CEN es la Universidad de las Ciencias Informáticas (UCI), siendo el Centro de Representación y Análisis de Datos de la universidad el encargado, desde 2015, de garantizar los procesos de elecciones, su soporte tecnológico y gestión del centro de información del Consejo Electoral Nacional.

Según (Camacho González, 2021) el Consejo Electoral Nacional (CEN), cuenta con los Sistemas de Gestión e Información del Proceso Electoral: SIGEL Parciales v3.0 y SIGEL Generales v2.0 para la gestión de la información de las actividades electorales entre los

diferentes niveles y los sistemas: SIGEL Miembro y Silectos para la gestión de miembros y de electos respectivamente durante los periodos no electorales.

Como consecuencia de cambios en la Ley Electoral, decretada en el año 2019, surge el ecosistema decidimOS como resultado del fuerte conocimiento de los procesos de negocio de las distintas organizaciones y el avance natural de las tecnologías que implican obligadas tareas de mantenimiento y evolución (Camacho González, 2021).

El Módulo Nominaciones tiene como principal objetivo: gestionar los datos pertenecientes al período de nominaciones del proceso de elecciones parciales realizado en Cuba. El sistema cuenta con un control de flujo que divide la aplicación en 3 etapas. El mismo consiste en ir mostrándose paulatinamente las funcionalidades durante todo el proceso eleccionario, lográndose con este que llegada una determinada fecha (definida por la Comisión Electoral Nacional) y dada por concluida la etapa, de muchas funcionalidades solo se puedan observar sus datos, mientras que otras harán su aparición para ser gestionadas.

El sistema que se utiliza actualmente presenta diversas limitaciones que afectan su desempeño. En primer lugar, la aplicación no está diseñada para gestionar la estructura electoral Distrito durante la etapa no electoral, lo que dificulta la gestión de la información por parte de los municipios que tienen distritos. Además, la asignación de roles no permite que los municipios con distrito tengan acceso completo a la información, limitándose únicamente a revisar y aprobarla. También se han producido cambios recientes en las estructuras de los consejos electorales y en la cantidad de integrantes, lo que genera un desajuste con los rasgos del proceso electoral deseado. Esto puede generar problemas en la gestión de la información y dificultar la toma de decisiones adecuadas. Además, existe el riesgo de que alguien tenga permisos que no le corresponden, lo que puede comprometer la integridad del proceso electoral.

En correspondencia con la situación problemática antes planteada se deriva como **problema de investigación:** ¿Cómo contribuir a la gestión de la información de los procesos electorales en la Suite decidimOS?

El **objeto de estudio** asociado al problema es: Aplicaciones web para la gestión de la información de los procesos electorales.

En correspondencia con el objeto de estudio planteado se define como **campo de acción**: Soporte al módulo Nominaciones para la gestión de la información de los procesos electorales en la Suite decidimOS.

Basándose en el problema planteado y considerando el objeto de estudio identificado, se plantea como **objetivo general:** desarrollar el módulo Nominaciones utilizando tecnologías web para la gestión de la información de los procesos electorales.

De dicho objetivo general se derivan los siguientes **objetivos específicos**:

- Analizar y sintetizar los referentes teóricos sobre la gestión de la información de los procesos electorales.
- Realizar el análisis y diseño de la propuesta de solución.
- Implementar la propuesta de solución.
- Validar el proceso de implementación e integración y verificar el correcto funcionamiento del Módulo Nominaciones.

Teniendo en cuenta lo antes visto y como parte del proceso de desarrollo de la investigación se presentan como **métodos teóricos** los siguientes:

- Análisis y síntesis: El análisis estará presente en las primeras etapas de la investigación y la síntesis fundamentalmente en las etapas finales y en la obtención de las conclusiones.
- Los métodos históricos: Se realizará un estudio de la versión anterior de la aplicación, así como las causas que dieron origen a la creación de la nueva aplicación.
- La modelación: Se crearán abstracciones para representar la realidad compleja, representar el negocio y realizar una descripción lo más exacta posible del sistema, sus Módulos y de las actividades llevadas a cabo en él.
- El sistemático: Se estudiará el software SIGEL-Nominaciones v3.0 mediante sus Módulos, así como la relación entre ellos. Esa relación determinará por un lado la estructura y la jerarquía de cada Módulo en el objeto y por otra parte su dinámica.

Los **métodos empíricos** a utilizar serán:

- Las entrevistas: Se le realizarán entrevistas al cliente con el fin de obtener respuestas verbales a las interrogantes planteadas sobre el problema propuesto, así como para comprender el negocio y detectar requisitos.

- La observación: Permitirá conocer la realidad del sistema existente mediante su percepción directa para identificar las descripciones de lo que el sistema debe hacer: el servicio que ofrece y las restricciones en su operación (Sommerville 2011).

Estructura del trabajo de diploma

El trabajo de diploma presenta la siguiente estructura: introducción, tres capítulos, conclusiones generales, recomendaciones, referencias bibliográficas y anexos.

Capítulo 1: **Fundamentación teórica del módulo Nominaciones.**

En el capítulo se definieron los conceptos relacionados con el dominio de investigación. De igual forma, se realizó un estudio sobre los sistemas informáticos de apoyo a los procesos eleccionarios existentes en Cuba con el fin de identificar posibles soluciones para el problema de investigación definido. También se describió la metodología y las herramientas a emplear en la construcción del sistema.

Capítulo 2: Análisis y diseño del módulo Nominaciones.

En el capítulo se generó un modelo conceptual con el cual se logró un mejor entendimiento del negocio, después de realizado un análisis se definieron los requisitos del sistema, lo que condujo a la creación de historias de usuarios. Además, se definieron las arquitecturas del sistema y los patrones de diseño que se van a emplear.

Capítulo 3: **Implementación y prueba del módulo Nominaciones.**

En el capítulo se definió la estrategia de pruebas a emplear para comprobar su correcto funcionamiento en distintos niveles y además se muestra el resultado de cada una de las pruebas realizadas.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En el siguiente capítulo se abordarán definiciones de diferentes autores, así como los principales conceptos del dominio de la investigación. Se evidenciarán cada una de las herramientas a utilizar para dicha investigación y a su vez quedará plasmada la importancia que cada una representa para la misma. También se analizarán sistemas similares para la gestión de procesos electorales tanto en Cuba como en el mundo adquiriendo de cada una de estas las buenas prácticas.

1.1 Conceptos asociados al problema

Para la mejor comprensión del estudio realizado se hace necesario definir los conceptos fundamentales relacionados con el dominio de la investigación. Para ello se analizará además los criterios de los diferentes autores y se adoptaran las definiciones que más se adecuen a este trabajo.

Ecosistema de software: Un ecosistema de aplicaciones se refiere a una colección de sistemas y plataformas de software que respaldan el desarrollo y la implementación de varias aplicaciones. Implica la interacción entre los desarrolladores, los usuarios y la plataforma, con el objetivo de crear valor y proporcionar un sistema operativo unificado para múltiples dispositivos y escenarios. (IEEE Software, 2022)

Gestión de la información: cubre todos los aspectos de la producción, coordinación, almacenamiento, recuperación y difusión de información, independientemente del formato o la fuente, y sugiere un aspecto organizativo que impartirá algún grado de valor a la información. (Prytherch, 2005)

Tecnologías Web: Las tecnologías web son herramientas y recursos utilizados para crear y administrar sitios web en Internet. Estas tecnologías incluyen lenguajes de programación, marcos de trabajo, protocolos de comunicación y herramientas de diseño gráfico, entre otros. (Torricella-Morales, 2020)

1.2 Estudio del Estado del Arte

En la tabla que se muestra a continuación, quedan plasmados ejemplos de aplicaciones similares que se encuentran en el mercado que poseen cierta semejanza a la solución buscada.

Tabla 1: Estudio de Aplicaciones Homólogas. (Elaboración Propia)

Nombre	Sistemas de Despliegue	Cumplimiento con la actual Ley Electoral	Software Libre	Alcance	Accesibilidad
Elena I y II	Escritorio	No	No	Nacional	Privativo
SIGEL (Parciales y Generales) v3.0	Web	No	Si	Nacional	No Privativo
Scytl Online Voting	Web	No	No	Internacional	Privativo
Smartmatic	Web	No	No	Internacional	Privativo

El estudio de estos sistemas arrojó como conclusión que ninguno de ellos cumple con lo establecido con la actual Ley Electoral del año 2019, además de que algunos de estos sistemas son privativos por lo que no es posible la modificación del código para adaptarlo a las necesidades del cliente.

1.3 Tecnologías

Para la implementación del Módulo Nominaciones se tomarán como herramientas base las establecidas por el equipo de desarrollo a cargo del sistema decidimOS, las mismas son: el lenguaje de modelado UML v2.5, la herramienta CASE Visual Paradigm para UML v15.0, los marcos de trabajo Django v3.8 y Angular v7.3.3, los lenguajes de programación Python v3.10.8

y TypeScript v3.2.4, el sistema gestor de base de datos PostgreSQL v12 y como entorno de desarrollo se utilizará VS Code. Estos frameworks fueron escogidos ya que juntos permiten la creación de sistemas web robustos y seguros, donde se tiene en cuenta la separación de responsabilidades, un desarrollo rápido y eficiente, y la alta escalabilidad de ambos sistemas que permite manejar grandes volúmenes de datos y usuarios.

Visual Paradigm v15.0

Se escogió esta herramienta profesional pues soporta el ciclo de vida completo del desarrollo de *software*, es multiplataforma, utiliza UML como lenguaje de modelado, cuenta con una versión libre para la comunidad internacional y apoya la construcción de aplicaciones con mayor calidad. Además, esta versión permite diseñar varios tipos de diagramas de clases, código inverso, generar código desde diagramas, generar documentación, proporciona una interfaz gráfica de usuario y soporta la revisión ortográfica, ofreciendo sugerencias para diversos idiomas (Diomidis, 2010).

