VNIVERSIDAD NACIONAL AVTÓNOMA DE MEXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

" IMPLEMENTACIÓN DE UN SISTEMA DE VOTACIÓN ELECTRÓNICA COMO UN PRODUCTO SOBRE LA PLATAFORMA PLONE "

T E S I S

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN INGENIERÍA (COMPUTACIÓN)

P R E S E N T A:

ALEXANDER ZAPATA LENIS

DIRECTOR DE TESIS:"SERGIO RAJSBAUM GORODESKY"

México, D.F. **2008.**

AGRADECIMIENTOS

Gracias a mi madre por apoyarme siempre en mis decisiones, por sus oraciones y por todas las buenas energías que me envió a través de la distancia.

Agradezco al Dr. Sergio Rajsbaum por su conocimiento, paciencia, sus buenos consejos y todo el soporte que me brindó para poder lograr la terminación exitosa de este proyecto.

Muchas gracias al Dr. Boris Escalante, quien desde el día que conocí me brindó siempre su confianza y apoyo para ser un mejor profesional.

Gracias a mis sinodales Dr. Enrique Daltabuit, Dr. Juan Voutssás, Dr .José de Jesús Vázquez e Ing. Mario Rodríguez por todas sus recomendaciones y sugerencias para mejorar mi tesis y llegar a un producto final de alta calidad.

Muchas gracias al M. en C. Gustavo Adolfo Solís por sus excelentes consejos, todo su apoyo profesional y su alta calidad humana.

Gracias a mis compañeros, con quienes formé equipos de trabajo y quienes me ayudaron a superar algunas de las pruebas más duras en este camino, además de ofrecerme su amistad.

Muchas gracias a Elia Fernández de ISACA Internacional por su confianza, amistad y consejos que han sido muy importantes en mi desarrollo profesional.

TABLA DE CONTENIDO

CAPITULO 1

1. Introducción

En este capítulo se presenta una visión general del trabajo realizado, partiendo de los antecedentes que impulsaron el desarrollo del proyecto, se formula el planteamiento del problema, se presenta el objetivo general y los objetivos específicos, la contribución y relevancia del trabajo, y se describen los principales aspectos metodológicos requeridos para culminar con éxito el proyecto.

1.1. Antecedentes

En el Instituto de Matemáticas (IMATE) de la Universidad Nacional Autónoma de México se realizan regularmente procesos de votaciones, con el objetivo de elegir miembros de comisiones de funcionarios internos y de representantes ante comisiones externas al Instituto, como la que se especifica en los artículos 51 fracción IV, 51 A, 52-A, 52-B, 52-C y 52-D del Estatuto General de la UNAM; 3º, 4º, 5º y 7º del Reglamento Interno del Consejo Técnico de la Investigación Científica (RCTIC), y en el Reglamento Interno de este Instituto, en donde se dispone que cada 3 años se debe realizar la elección de un representante de su personal académico ante el consejo técnico de la investigación científica, mediante voto universal, libre y secreto [11]. Este proceso se ha realizado hasta ahora en forma manual mediante una urna física considerando que este proceso es relativamente simple y que los sistemas de votación existentes no están dirigidos a grupos, organizaciones o comunidades pequeñas en donde lo más importante es la facilidad para su utilización y la transparencia respecto a la confirmación del registro de los votos en el conteo final.

Por otra parte en el IMATE se ha realizado un esfuerzo muy importante por la automatización de algunos de sus procesos, mediante el Sistema de Información del Instituto de Matemáticas denominado InfoMatem [24], el cual permite compartir información del Instituto, actualizar información curricular de los investigadores, controlar y realizar búsquedas de la información de los mismos, manejar las actividades del Instituto (Seminarios, Coloquios y Eventos), así como administrar usuarios y procesos relacionados con el Consejo Interno, la Secretaría Académica y la Secretaría Técnica. Este sistema funciona sobre un ambiente Plone, el cual se ha convertido en la nueva generación de sistemas de gestión de contenidos de código abierto para la Web, soportando todas las tecnologías actuales de XHTML y CSS.

Las situaciones anteriores conducen a que se establezca como prioridad la

definición de los requerimientos de funcionalidad, flexibilidad y seguridad necesarios para implementar una solución de votación electrónica para el Instituto que cumpla con la normatividad electoral establecida por el mismo, que brinde el nivel de confiabilidad y transparencia requerido y que se encuentre en la plataforma abierta Plone, totalmente integrable con InfoMatem.