UML (*Unified Modeling Language*) v 2.5

Se definió para el proyecto el Lenguaje de Modelado Unificado v2.5, pues es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de *software*. Capta decisiones y conocimientos sobre los sistemas que se deben construir. Se usa para entender, diseñar, hojear, configurar, mantener, y controlar la información sobre tales sistemas. Está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios (Visual Paradigm, 2023).

Django v3.8

Django es un *framework* de desarrollo que puede llevar las aplicaciones web desde el concepto al lanzamiento en cuestión de horas, se ocupa de gran parte de la molestia del desarrollo web. Fue seleccionado este *framework* para el desarrollo de la investigación por ser gratuito y de código abierto, incluye docenas de extras que puede usar para manejar tareas comunes de desarrollo web, se encarga de la autenticación del usuario, la administración de contenido, los mapas del sitio y muchas más tareas, se toma muy en serio la seguridad y ayuda a los desarrolladores a evitar muchos errores de seguridad comunes, como la inyección de SQL, las secuencias de comandos entre sitios y la falsificación de solicitudes. Su sistema de autenticación de usuario proporciona una forma segura de administrar cuentas de usuario y contraseñas. Las empresas, organizaciones y gobiernos han utilizado Django para construir

todo tipo de cosas: desde sistemas de gestión de contenido hasta redes sociales y plataformas informáticas científicas (Tim, 2021).

Angular v7.3.3

Es un framework para aplicaciones web desarrollado en TypeScript y de código abierto. Se utiliza para crear y mantener aplicaciones web, su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC). Angular te permite controlar la escalabilidad de las aplicaciones web que tienen su base en dicho framework (Camejo Domínguez, 2019).

Visual Studio Code v1.78.2

Visual Studio es un editor de código fuente desarrollado por Microsoft. Es software libre y multiplataforma, está disponible para Windows, GNU/Linux y macOS. VS Code tiene una buena integración con Git, cuenta con soporte para depuración de código, y dispone de un sinnúmero de extensiones, que básicamente te da la posibilidad de escribir y ejecutar código en cualquier lenguaje de programación (Microsoft, 2023).

Python v3.10.8

Fue escogido este lenguaje de programación pues es poderoso y fácil de aprender. Cuenta con estructuras de datos eficientes y de alto nivel y un enfoque simple pero efectivo a la programación orientada a objetos. La elegante sintaxis de Python y su tipado dinámico, junto con su naturaleza interpretada, hacen de éste un lenguaje ideal para scripting y desarrollo rápido de aplicaciones en diversas áreas y sobre la mayoría de las plataformas (Peng, 2021).

TypeScript v3.2.4

TypeScript es un lenguaje de programación que facilita la escritura multiplataforma, escala de aplicaciones, se ejecuta en cualquier navegador o en cualquier host, es un superconjunto mecanografiado de JavaScript, puede mejorar drásticamente su productividad al permitir experiencias de herramientas enriquecidas, todo mientras mantiene su código existente y continúa utilizando las mismas bibliotecas de JavaScript (Microsoft, 2023).

PostgreSQL v14

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto que usa y amplía el lenguaje SQL combinado con muchas características que almacenan y

escalan de manera segura las cargas de trabajo de datos más complicadas. Sus orígenes se remontan a 1986 como parte del proyecto POSTGRES en la Universidad de California en Berkeley y tiene más de 30 años de desarrollo activo en la plataforma central (Martin, 2014).

1.4 Metodología de desarrollo de software AUP-UCI

Es una variante metodológica para la actividad productiva de la UCI, basada en la metodología AUP en unión con el modelo CMMI-DEV v1.3. Aplica técnicas ágiles entre las que se incluye el modelado ágil, el desarrollo dirigido por pruebas y la gestión de cambios ágil. Es el elegido para este proyecto pues está definido para las características específicas de los centros de producción de la universidad. Se utilizará el escenario No. 4 que consiste en modelar el sistema con Historias de Usuario cuando no se realice modelado de negocio (Sánchez, 2015).

1.5 Conclusiones del capítulo

Al finalizar este capítulo, se concluye que fueron abordados los principales conceptos referentes a la gestión de la información y los procesos eleccionarios. Fueron analizados además diferentes sistemas, tanto nacionales como internacionales, para determinar algunas de las características necesarias para el software que se desea desarrollar. Se plasmaron las tecnologías que serán utilizadas teniendo en cuenta el soporte que recibirán estas en los próximos años garantizando la seguridad y la integridad de la información, además la metodología de desarrollo utilizada en el desarrollo de este módulo.

CAPÍTULO 2: DISEÑO DE LA PROPUESTA DE SOLUCIÓN

El siguiente capítulo abordará lo relacionado con el diseño del módulo, el mismo incluye especificación de los requisitos, tanto funcionales como no funcionales, las historias de usuario con sus respectivos prototipos, la representación y características de la arquitectura escogida para el proyecto así como los diagramas a confeccionar de acuerdo al escenario escogido para dicho módulo, con el fin de crear una visión clara de las funciones que el software debe realizar una vez puesto en funcionamiento.

2.1 Modelo Conceptual

Un Modelo de Dominio contiene principalmente los conceptos y sus relaciones que sean significativos en el dominio del problema. También se les conoce como modelos conceptuales, modelos de objetos del dominio o modelos de objetos de análisis. Estos modelos reflejan el entendimiento de las entidades del mundo real, así como sus relaciones y responsabilidades dentro del ámbito del problema en cuestión. Por lo tanto, estos modelos pueden formar parte de los requisitos a nivel del sistema y a menudo se capturan en forma de casos de uso, historias de usuario u otros medios (Sayago, 2023). Se realiza cuando no se logra determinar el proceso del negocio con fronteras bien definidas y donde los flujos de información son difusos, cuando existe solapamiento de responsabilidades, así como múltiples responsabilidades (Andrea, 2019).

En la siguiente figura se muestra el modelo de dominio del Módulo Nominaciones:

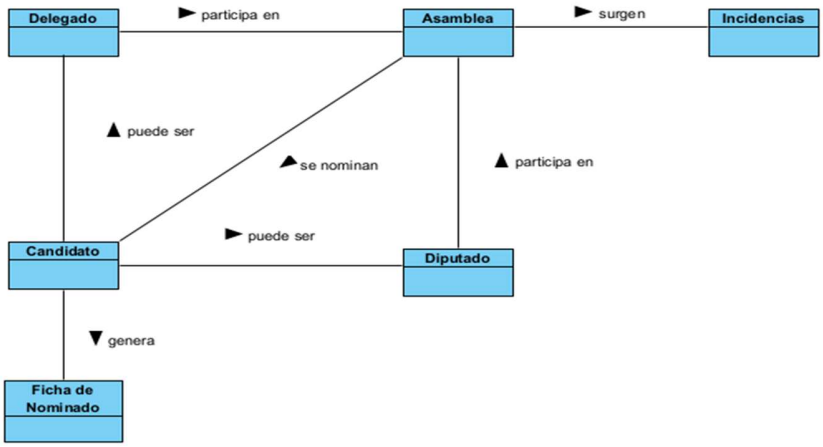


Figura 1: Modelo de Dominio. (Elaboración propia)

2.1.1 Descripción de Conceptos

Candidato: Persona que opta a un cargo, premio o distinción a solicitud propia o de otra persona.

Diputado: Son las personas que ocupan una diputación, en su carácter de representantes del pueblo, los que tendrán derecho de opinar, discutir, defender sus ideas y los intereses que representan.

Delegado: es quien se encarga de representar a una colectividad o a una persona o de actuar en su nombre.

Asamblea: es la denominación genérica del órgano representativo de los miembros de una organización o institución que toma decisiones.

Incidencias: acontecimientos que sobreviene en el curso de un asunto o negocio y tiene con él alguna explicación.

Ficha de nominado: formulario en el que se almacenan los datos de la persona nominada.

2.2 Especificación de Requisitos

La ingeniería de requisitos proporciona el mecanismo apropiado para entender lo que desea el cliente y de esta forma analizar sus necesidades, estableciendo de esta forma una sólida base

para el diseño y la implementación. La ingeniería de requisitos debe adaptarse a las necesidades del proceso, el proyecto, el producto y del personal que participa en el trabajo, teniendo en cuenta de que todas las tareas se realizan de forma iterativa mientras el equipo de desarrollo y los stakeholders continúan intercambiando información (Pressman, 2019).

El Módulo Nominaciones cuenta con un total de 51 requisitos funcionales y 14 no funcionales detallados a continuación:

2.2.1 Requisitos Funcionales

Los requisitos funcionales son enunciados acerca de servicios que el sistema debe proveer, de cómo debería reaccionar el sistema a entradas particulares y de cómo debería comportarse en situaciones específicas. Estos son comprobables y como regla general son una declaración de obligación (Sommerville, 2021). A continuación, se muestra una tabla con los requisitos identificados, con su descripción, prioridad, complejidad y agrupados:

Tabla 2: Requisitos funcionales.

Módulo Nominaciones				
ID	Nombre	Descripción	Prioridad	Complejidad
Módulo Nominaciones. Gestionar incidencias				
RF1	Listar incidencias	El sistema debe permitirle al usuario visualizar el listado de las incidencias, mostrándose los campos: Reprogramada, Tipo de incidencia, Detalles y Fecha de incidencia y Fecha de Reprogramación.	Alta	Baja
RF2	Registrar incidencias	El sistema debe permitirle al usuario la posibilidad de registrar una nueva incidencia en el nivel Municipio, mostrándose los siguientes campos: Reprogramada, Tipo de incidencia, Detalles, Fecha de incidencia y Fecha de Reprogramación.	Alta	Media
RF3	Visualizar detalles de incidencias	El sistema debe permitirle al usuario visualizar los detalles de la incidencia seleccionada, mostrándose los campos: Reprogramada, Tipo de incidencia, Detalles, Fecha de incidencia y Fecha de Reprogramación.	Baja	Baja
RF4	Modificar incidencias	El sistema debe permitirle al usuario modificar la incidencia seleccionada, mostrándose los campos: Reprogramada, Tipo de incidencia, Detalles, Fecha de la incidencia y	Alta	Meda

		Fecha de Reprogramación.		
RF5	Eliminar incidencias	El sistema debe permitirle al usuario eliminar la incidencia seleccionada.	Media	Baja
RF6	Buscar incidencias	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Reprogramada, Tipo de incidencia, Detalles, Fecha de incidencia y Fecha de Reprogramación.	Media	Baja
Módulo Nominaciones. Gestionar lugares a exponer y expuestos				
RF7	Listar lugares a exponer y expuestos	El sistema debe permitirle al usuario visualizar el listado de los lugares a exponer y expuestos, mostrándose los campos: Circunscripción, Cantidad de lugares a exponer, Cantidad de lugares expuestos, Cantidad de lugares pendientes por exponer, Detalles, Estado de aprobación, y los campos Municipio y Provincia para los niveles Municipio con distrito, Provincia y Nación según corresponda.	Alta	Baja
RF8	Registrar/Modificar lugares a exponer y expuestos	El sistema debe permitirle al usuario Registrar/Modificar los lugares a exponer y expuestos de la circunscripción seleccionada.	Alta	Media
RF9	Visualizar lugares a exponer y expuestos	El sistema debe permitirle al usuario visualizar los detalles de los lugares a exponer y expuestos de una circunscripción, mostrándose los campos: Circunscripción, Cantidad de lugares a exponer, Cantidad de lugares expuestos, Cantidad de lugares pendientes por exponer, Detalles, Estado de aprobación, y los campo Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Baja	Baja
RF10	Enviar lugares a exponer y expuestos	El sistema debe permitir al usuario Municipio enviar una o varias circunscripciones con los lugares a exponer y expuestos. Cambiando el estado de En elaboración a Pendiente.	Media	Baja
RF11	Buscar lugares a exponer y expuestos	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Circunscripción, Cantidad de lugares a exponer, Cantidad de lugares expuestos, Cantidad de lugares pendientes por exponer, Detalles, Estado de aprobación, y los campos Municipio y Provincia para los niveles Provincia y Nación según	Media	Baja

		corresponda.		
RF12	Aprobar lugares a exponer y expuestos	El sistema debe permitir al usuario Municipio y Provincia aprobar una o varias circunscripciones con los lugares a exponer y expuestos. Cambiando el estado de: · Pendiente municipio a Pendiente provincia para el caso del usuario Municipio. · Pendiente provincia a Aprobado para el caso del usuario Provincial.	Media	Baja
RF13	Rechazar lugares a exponer y expuestos	El sistema debe permitir al usuario municipio, provincial y nacional rechazar una o varias circunscripciones los lugares a exponer y expuestos. Cambiando el estado de: · Pendiente municipio a En Elaboración con nota de rechazo para el caso del usuario Municipio. · Pendiente provincia a En Elaboración con nota de rechazo para el caso del usuario Provincial. · Aprobado a En Elaboración con nota de rechazo para el caso del usuario Nacional.	Media	Baja
Módulo Nominaciones. Visualizar incidencias de lugares a exponer y expuestos				
RF14	Listar incidencias de los lugares a exponer y expuestos	El sistema debe permitirle al usuario visualizar el listado de las incidencias de los lugares a exponer y expuestos mostrándose los campos: Circunscripción, Incidencias, Detalles y Fecha. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Alta	Baja
RF15	Visualizar detalles de las incidencias de los lugares a exponer y expuestos	El sistema debe permitirle al usuario visualizar los detalles de una de las incidencias de los lugares a exponer y expuestos mostrándose los campos: Circunscripción, Incidencias, Detalles y Fecha. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Baja	Baja
RF16	Buscar incidencias de los lugares a exponer y expuestos	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Circunscripción, Incidencias, Detalles y Fecha. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Media	Baja
Módulo Nominaciones. Gestionar visita de los candidatos				
RF17	Listar visita de los	El sistema debe permitirle al usuario visualizar el listado de	Alta	Baja

	candidatos	los datos referentes a la visita de los candidatos a los centros laborales, mostrándose los campos: Circunscripción, Fecha prevista, Lugares e Incidencias. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.		
RF18	Registrar visita de los candidatos	El sistema debe permitirle al usuario registrar los datos referentes a la visita de los candidatos a los centros laborales mostrándose los siguientes campos: Circunscripción, Fecha prevista, Lugares e Incidencias.	Alta	Media
RF19	Modificar visita de los candidatos	El sistema debe permitirle al usuario modificar los datos referentes a la visita de los candidatos a los centros laborales mostrándose los siguientes campos: Circunscripción, Fecha prevista, Lugares e Incidencias. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Alta	Baja
RF20	Visualizar visita de los candidatos	El sistema debe permitirle al usuario visualizar los detalles de las visitas de los candidatos a los centros laborales, mostrándose los campos: Circunscripción, Fecha prevista, Lugares e Incidencias. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Baja	Baja
RF21	Buscar visita de los candidatos	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Circunscripción, Fecha prevista, Lugares e Incidencias. Se muestran además los campos: Municipio y Provincia para los niveles, Provincia y Nación según corresponda.	Media	Baja
RF22	Importar visita de los candidatos	El sistema permite cargar un archivo en formato csv donde se encuentran registradas todas las fechas de las visitas a los centros laborales.	Media	Baja
Módulo Nominaciones. Gestionar ficha de nominado				
RF23	Listar ficha de nominado	El sistema debe permitirle al usuario visualizar el listado de las fichas de nominados de los diputados y delegados, mostrándose los campos: Circunscripción, Primer Nombre, Segundo nombre, Primer apellido, Segundo apellido, Carnet de identidad, Ficha confeccionada, Ficha entregada, Ficha tomada, Ficha impresa y Estado de aprobación. Se muestran además los campos: Municipio y	Alta	Baja

		Provincia para los niveles Provincia y Nación según corresponda.		
RF24	Modificar ficha de nominado	El sistema debe permitirle al usuario modificar las fichas de nominados de los diputados y delegados mostrándose los siguientes campos: Circunscripción, Nombre, Segundo nombre, Primer apellido, Segundo apellido, Carnet de identidad, Ficha confeccionada, Ficha entregada, Ficha tomada, Ficha impresa y Estado de aprobación. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Alta	Media
RF25	Visualizar ficha de nominado	El sistema debe permitirle al usuario visualizar las fichas de nominados de los diputados y delegados, mostrándose los campos: Circunscripción, Primer Nombre, Segundo nombre, Primer apellido, Segundo apellido, Carnet de identidad, Ficha confeccionada, Ficha entregada, Ficha tomada, Ficha impresa y Estado de aprobación. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Baja	Baja
RF26	Enviar ficha de nominado	El sistema debe permitir al usuario Municipio enviar una o varias circunscripciones con las fichas de nominados de los diputados y delegados. Cambiando el estado de En elaboración a Pendiente provincia.	Media	Baja
RF27	Aprobar ficha de nominado	El sistema debe permitir al usuario Municipio y Provincia aprobar una o varias circunscripciones con las fichas de nominados de los diputados y delegados. Cambiando el estado de: · Pendiente municipio a Pendiente provincia para el caso del usuario municipio. · Pendiente provincia a Aprobado para el caso del usuario provincial.	Media	Baja
RF28	Rechazar ficha de nominado	El sistema debe permitir al usuario Municipio, Provincial y Nacional rechazar una o varias circunscripciones con las fichas de nominados de los diputados y delegados. Cambiando el estado de: · Pendiente municipio a En Elaboración con nota de rechazo para el caso del usuario Municipio. · Pendiente provincia a En Elaboración con nota de	Media	Baja

		rechazo para el caso del usuario Provincial. · Aprobado a En Elaboración con nota de rechazo para el caso del usuario Nacional.		
RF29	Buscar ficha de nominado	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Circunscripción, Nombre, Segundo nombre, Primer apellido, Segundo apellido, Carnet de identidad, Ficha confeccionada, Ficha entregada, Ficha tomada, Ficha impresa y Estado de aprobación. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Media	Baja
Módulo Nominaciones. Gestionar realización de asamblea				
RF30	Registrar/Modificar informe de realización de asamblea	El sistema debe permitirle al usuario Registrar/Modificar informe de realización de asamblea mostrándose los siguientes campos: Asistencia planificada, Asistencia real y Estado de aprobación.	Alta	Alta
RF31	Enviar informe de realización de asamblea	El sistema debe permitirle al usuario la posibilidad de enviar el Informe de la realización de la asamblea cambiando el estado de En elaboración a Pendiente a provincia.	Alta	Baja
RF32	Listar informe de realización de asamblea	El sistema debe permitirle al usuario visualizar el listado de los datos referentes al informe de realización de asamblea, mostrándose los campos: Asistencia planificada, Asistencia real y Estado de aprobación. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Alta	Baja
RF33	Visualizar informe de realización de asamblea	El sistema debe permitirle al usuario visualizar el informe de realización de asamblea, mostrándose los campos: Asistencia planificada, Asistencia real y Estado de aprobación. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Baja	Baja
RF34	Buscar informe de realización de asamblea	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Asistencia planificada, Asistencia real y Estado de aprobación. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Media	Baja