1.2. Objetivo General de la Tesis

Implementar un sistema de votaciones electrónicas de un solo nivel que permita la elección de un solo candidato, basándose en los requerimientos del micro universo del Instituto de Matemáticas de la Universidad Nacional Autónoma de México, que pueda ser extrapolado a otro tipo de organizaciones y micro universos más complejos, gracias a que cumple con las condiciones fundamentales de aceptación de un sistema de este tipo, que serán discutidas más adelante.

El sistema debe ser parametrizable respecto a los requisitos que deben cumplir los votantes y los candidatos y las funciones que deben cumplir los grupos de control, de acuerdo a reglas particulares de cada institución.

1.3. Objetivos Específicos

Para poder desarrollar este proyecto de Tesis se deben cumplir los siguientes objetivos específicos:

- *Determinación de los requisitos de seguridad que debe cumplir un sistema de votación electrónica.*
- *Conocimiento y evaluación del estado del arte de protocolos seguros de votaciones.*
- *Establecimiento de un protocolo de votación que pueda ser implementado en Plone y que cumpla con la mayor parte de los requerimientos del estado del arte de los protocolos seguros de votaciones.*
- *Conocimiento de la plataforma de Plone como Gestor de Contenido, Zope como Gestor de Base de datos y Python como ambiente de Programación.*
- *Implementación del sistema de votaciones electrónicas del IMATE, mediante un producto Plone, en dos fases:*
    - *Fase 1: Funcionalidad básica sin requerimientos de seguridad y sin*

conectividad con los demás sistemas del IMATE.

- *Fase 2: Sistema integrado a las demás aplicaciones del IMATE e implementación de su configuración básica de seguridad.*

1.4. Metodología

A continuación se describen las actividades realizadas para la implementación del proyecto, mencionando las técnicas y métodos de la Ingeniería de la Computación más relevantes en el proceso seguido:

- *Establecimiento de los requerimientos funcionales que a ser validados durante el proceso de implementación, con el fin de asegurar el nivel de calidad requerido por el IMATE, mediante la lectura y análisis de todos los documentos, actas y procedimientos relacionados con la elección del representante del personal académico del instituto de matemáticas ante el consejo técnico de la investigación científica y entrevistas con los funcionarios de los grupos controlan el proceso.*

- *Investigación sobre los métodos y técnicas de seguridad requeridas por un sistema de votaciones electrónicas, en relación con criptografía, protocolos de comunicación y métodos de autenticación*

- *Definición de una estrategia para la construcción del producto en la cual se incluyó la apropiada definición de estándares de programación y aseguramiento de la calidad del código, mediante el aprendizaje autodidacta de la plataforma Plone, Zope y Phyton.*

- *Análisis de protocolos de votación del estado del arte para la elección de las propiedades dentro del proceso de votación y escrutinio que debieron ser traducidas a los requerimientos técnicos a ser implementados en el Sistema.*

- *Definición del esquema para la implementación de las funciones criptográficas necesarias para dar cumplimiento a los requerimientos técnicos del paso anterior.*

- *Aplicación de las técnicas de administración de proyectos tecnológicos requeridas para dividir el proceso en fases y darle un enfoque efectivo y eficiente. Los resultados han sido supervisados mediante revisiones periódicas del avance en la implementación del producto con el asesor de Tesis y un experto en la construcción de sistemas de votación electrónica.*

1.5. Resultados

El resultado más relevante de esta tesis fue aportar una solución a un problema práctico del Instituto de Matemáticas de la Universidad Nacional Autónoma de México, que muchas instituciones y organismos privados y estatales más complejos tienen en el momento de querer llevar a cabo un proceso de votación, en el que los participantes tengan un alto nivel de confianza, a pesar de que se utilicen medios electrónicos.