RF35	Rechazar informe de realización de asamblea	El sistema debe permitir al nivel provincial y nacional rechazar el informe de realización de asamblea. Cambiando el estado de: • Pendiente provincia a En Elaboración con nota de rechazo para el caso del usuario provincial. • Aprobado a En Elaboración con nota de rechazo para el caso del usuario nacional.	Media	Baja
RF36	Aprobar informe de realización de asamblea	El sistema debe permitir al usuario provincia aprobar el informe de realización de asamblea. Cambiando el estado de: · Pendiente provincia a Aprobado para el caso del usuario provincial.	Media	Baja
Módulo Nominaciones. Gestionar nominados a delegados				
RF37	Listar nominados a delegados	El sistema debe permitirle al usuario listar los nominados a delegados mostrándose los campos: Nombre y Apellidos, Número de identidad, Variantes y Cantidad de votos Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Alta	Baja
RF38	Buscar nominados a delegados	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Nombre y Apellidos, Número de identidad, Variantes y Cantidad de votos. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Media	Baja
RF39	Visualizar nominados a delegados	El sistema debe permitirle al usuario visualizar los nominados a delegados, mostrándose los campos: Nombre y Apellidos, Número de identidad, Variantes y Cantidad de votos. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Baja	Baja
RF40	Guardar cambios	El sistema debe permitirle al usuario guardar los cambios de modificación de la cantidad de votos de los nominados a delegados mostrándose los siguientes campos: Nombre y Apellidos, Número de identidad, Variantes y Cantidad de votos.	Alta	Media
Módulo Nominaciones. Gestionar nominados a diputados				

RF41	Listar nominados a diputados	El sistema debe permitirle al usuario listar los nominados a diputados mostrándose los campos: Nombre y Apellidos, Número de identidad, Variantes y Cantidad de votos. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Alta	Baja
RF42	Buscar nominados a diputados	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Nombre y Apellidos, Número de identidad, Variantes y Cantidad de votos. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Media	Baja
RF43	Visualizar nominados a diputados	El sistema debe permitirle al usuario visualizar los nominados a diputados, mostrándose los campos: Nombre y Apellidos, Número de identidad, Variantes y Cantidad de votos. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.	Baja	Baja
RF44	Guardar cambios	El sistema debe permitirle al usuario guardar los cambios de modificación de la cantidad de votos de los nominados a diputados mostrándose los siguientes campos: Nombre y Apellidos, Número de identidad, Variantes y Cantidad de votos.	Alta	Media
Módulo Nominaciones. Gestionar propuesta de candidatos a la ANPP				
RF45	Registrar propuesta de precandidato a la ANPP	El sistema debe permitirle al usuario la posibilidad de registrar una propuesta de precandidato a la ANPP en el nivel Nación, mostrándose los siguientes campos: Provincia, Municipio, Nombre y Apellidos, Número de identidad y Variantes.	Alta	Media
RF46	Listar propuesta de precandidato a la ANPP	El sistema debe permitirle al usuario listar las propuestas de precandidato a la ANPP mostrándose los campos: Provincia, Municipio, Nombre y Apellidos, Número de identidad y Variantes.	Alta	Baja
RF47	Buscar propuesta de precandidato a la ANPP	El sistema debe permitirle al usuario realizar una búsqueda filtrando por los siguientes campos: Provincia, Municipio, Nombre y Apellidos, Número de identidad y Variantes.	Media	Baja
RF48	Modificar	El sistema debe permitirle al usuario modificar las	Alta	Media

	propuesta de precandidato a la ANPP	propuestas de precandidato a la ANPP mostrándose los siguientes campos: Nombre y Apellidos, Número de identidad y Variantes.		
RF49	Visualizar propuesta de precandidato a la ANPP	El sistema debe permitirle al usuario visualizar las propuestas de precandidato a la ANPP, mostrándose los campos: Provincia, Municipio, Nombre y Apellidos, Número de identidad y Variantes.	Baja	Baja
RF50	Eliminar propuesta de precandidato a la ANPP	El sistema debe permitirle al usuario eliminar la propuesta de precandidato a la ANPP seleccionada.	Media	Baja
RF51	Validar propuesta de precandidato a la ANPP	El sistema debe permitirle al usuario validar la propuesta de precandidato a la ANPP seleccionada.	Media	Baja

2.2.2 Requisitos No Funcionales

Los requisitos no funcionales son limitaciones sobre servicios o funciones que ofrece el sistema. Incluyen restricciones tanto de tiempo y desarrollo, como las impuestas por los estándares. Los requerimientos no funcionales se suelen aplicar al sistema como un todo, más que a características o a servicios individuales del sistema (Sommerville, 2021). A continuación, se muestran los requisitos no funcionales clasificados según Sommerville:

Tabla 3: Requisitos del producto**:**

Usabilidad:	RNF 1	El sistema deberá ser una aplicación web.
	RNF 2	El sistema deberá mantener una interfaz semejante a la versión anterior para cumplir con el principio de familiaridad con el usuario.
Seguridad:	RNF 3	Para acceder a las funcionalidades del sistema el usuario deberá estar correctamente autenticado.

	RNF 4	Se crearán usuarios con varios roles para gestionar los diferentes niveles de acceso a la aplicación.
	RNF 5	Los usuarios estarán conectados al sistema dentro de la red nacional.
Eficiencia:	RNF 6	En el servidor donde se va a desplegar la aplicación se recomienda que tenga 8 GB de memoria RAM y un procesador de 3.2 GHz con 4 núcleos como mínimo.
	RNF 7	Las computadoras clientes deberán tener 4 GB de memoria RAM y un procesador de 2.4 GHz como mínimo además de que es necesario utilizar el navegador web Mozilla Firefox a partir de la versión 80 en adelante.
	RNF 8	El servidor donde se va a desplegar la aplicación debe tener como mínimo un disco duro de 1 TB de almacenamiento.
Rendimiento:	RNF 9	El servidor donde se va a desplegar la aplicación debe ser capaz de procesar como mínimo 50 solicitudes por segundo.
Portabilidad:	RNF 10	El sistema se podrá usar indistintamente en los sistemas operativos derivados del Nova 2015 o Nova Kiosco.

Tabla 4: Requisitos de la organización**:**

Estándares:	RNF 11	Se debe utilizar como metodología de desarrollo la AUP en su versión UCI.
Desarrollo:	RNF 12	Se utilizará como marco de trabajo para la implementación de las interfaces visuales Angular v7.3.3
	RNF 13	Se utilizará como marco de trabajo para la implementación de la aplicación Django v3.8 con lenguaje de programación Python v3.10.8

Tabla 5: Requisitos externos**:**

Legales:	RNF 14	El sistema debe estar acorde con la Ley Electoral No. 127

2.3 Historias de Usuario

Las historias de usuario se usan, en el contexto de la ingeniería de requisitos ágil, como una herramienta de comunicación que combina las fortalezas de ambos medios: escrito y verbal. Describen, en una o dos frases, una funcionalidad de software desde el punto de vista del usuario, con el lenguaje que éste emplearía (Menzinsky, y otros, 2020). A continuación, se muestran algunas historias de usuario(Véase anexos 1 y 2):

Tabla 6: Historia de Usuario RF4

Historia de Usuario	
Número:4	**Usuario:** Municipal
Nombre de Historia: Modificar Incidencias	
Prioridad en Negocio: Alta	**Riesgo en Desarrollo:** Alto
Tiempo estimado: 4 Horas	**Iteración Asignada:**1
Programador Responsable: Roberto Valladares Martín	
Descripción: El sistema debe permitirle al usuario modificar la incidencia seleccionada del listado, mostrándose los campos: Reprogramada, Tipo de incidencia, Detalles, Fecha de la incidencia y Fecha de Reprogramación.	

Prototipo elemental de interfaz gráfica de usuario:

Fuente: Manual de Usuario

Tabla 7: Historia de Usuario RF26

		Historia de Usuario
Número:26	**Usuario:** Nacional	
Nombre de Historia: Registrar propuesta de precandidato a la ANPP.		
Prioridad en Negocio: Alta	**Riesgo en Desarrollo:** Alto	
Tiempo estimado: 4 Horas	**Iteración Asignada:**2	
Programador Responsable: Roberto Valladares Martín		

El sistema debe permitir al usuario Municipio enviar una o varias circunscripciones con las fichas de nominados de los diputados y delegados, cambiando el estado de En elaboración a Pendiente provincia.

Prototipo elemental de interfaz gráfica de usuario:

Guardar cambios | Vista previa | Detalles

Primer nombre	Segundo nombre	Primer apellido	Segundo apellido	No. identidad	Ficha confeccionada	Ficha entregada	Foto tomada	Ficha impresa	Estado
Circunscripción: 2									
Barbara		Valdés	Pérez	69012105255	No	No	No	Si	Pendiente Provincia
José	Luis	Barrios	Martinez	54082504581	Si	Si	Si	Si	En elaboración
Circunscripción: 3									
Aracely		Delgado	Bosmenier	68090123076	Si	No	No	No	En elaboración
Orlando		Garrido	Boy	57100804440	No	No	No	No	En elaboración
Circunscripción: 4									
Heriberto		Miranda	Gutierrez	56110304427	Si	Si	No	Si	En elaboración
Norberto		Pérez	Herrera	65021405941	No	No	No	No	En elaboración
Circunscripción: 5									
Ida	Yraida	Ruiz	Valdés	63041304955	No	No	No	Si	En elaboración
Samuel		Rodriguez	Fernández	73091003743	No	No	No	No	En elaboración
Circunscripción: 6									
Hortencia		Hernández	Miranda	51111403910	No	No	No	No	En elaboración

Fuente: Manual de Usuario

2.4 Arquitectura de software

La arquitectura del software representa la estructura de los datos y los componentes que se requieren para construir un programa o sistema de cómputo. Considera el estilo arquitectónico que tomará el sistema, la estructura y las propiedades de los componentes que constituyen el sistema, además de las interrelaciones que ocurren entre todos estos componentes (Pressman, 2019).

Dada las características del proyecto decidimOS y los framework a utilizar se define como arquitectura a emplear en el FrontEnd, el llamado Modelo-Vista-Controlador, esquema de capas muy utilizado en el desarrollo de software basado en aplicaciones web (Garzón, 2019). **El modelo:** es un conjunto de clases que representan la información del mundo real que el

sistema debe procesar, **las vistas:** son el conjunto de clases que se encargan de mostrar al usuario la información contenida en el modelo. Una vista está asociada a un modelo, pudiendo existir varias vistas asociadas al mismo modelo, **el controlador:** es un objeto que se encarga de dirigir el flujo del control de la aplicación debido a mensajes externos, como datos introducidos por el usuario u opciones del menú seleccionadas por él. A partir de estos mensajes, el controlador se encarga de modificar el modelo o de abrir y cerrar vistas.

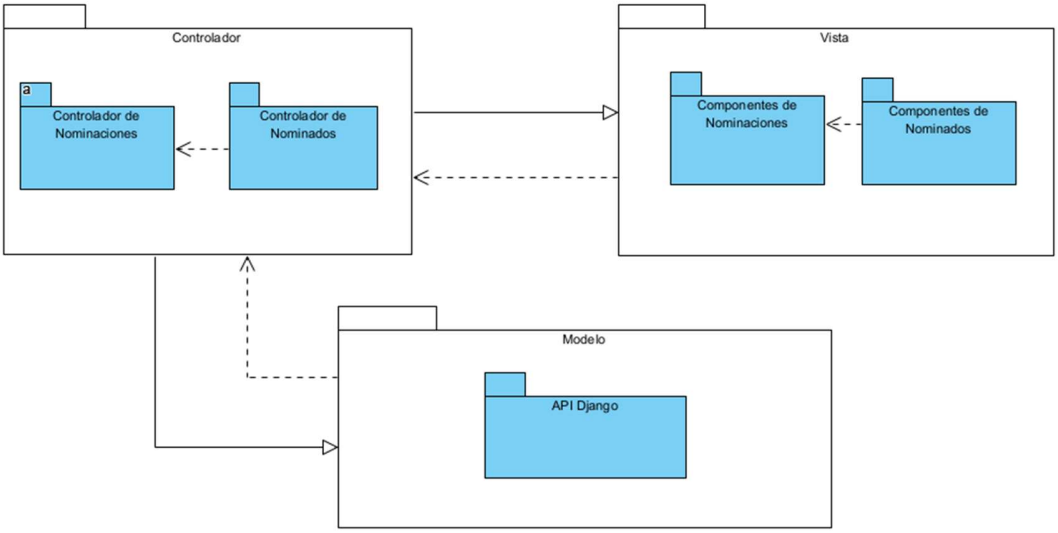


Figura 2: Patrón Modelo Vista Controlador. (Elaboración propia)

El framework Django, utilizado para el BackEnd, facilita el desarrollo de aplicaciones web con el uso del patrón Modelo-Vista-Plantilla, descritas a continuación, **el modelo:** es el responsable de manejar la parte lógica de la aplicación web y de cómo se almacena la información en la base de datos, l**as vistas:** estas encapsulan la lógica necesaria para procesar las peticiones de los usuarios y devolver una respuesta y **las plantillas:** son las que proveen de una sintaxis de diseño amigable para renderizar la información que va a ser presentada al usuario (K., 2020).

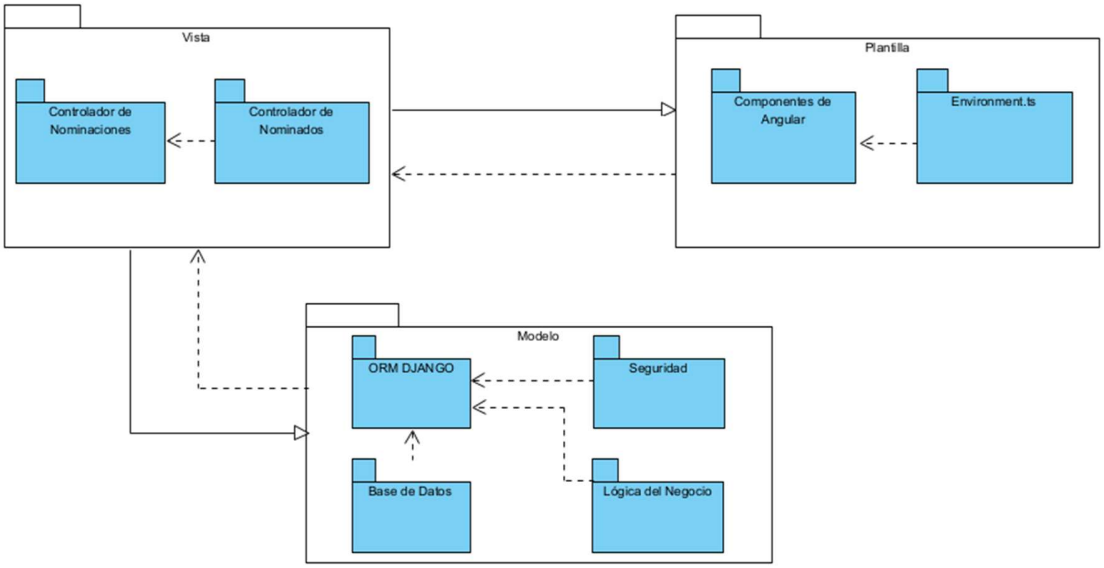


Figura 3: Patrón Modelo Vista Plantilla. (Elaboración propia)

2.5 Patrones del diseño

Los patrones de diseño son una solución general, reutilizable y aplicable a diferentes problemas de diseño de software. Se trata de plantillas que identifican problemas en el sistema y proporcionan soluciones apropiadas a problemas generales a los que se han enfrentado los desarrolladores durante un largo periodo de tiempo, a través de prueba y error. A continuación, se describen los patrones de diseño GRASP Y GOF utilizados. (Jan, 2020)

Patrones GRASP

Estos patrones describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. A continuación, se muestran algunos de los patrones GRASP identificados.

Patrón Controlador: Se emplea en la clase responsable de recibir o manejar un evento del sistema, definiendo para ello el método de operación. Su uso se pone de manifiesto en la clase SP_Controlador la cual es la encargada de interactuar con el modelo y enviar la respuesta a la vista.

Figura 4: Patrón Controlador. (Elaboración Propia)

Bajo acoplamiento: El objetivo es tener las clases lo menos relacionadas entre sí, previniendo que, en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de las clases. Esto potencia la reutilización y la disminución de la dependencia entre las clases. Se evidencia en la misma estructura del framework Django.

Alta cohesión: Se aplica en la mayoría de las clases del diseño, ya que en cada una solo se implementan las funcionalidades que le corresponden, en este caso, los formularios se enfocan totalmente en realizar la captura de información dentro del sistema.

Figura 5: Patrón Alta Cohesión. (Elaboración Propia)

Patrones GOF

Los patrones GOF son aquellos que describen soluciones simples y elegantes a problemas específicos en el diseño de software orientado a objetos.

Patrón observador: Este patrón está presente en las clases de Angular las cuales están pendientes de validar la información que introduce el usuario y en caso de que esta no está correcta se notificaran en los formularios.

```
public submit() {
  this.reprogrammedAssemblyService.urlResource = '/municipales/reprogrammedAssembly' + this.idReprogrammedAssembly;
  this.reprogrammedAssemblyService.editReprogrammedAssembly(this.reprogrammedAssembly)
    .subscribe(() => {
      this.reprogrammedAssemblyService.urlResource = '/part';
      this.notificationService.success('DOMAIN.MSG.ASAMBLEA_REPROGRAMADA_MODIFICADA');
      this.onAcept.emit();
    },
      (error: any) => {
        console.log(error);
      });

}
```

Figura 6: Patrón Observador. (Elaboración Propia)

Patrón decorador: Se evidencia en las vistas de Angular las cuales contienen estructuras y diseños comunes para todo el sistema heredando Módulos de manera dinámica y decorándola con contenido propio del mismo.

```
@Component({
  selector: 'app-list-nominee-file',
  templateUrl: './list-nominee-file.component.html',
  styleUrls: ['./list-nominee-file.component.sass']
})
export class ListNomineeFileComponent implements OnInit {
```

Figura 7: Patrón Decorador. (Elaboración Propia)

2.6 Modelo Entidad Relación

El diagrama entidad relación es una herramienta para el modelo de datos que permite representar las entidades relevantes de un sistema de información (Nur, 2023). Representa a la realidad a través de un esquema gráfico donde se manifiestan los elementos que posteriormente se debe de implementar en un gestor de base de datos, así como las relaciones

entre ellos (Ritchell, 2020). A continuación, un fragmento del modelo de datos del ecosistema decidimOS:

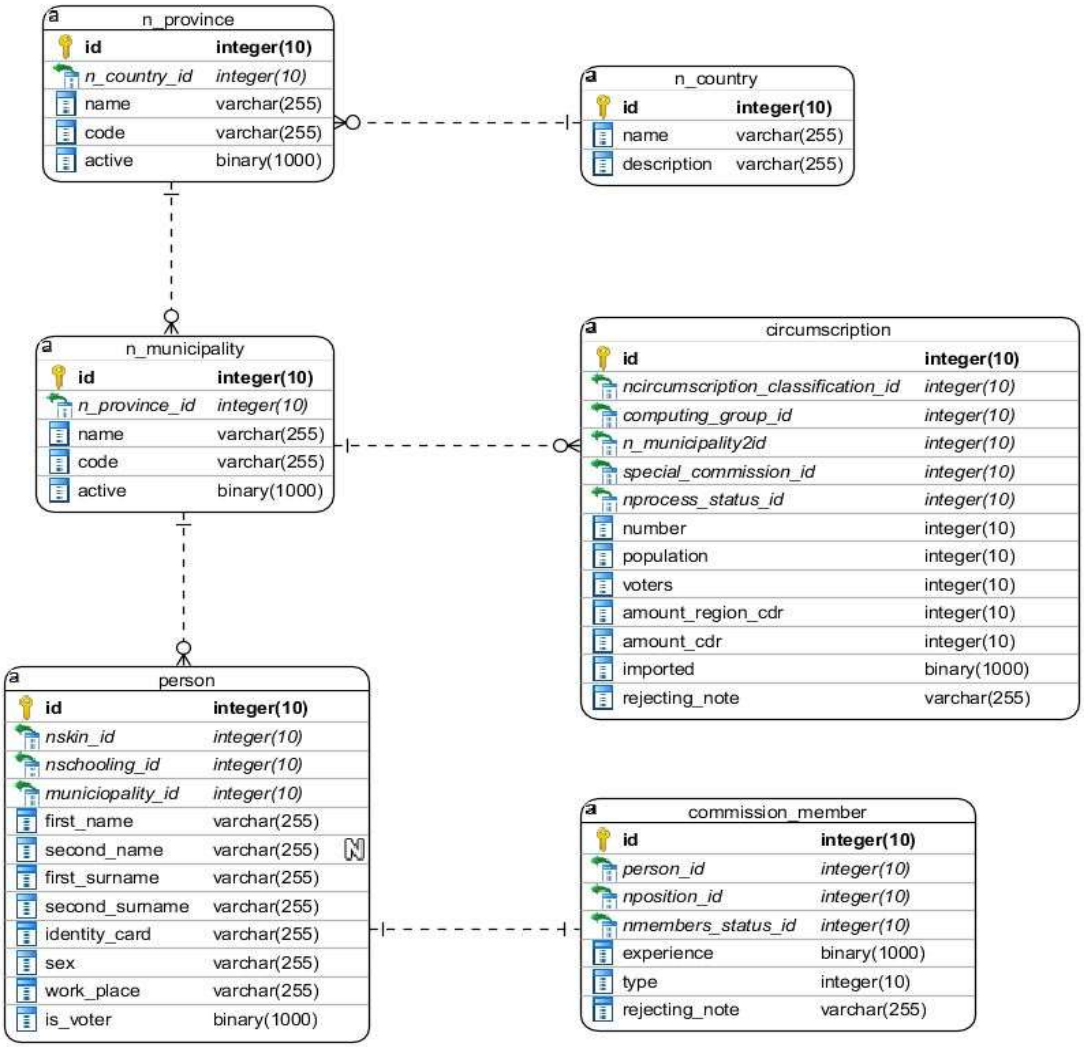


Figura 8: Modelo de Datos. (Elaboración Propia)

2.7 Conclusiones del capítulo

Luego de realizar el proceso de análisis y diseño del Módulo Nominaciones se puede concluir que gracias al proceso de especificación de los requisitos se recopilaron un total de 13 requisitos no funcionales y 51 funcionales, de cada uno de estos últimos se crearon las correspondientes historias de usuario con los respectivos prototipos de cada uno de ellos. Se definieron como arquitecturas de software el Modelo Vista Controlador para el FrontEnd y el

Modelo Vista Plantilla para el BackEnd, con los cuales se pudo identificar los Módulos que participarán en el desarrollo de la aplicación. Se identificaron los patrones del diseño empleados para una mejor confección del sistema.

CAPÍTULO 3: VALIDACIÓN DEL MÓDULO NOMINACIONES

En el siguiente capítulo se realizarán las pruebas al módulo para validar su correcto funcionamiento, como pruebas de integración y pruebas de rendimiento, además se tendrá en cuenta la aceptación por parte del cliente.

3.1 Estrategia de pruebas

Una estrategia de pruebas de software es una guía que proporciona una estrategia general y un procedimiento que define pasos de prueba específicos y los tipos de prueba que van a ser utilizados (Pressman, 2019).

Pressman define 4 niveles de prueba, estos son: pruebas de unidad, pruebas de integración, pruebas de validación y pruebas de sistema. Para efectuar estas pruebas se van utilizar herramientas automatizadas para obtener resultados con una mayor agilidad.

3.1.1 Prueba Unitaria

Las pruebas unitarias se realizan para comprobar el correcto funcionamiento de una porción de código, y determinar si cumple con los requisitos pertinentes; para el desarrollo de esta prueba se hizo uso del módulo Unittest escrito en Python para probar el funcionamiento unitario de la clase Nominee y que arrojó el siguiente resultado:

```
test_nominee.py U
decidimos > tests > test_nominee.py > ...
1   from django.test import TestCase
2   from municipales.modelos.NomineeModel import Nominee
3   from unittest import TestCase
4
5   class NomineeTestCase(TestCase):
6       def setUp(self):
7           self.nominee = Nominee.objects.create(
8               id='17',
9               votes='3',
10              elect=False,
11              type='1'
12          )
13      def test_nominee(self):
14          self.assertEqual(self.nominee.id, "17")
15          self.assertEqual(self.nominee.votes,"3")
16          self.assertEqual(self.nominee.elect, False)
17          self.assertEqual(self.nominee.type,"1")
18
19

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

D:\Proyecto\decidimos>py manage.py test
System check identified no issues (0 silenced).
.
----------------------------------------------------------------------
Ran 1 test in 0.065s

OK

D:\Proyecto\decidimos>
```

Figura 9: Prueba Unitaria Crear Nominados. (Elaboración Propia)

Durante la primera iteración realizada, la herramienta no almacenaba correctamente la información en la base de datos por un problema en la definición de la clase. En la segunda iteración, este problema fue resuelto y la prueba concluyó satisfactoriamente.

3.1.2 Prueba de Integración

Para realizar la prueba de integración se va a utilizar el cliente HTTP Postman, el cuál es una aplicación que nos permite testear peticiones HTTP mediante una interfaz gráfica de usuario (Dimoski, 2022). La prueba se realizó mediante el uso de los métodos GET y POST, de donde se obtuvieron los siguientes resultados:

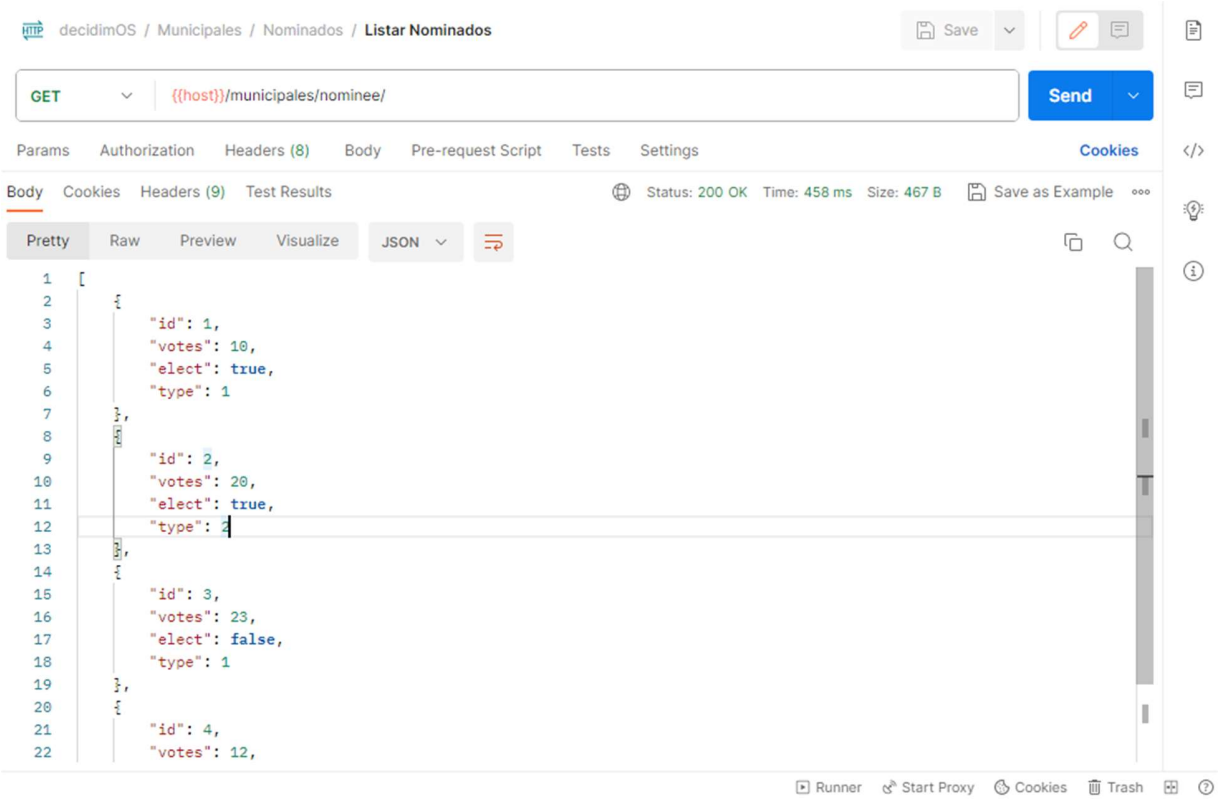


Figura 10: Prueba de Integración Listar Nominados. (Elaboración Propia)