Como parte de la solución se diseñó un protocolo de votación que cumple en gran medida con los principios básicos de las votaciones electrónicas, los cuales se especifican en el estado del arte, particularmente del Sistema KOA desarrollado por un grupo de investigadores de la Universidad de Dublín y el sistema de votaciones del Gobierno de Suiza. El protocolo diseñado tiene entre otras, las siguientes características:

1. *Lo primero que se define es el conjunto de usuarios que cumple con ciertas condiciones preestablecidas para poder ser elector del proceso de votaciones, dando un tiempo para que puedan realizarse los ajustes necesarios. Una vez que se tenga la lista final de electores, el organismo de control de la votación firma el padrón definitivo de electores y lo publica para que todos los usuarios lo puedan consultar (requerimiento clave de cualquier sistema de votación reconocido).*

2. *A continuación, se define el conjunto de usuarios que cumple con ciertas condiciones preestablecidas para poder ser candidato del proceso de votaciones, dando un tiempo para que puedan realizarse los ajustes necesarios, para que estos puedan aceptar, rechazar o cancelar después de haber aceptado sus candidaturas. Una vez que se tenga la lista final de candidatos, el organismo de control de la votación firma el padrón definitivo de candidatos y lo publica para que todos los usuarios lo puedan consultar (requerimiento clave de cualquier sistema de votación reconocido).*

3. *Los usuarios se autentican mediante contraseña, la cual es asignada en forma previa por un administrador después de un proceso de registro. En el caso particular del IMATE se utiliza el mismo acceso que ya tienen los usuarios al sistema InfoMatem (esta opción se seleccionó por facilidad en otros sistemas como el caso de KOA se genera un NIP específico para cada usuario y proceso de votación, el cual se entrega en forma segura).*

4. *Antes de la votación se genera una tabla de códigos aleatorios para cada uno de los candidatos definitivos que representan su*

"identificación", la cual es cifrada con la llave pública del organismo de control de la votación, es importante mencionar que esta tabla sólo puede ser obtenida en el momento de iniciar el conteo de votos, luego de terminar la votación (KOA – con la diferencia que se genera un número fijo de números aleatorios).

5. *Posteriormente, se asignan en forma aleatoria las "identificaciones" generadas de los candidatos a cada uno de los posibles electores, objeto al que se le denomina Boleta de votación. Es importante mencionar que para dos electores distintos, el mismo candidato tendrá identificaciones diferentes, lo que dificulta el registro de un voto por parte de un usuario externo que no sea elector, pues desconocerá las identificaciones válidas de cada candidato (KOA – con la diferencia de que la boleta se envía a los electores).*

6. *Cuando un elector quiere registrar su voto, el cual corresponde a la "identificación" del candidato para ese elector, se le agrega un número aleatorio generado en ese momento (KOA). De esta nueva cadena de números se obtiene el hash, y ese valor corresponde al número de recibo del voto (KOA y Suiza – se genera un recibo de voto, pero no se especifica la forma de obtenerlo), el cual le servirá al elector para verificar al final del escrutinio si su voto fue apropiadamente contado. Es importante mencionar, que las "identificaciones" de los candidatos para cada elector, sólo están disponibles en el momento el usuario que va a registrar su voto, pero no son visibles, además de que no pueden ser consultadas en forma previa o posterior (KOA – con la diferencia de que el elector si puede dejar soporte de su boleta de votación).*

   Nota: En la normatividad del proceso de votación del Instituto de Matemáticas no se considera la opción del voto en blanco, pero éste podría configurarse como un candidato adicional que se incluya en el padrón y cuya denominación sea "Voto en Blanco", sin ningún tipo de información adicional. Este manejo permitiría que el código del voto en blanco varíe para cada elector y no se registren en forma no autorizada votos en blanco en un ataque de tipo denegación de servicio selectiva.

7. *La cadena integrada por la "identificación" del candidato y el número aleatorio generado al momento del voto, es cifrada doblemente, primero con la llave pública del administrador de la votación y después por la llave pública del organismo que controla la votación, esto dificulta el proceso de criptoanálisis y exige que en la votación participe un grupo de control que le mayor confiabilidad al proceso (KOA y Suiza).*

8. *Las "identificaciones" de los demás candidatos, diferentes al seleccionado por el elector en su voto, se guardan en una tabla de*

códigos ya utilizados, y en el caso de que alguien llegue a tener acceso a la tabla de códigos de candidatos y se comprometa la llave privada del organismo de control, si utiliza uno de estos códigos válidos pero ya utilizado, su voto será inválido.