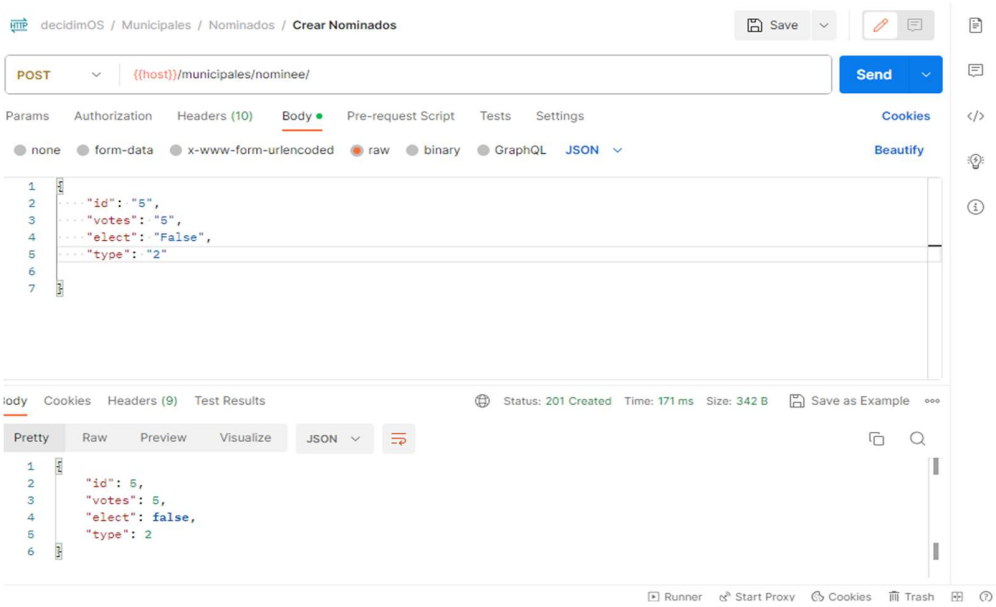


Figura 11: Prueba de Integración Crear Nominados satisfactoria. (Elaboración Propia)

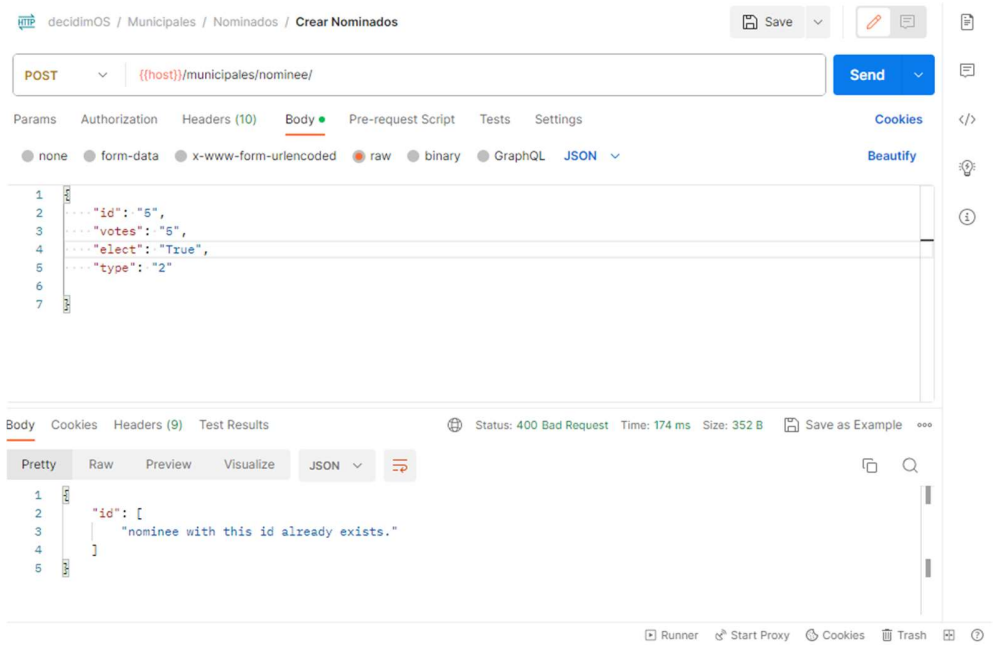


Figura 12: Prueba de Integración Crear Nominados fallida. (Elaboración Propia)

La herramienta devuelve la información que se desea en diferentes formatos de texto, en este caso se hizo uso del formato JSON, además la herramienta muestra el código de estado asociado a la petición. Dependiendo del rango que posea el código será la respuesta del servidor. El rango de los “200” se refiere a respuestas exitosas, el rango de los “400” indican

problemas con el cliente(navegador), mientras que los “500” son problemas relacionados con el servidor. En la primera iteración, el método GET devolvió correctamente la información, mientras que el método POST devolvió un error 400 debido a que ya existía un nominado con ese identificador. Esto fue solucionado en la segunda iteración donde se introdujo un identificador que no estuviera almacenado en la base de datos.

3.1.3 Prueba de Rendimiento

Las pruebas de rendimiento deben diseñarse para garantizar que el sistema procese una carga pretendida lo que implica efectuar una serie de pruebas donde se aumenta la carga, hasta que el rendimiento del sistema se vuelve inaceptable (Sommerville, 2021). Para evaluar el rendimiento del sistema se hizo uso de la herramienta JMeter, donde se simuló un total de 500 peticiones en un tiempo de 5 segundos y donde se arrojaron los siguientes datos:

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo | Navegar... | Log/Mostrar sólo: Escribir en Log Sólo Errores | Éxitos | Configurar

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
Listar Nominados	500	3038	183	4894	1629,25	52,80%	52,9/sec	98,01	6,75	1898,5
Total	500	3038	183	4894	1629,25	52,80%	52,9/sec	98,01	6,75	1898,5

Figura 13: Prueba de Rendimiento Listar Nominados Reporte Resumen. (Elaboración Propia)

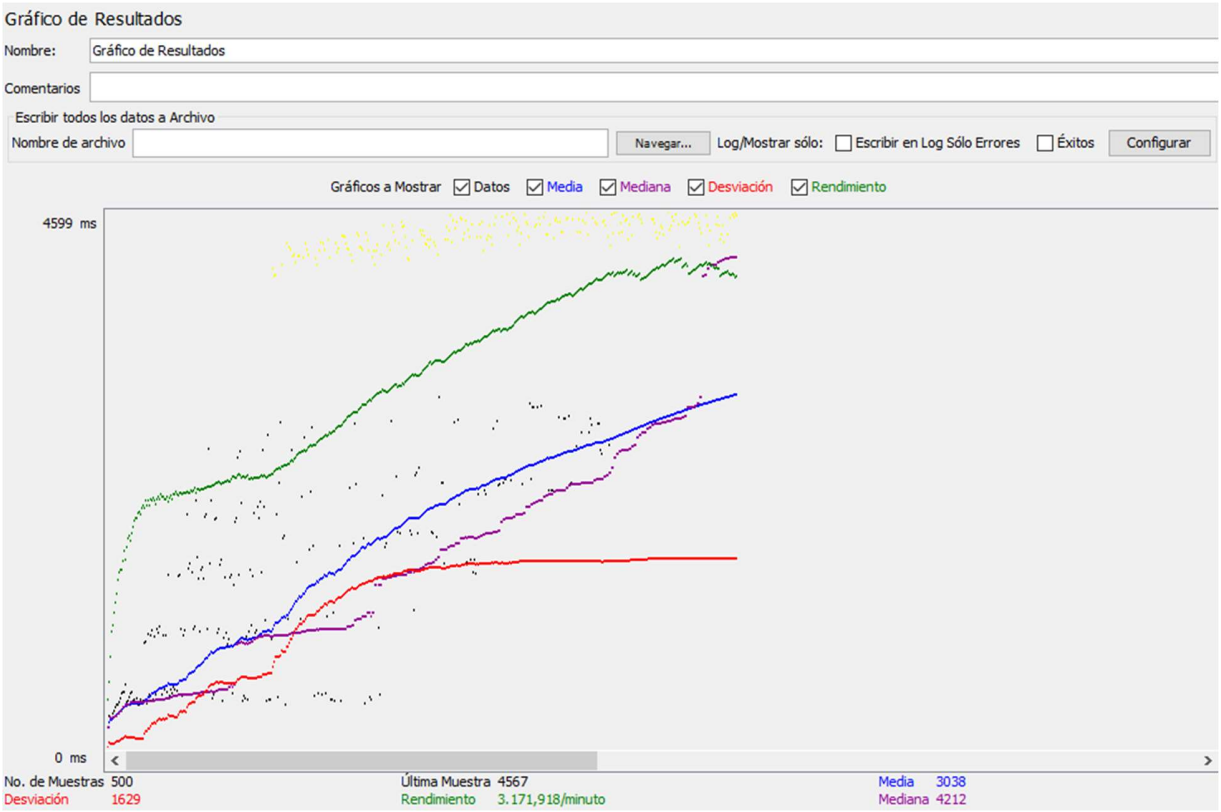


Figura 14: Prueba de Rendimiento Listar Nominados Gráfico de Resultados. (Elaboración Propia)

La herramienta devuelve un conjunto de datos, de los cuales se van a tener en cuenta los valores de la media(azul), la desviación estándar(roja) y el rendimiento(verde). Aunque el rendimiento promedio es razonablemente alto, existe una variabilidad significativa en los tiempos de respuesta o el rendimiento de la aplicación, lo que puede ocasionar solicitudes con retrasos inesperados o un rendimiento no uniforme. En cualquier caso, el valor del rendimiento es positivo para las condiciones de prueba actuales.

3.2 Conclusiones del capítulo

El desarrollo del presente capítulo permitió mediante la formulación de una estrategia de pruebas, la verificación del trabajo realizado en capítulos anteriores, así como la validación de sus resultados. De esta manera se detectaron y corrigieron errores durante las pruebas a distintos niveles con la ayuda de herramientas de automatización, mejorando a su vez el software para una posterior presentación al cliente, quien valoró y mostró su conformidad con el producto.

CONCLUSIONES GENERALES

- ✓ El estudio de los referentes teóricos y el análisis de las diferentes herramientas y tendencias para la gestión de la información de los procesos electorales permitió determinar la no existencia de un sistema informático que responda a las necesidades requeridas por el cliente y sentar una base sólida para el diseño e implementación del módulo Nominaciones.
- ✓ El análisis y diseño de la propuesta de solución permitió determinar un conjunto de requisitos funcionales, que posteriormente fueron modelados en historias de usuario y que al implementarse se obtuvo un módulo flexible, robusto y escalable.
- ✓ El uso de las herramientas y lenguajes de programación seleccionados permitió la implementación del módulo, y su posterior integración en la Suite decidimOS, facilitando así la gestión de la información referente a las nominaciones.
- ✓ El uso de una estrategia de pruebas permitió la detección y corrección de múltiples errores una vez implementado el módulo Nominaciones y que no garantizaba una correcta integración a la Suite decidimOS.