9. *Al elector se le coloca una marca que establece que ya votó por lo cual no tendrá disponible la opción para votar de nuevo (Todos los sistemas de votación).*

10. *Antes de realizar el conteo de votos la "urna" con los votos cifrados es revuelta en forma aleatoria mediante un proceso de reordenamiento, para que no haya forma de asociar el orden de los votos con los eventos de la bitácora de la elección, donde quedan registradas todas las actividades realizadas desde el momento en que se crea la votación hasta la publicación de los resultados (KOA y Suiza).*

11. *Para realizar el conteo de votos, lo primero que se debe hacer, es que el organismo de control descifre con su clave privada la tabla de códigos de candidatos, luego el administrador de las votaciones debe descifrar la urna de votos con su clave privada y finalmente el organismo de control debe realizar una segunda ronda de descifrado de la urna con su clave privada. Los votos especificados en el punto 5, los que no correspondan a códigos válidos de candidatos y los que se encuentren duplicados son anulados (Suiza).*

12. *El paso final del protocolo es la publicación de los resultados, la cual incluye la presentación del número total de votos por candidato y el detalle de los números de recibo contados para cada uno, los cuales les permitirán a los electores confirmar que su voto fue apropiadamente contado, lo que le da mucha transparencia y confiabilidad al proceso (KOA y Suiza).*

El protocolo anterior permite cumplir con los siguientes requerimientos que más adelante se explicarán en detalle, por ser claves para la implementación de un sistema de votaciones:

1) *Sólo los votantes autorizados son capaces de votar (los que cumplan los criterios preestablecidos).*

2) *Ningún votante podrá votar más de una vez (Debido a que el usuario que vota es marcado y no tendrá disponible de nuevo la opción para votar).*

3) *Los votos son almacenados correctamente (se guarda la "identificación" del candidato asignada a cada elector).*

4) *Los votos no pueden ser modificados o borrados sin detección (con la doble ronda de cifrado se asegura que no se realicen modificaciones inválidas a*

los votos, en el caso que se realicen el voto será anulado). Estaría pendiente la verificación del borrado de votos.

5) *Es posible verificar que todos los votos fueron correctamente contados en la elección final (en el conteo de votos se realizan comparaciones con los eventos de voto en la bitácora de las votaciones, se verifican y reportan votos inválidos y duplicados)*

6) *Debe trabajar en forma robusta, sin pérdida de votos aunque se presenten numerosas fallas en el sistema o haya pérdida completa de comunicación (este requerimiento no se cumple y corresponderá a un desarrollo futuro relacionado con el fortalecimiento del sistema ante ataques de denegación de servicio).*

7) *Los votantes tendrán un mayor nivel de confianza en el sistema debido a que éste permitirá al terminar una votación, que verifiquen efectivamente que su voto fue considerado en forma apropiada (con la publicación de los números de recibo de voto los electores pueden verificar que su voto fue adecuadamente contado).*

8) *Los votantes no requieren muchas habilidades para poder votar y el acceso al sitio de votación es fácil (los electores y candidatos, particularmente del IMATE ya utilizan el sistema InfoMatem sobre la plataforma Plone, por lo que utilizan su mismo usuario y contraseña y además están familiarizados con la interface del sistema).*

El reto más importante fue integrar la funcionalidad de un proceso de votaciones en una plataforma de software abierto que permitiera ser objeto de un mejoramiento continuo por parte de una importante comunidad internacional de usuarios, gozando a su vez de la propiedad de total transparencia, sin que se sacrificaran las aspectos de seguridad, este mismo ideal ha sido perseguido por Joseph Kiniry y su equipo de trabajo de la Universidad de Dublín en su sistema KOA, descrito más adelante en el estado del arte.

Por otra parte se tomó la decisión de no construir funciones criptográficas propias del producto de Votaciones para la generación de claves privadas y públicas, encripción de votos, desencripción de votos, firma de documentos y verificación de firma de documentos, sino utilizar GnuPG, la cual tiene entre otras ventajas las siguientes:

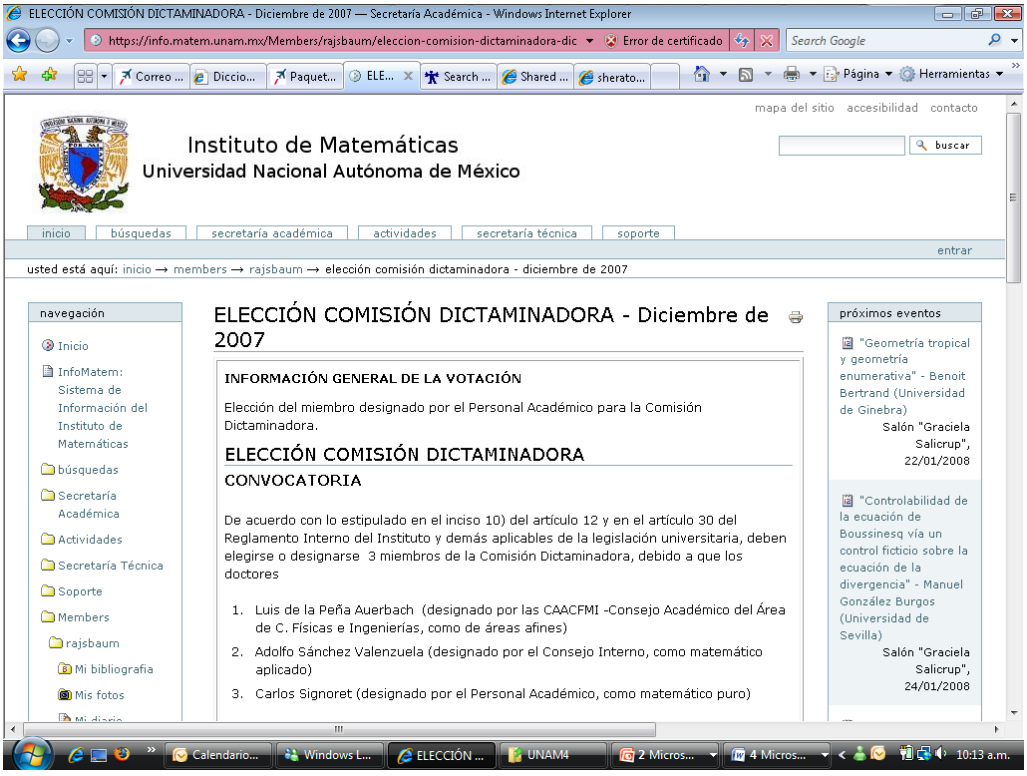
- *Es software libre, susceptible de mejoramiento continuo.*
- *Es software desarrollado en Europa, por lo que se evitan restricciones de los EEUU a la exportación de software criptográfico.*
- *Es compatible con el estándar OpenPGP y por lo tanto está demostrado que bien utilizado ofrece un gran nivel de seguridad.*

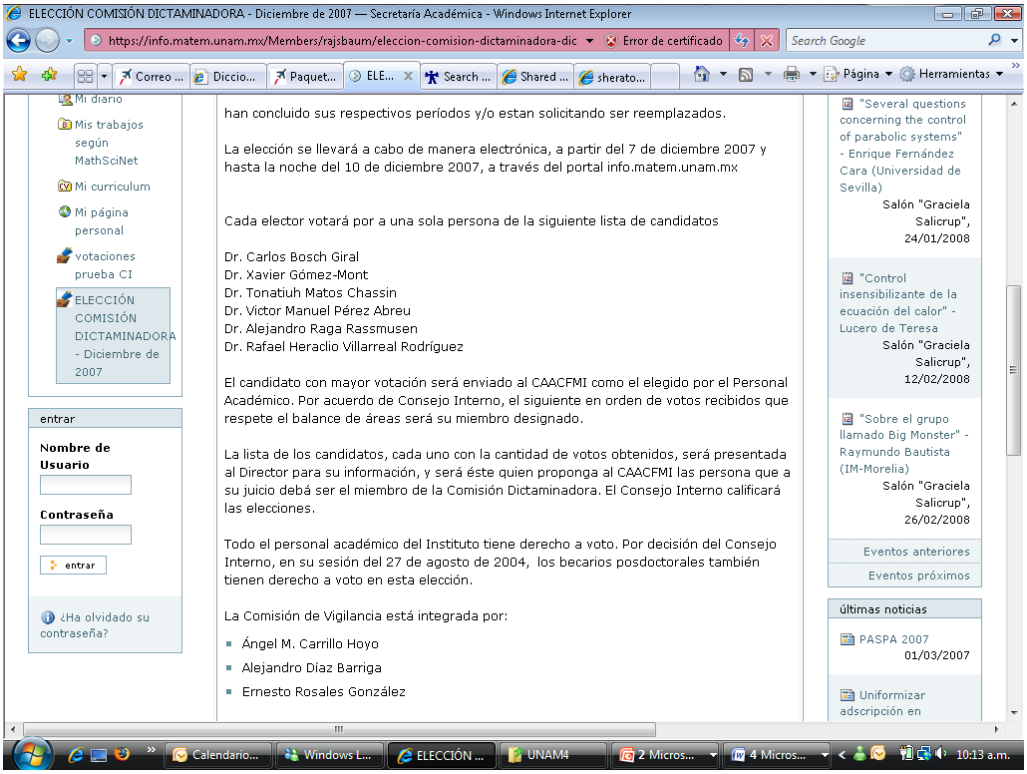
- *Está disponible en multitud de sistemas.*