RECOMENDACIONES

Se propone mantener la vigilancia tecnológica sobre las principales herramientas y tecnologías utilizadas en la búsqueda de mejorar el rendimiento y la seguridad de la Suite decidimOS. Además, se propone el uso de técnicas de inteligencia artificial para validar ciertos datos que podrían no tener una coherencia lógica a la hora de ser insertados por el usuario final.

BIBLIOGRAFÍA

Andrea, Giorgini. 2019. *Well-posedness of a diffuse interface model for Hele-Shaw flows.* 2019.

Camacho González, L. M., Alvarez Luna, R., Collado Rolo, L., Coca Conesa, L. L., Rodríguez Pérez, L., Aguilar Dowins, O., & Castro Díaz, Y. 2021. *Ecosistema de software para la gestión de procesos electorales.* s.l. : UCIENCIA 2021, 2021.

Camejo Domínguez, R. 2019. *Sistema de gestión de productos y servicios para proveedores y consumidores empresariales utilizando la arquitectura de microservicios.* s.l. : Universidad de las Ciencias Informáticas. Empresa de Tecnologías de la Información para la Defensa. (Xetid), 2019.

Comisión de Asistencia Electoral de los EE. UU. 2023. *Glosario de terminología Electoral.* 2023.

Dimoski, D., Koteska, B., Pejov, L., & Mishev, A. 2022. *Testing RESTful APIs–Use Case: RESTful API for Solving Multidimensional Time–Independent Schrödinger Equation.* 2022.

Diomidis, Spinellis. 2010. *UML Everywhere.* s.l. : IEEE Software, 2010.

European Comission. 2020. european-democracy-action-plan_en. *commission.europa.eu.* [En línea] 2020. https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/new-push-european-democracy/european-democracy-action-plan_en.

Gaceta Oficial de la República de Cuba. 2019. *Ley No. 127 "Ley Electoral".* 2019.

Garzón, Cecilia Ávila. 2019. *Modelo Vista Controlador.* 2019.

GRANMA. 2015. la-uci-soporte-tecnologico-del-consejo-electoral-nacional. *granma.cu.* [En línea] 2015. http://www.granma.cu/cuba/2015-01-13/la-uci-soporte-tecnologico-del-consejo-electoral-nacional.

IEEE Software. 2022. *openEuler: Advancing a Hardware and Software Application Ecosystem.* 2022.

Jan, Marco, Leimeister., Jan, Marco, Leimeister., Ernestine, Dickhaut., Andreas, Janson. 2020. *Design Pattern as a Bridge Between Problem-Space and Solution-Space. Social Science Research Network.* 2020. pág. 32.

K., Manikanta, Vamsi., P., Lokesh., K., Neha, Reddy., P., Swetha. 2020. *Visualization of Real World Enterprise Data using Python Django Framework.* 2020.

Martin, Lindblom., Fredrik, Strandin. 2014. *Performance analysis and improvement of PostgreSQL.* 2014.

Menzinsky, Alexander, y otros. 2020. *Historias de Usuario.* s.l. : Scrum Manager, 2020.

Microsoft. 2023. Documentation for Visual Studio Code. */docs.* [En línea] 2023. https://code.visualstudio.com.

—. 2023. Documentation for TypeScript. */docs.* [En línea] 2023. https://www.typescriptlang.org.

Ministerio de Comunicaciones. 2018. tecnologias-de-la-informacion-y-la-comunicacion-tic-en-cuba. *mincom.gob.cu.* [En línea] 2018. http://www.mincom.gob.cu/es/tecnologias-de-la-informacion-y-la-comunicacion-tic-en-cuba/.

Muzammil, Shahbaz., Kashif, Saghar., Saifullah, Amin. 2021. *Unit Testing from Partial and Informal Requirements.* 2021.

Nur, Shaffiqa, Muhammad, Soffian., Amelia, Ahmad. 2023. *Hiking incidents database structure using entity-relationship diagram.* s.l. : Journal of Information Systems and Technology Management, 2023.

Peng, Yun., Yu, Zhang., Mingzhe, Hu. 2021. *An Empirical Study for Common Language Features Used in Python Projects.* 2021.

Pressman. 2019. *Software Engineering A practitioner´s approach.* 2019.

Prytherch, Ray. 2005. *HARROD'S LIBRARIANS' GLOSSARY AND REFERENCE BOOK.* 2005.

REAL ACADEMIA ESPAÑOLA: Diccionario de la lengua española. 2023. candidato. *dle.rae.* [En línea] 2023. https://dle.rae.es.

—. 2023. nominación. *dle.rae.* [En línea] 2023. https://dle.rae.es.

REAL ACADEMIA ESPAÑOLA: Diccionario panhispánico del español jurídico. 2023. elecciones. *dpej.rae.* [En línea] 2023. https://dpej.rae.es/.

—. 2023. proceso electoral. *dpej.rae.* [En línea] 2023. https://dpej.rae.es/.

Ritchell, S., Villafuerte., Deborah, G., Brosas., Dindo, C., Obediencia., Jessie, R., Paragas. 2020. *EnORS: An Enhanced Object Relationship Schema.* 2020.

Sánchez, Tamara Rodríguez. 2015. *Metodología de desarrollo para la Actividad productiva de la UCI.* 2015.

Sayago, S., & Pascual-Almenara, A. 2023. *Accesibilidad. Diseño para Todos.* 2023.

Simon, J., Bass, T., Boelman, V., & Mulgan, G. 2018. *Digital Democracy The tools transforming political engagement.* 2018.

Sommerville, Ian. 2021. *Ingenieria de Software.* 2021.

Tim, Ewing., Jillian, Redfern., Amanda, Alexander., Richard, Medina., Emma, Birath. 2021. *Django as a Mission Planning Tool Interface for the CYGNSS Mission.* 2021.

Torricella-Morales, R. G., Lee-Tenorio, F., & Carbonell-de-la-Fé, S. 2020. *Infotecnología: La cultura informacional para el trabajo en la Web.* s.l. : Editorial Universitaria(Cuba), 2020.

Visual Paradigm. 2023. what-is-uml. *visual-paradigm.* [En línea] 10 de 10 de 2023. https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/.

ANEXOS

Anexo 1: Historia de Usuario RF23

		Historia de Usuario
Número:23	**Usuario:** Municipal, Provincial, Nacional	
Nombre de Historia: Listar Ficha de Nominado		
Prioridad en Negocio: Alta	**Riesgo en Desarrollo:** Bajo	
Tiempo estimado: 4 Horas	**Iteración Asignada:**1	
Programador Responsable: Roberto Valladares Martín		
El sistema debe permitirle al usuario visualizar el listado de las fichas de nominados de los diputados y delegados, mostrándose los campos: Circunscripción, Primer Nombre, Segundo nombre, Primer apellido, Segundo apellido, Carnet de identidad, Ficha confeccionada, Ficha entregada, Ficha tomada, Ficha impresa y Estado de aprobación. Se muestran además los campos: Municipio y Provincia para los niveles Provincia y Nación según corresponda.		

Prototipo elemental de interfaz gráfica de usuario:

Primer nombre	Segundo nombre	Primer apellido	Segundo apellido	No. de identidad	Ficha confeccionada	Ficha entregada	Foto tomada	Ficha impresa
Tomas		Amarán	Diaz	54011823564	Si	No	No	No
Lidia		Montes de Oca	Fernández	65081408739	Si	Si	Si	Si
Lidia		Montes de Oca	Fernández	65081408739	Si	Si	Si	Si
Tomas		Amarán	Diaz	54011823564	No	Si	Si	No
Lidia		Montes de Oca	Fernández	65081408739	Si	No	Si	No
Caridad		Alvarez	Rivero	61051902333	Si	No	Si	No
Minerva	de la Caridad	Valdés	Temprana	45103104851	Si	No	Si	No
Minerva	de la Caridad	Valdés	Temprana	45103104851	Si	No	Si	No

Mostrando 1 - 8 de 8 15 ‹ 1 ›

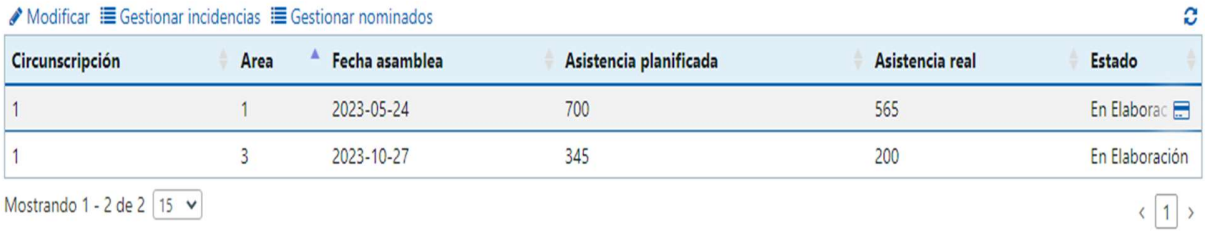
Fuente: Manual de Usuario

Anexo 2: Historia de Usuario RF32

		Historia de Usuario
Número:32	**Usuario:** Nacional	
Nombre de Historia: Listar informe de realización de asamblea.		
Prioridad en Negocio: Alta	**Riesgo en Desarrollo:** Bajo	
Tiempo estimado: 4 Horas	**Iteración Asignada:**1	
Programador Responsable: Roberto Valladares Martín		
El sistema debe permitirle al usuario visualizar el listado de los datos referentes al informe de realización de asamblea, mostrándose los campos: Asistencia planificada, Asistencia real y Estado de aprobación. Se muestran además los campos: Municipio y		

Provincia para los niveles Provincia y Nación según corresponda.

Prototipo elemental de interfaz gráfica de usuario:



Fuente: Manual de Usuario