- *Se puede usar con fines personales y comerciales*

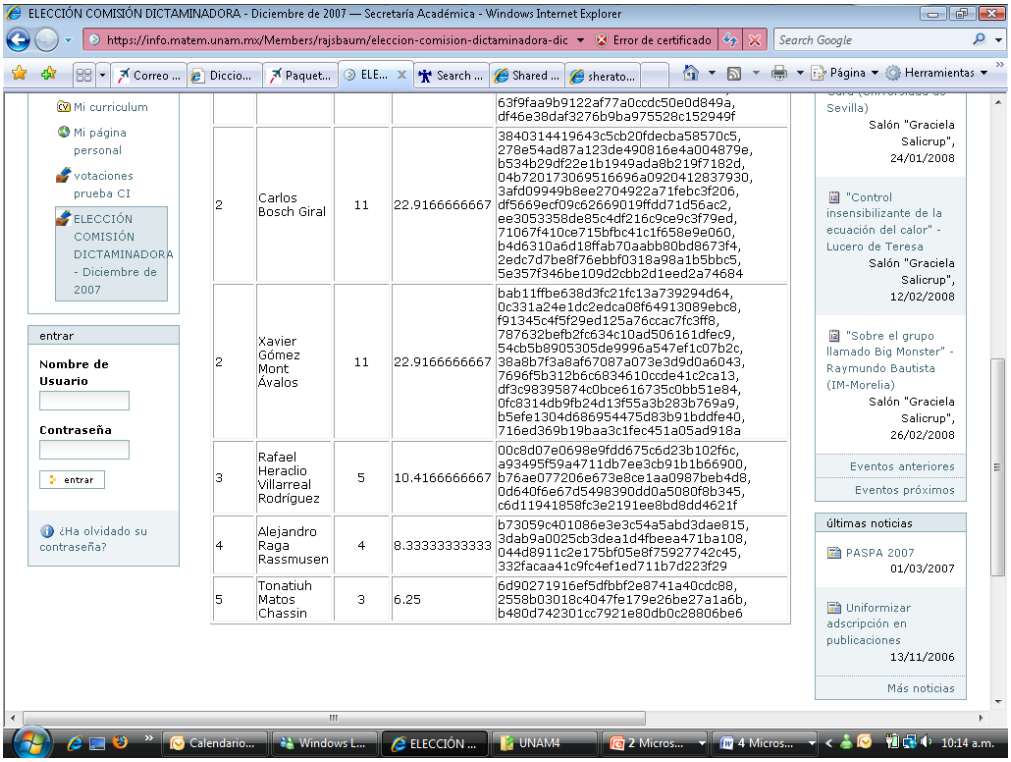
Es importante mencionar que como parte del resultado de esta tesis se entrega la documentación de ayuda necesaria para garantizar un uso efectivo del sistema por parte de los usuarios que van a administrarlo y controlarlo, como por parte de los que tienen un rol de participantes en el proceso, como es el caso de los electores y candidatos.

Finalmente, es muy importante mencionar que el software que automatiza la solución aportada en esta tesis fue utilizado exitosamente por los investigadores del Instituto de Matemáticas en el proceso de elección de la Comisión Dictaminadora entre el 7 y el 10 de diciembre de 2007, tal como se muestra en las pantallas siguientes:





Los resultados del proceso de votación, sobre los cuales no hubo ningún tipo de reclamación y en los cuales los votantes tuvieron la oportunidad de verificar que su voto fue considerado apropiadamente en el conteo final, fueron los siguientes:



1.6. Supuestos

Como se explicó en los antecedentes, la decisión de implementar el sistema de votaciones electrónicas sobre Plone se basó, entre otras condiciones, en la existencia de soluciones del Instituto de Matemáticas que ya operaban en esta plataforma abierta y la familiaridad de los usuarios en el manejo de la misma, por lo que el proyecto se orientó a la implementación de las funcionalidades claves requeridas, partiendo del cumplimiento de los siguientes supuestos, los cuales son relevantes para que el sistema de votación sea operable y confiable, respecto a su seguridad:

1. *El servidor debe contar con un certificado digital válido que le permita a la comisión de vigilancia, los electores y los candidatos verificar la autenticidad del sitio web de las votaciones al cual se están conectando y evitar ataques de suplantación de IP (IP Spoofing)*

2. *La comunicación de los usuarios con el servidor debe basarse en un protocolo tipo SSL/TLS que permita que la información viaje en forma cifrada, incluyendo el usuario/contraseña con el que se autentican.*

3. *No se incluyen programas de tipo malicioso por parte de los desarrolladores.*

CAPITULO 2

2. Estado del arte en sistemas de votación electrónica.

Después de revisar una amplia bibliografía en Internet respecto a las características de los sistemas de votación electrónica disponibles, no es difícil entender por qué no se cuenta aún con un nivel de confianza considerable por parte de la sociedad en general, respecto a la funcionalidad y seguridad de los mismos, es importante aclarar que en muchos de los casos no se debe a la debilidad en sí de los algoritmos o protocolos sobre los que se fundamentan sino más bien en los controles con los que cuentan para evitar manejos indebidos por parte de los encargados de su administración o personas externas al sistema. No obstante, los sistemas de votación electrónica si están considerándose como una alternativa factible a las votaciones tradicionales [20], si se logra cumplir a un costo razonable con requerimientos como, exactitud, privacidad, verificabilidad, simplicidad, flexibilidad, entre otras que se describirán más adelante.

Antes de revisar las características específicas de esquemas de votación electrónica es importante considerar una posible clasificación general para los mismos [38]:

1. *Urna Electrónica - Voto presencial:*

   Son los más difundidos y existe una gran diversidad, combinan procedimientos tradicionales como el uso de boletas y lectores ópticos, urnas touch screen o teclados numéricos (no es necesario el uso de boletas), emiten un certificado ya sea parcial o total de los resultados y algunos sistemas incluso prevén la identificación electrónica del elector

2. *E-vote – voto a distancia:*

   Tienen mayor grado de complejidad, dependiendo de la estrategia de implementación (lo más básico sería enviar el voto por email), pueden ser de fácil acceso para los ciudadanos, incluyen portales en la red diseñados específicamente para la votación y requieren de software y sistemas de encripción especiales.

Debido a que uno de los requerimientos del Instituto de Matemáticas es que los electores puedan votar en forma remota (e-vote), a continuación se presenta una comparación [21] de diferentes esquemas de votación electrónica de este tipo que

han sido propuestos entre 1983 y el 2004, y se describen en forma general dos esquemas, uno utilizado por el sistema de elecciones en Suiza y otro en el proyecto KOA desarrollado por un equipo de trabajo de la Universidad de Dublín, de los cuales como se podrá comprender más adelante, se utilizaron características claves a cumplir por parte del sistema implementado en esta tesis. En forma adicional, se incluye la descripción del Sistema de Votaciones Electrónicas desarrollado por DGSCA, el cual no fue considerado como base para el diseño del sistema propuesto para el Instituto de Matemáticas.

2.1. Comparación de esquemas de votación electrónica.

Las sociedades democráticas y algunas instituciones organizacionales están fundadas en el principio de las elecciones y las capacidades de expresar su opinión.

Particularmente, los sistemas de votación electrónica han buscado ser un soporte para cumplir con el principio anterior y se han convertido en una aplicación social de diversos protocolos criptográficos. Estos sistemas pueden llegar a ser comparables si se especifica un adecuado conjunto de requerimientos [21], los cuales generalmente están relacionados con aspectos de seguridad, como entre otras, las que se especifican a continuación:

- *Eligibilidad (Eleg): se refiere a que sólo los votantes válidos que cumplen ciertas condiciones predefinidas pueden votar y el sistema valida estas condiciones antes de aceptar un voto.*

- *Privacidad (Priv): intervienen dos factores. En primer lugar el esquema debe ser anónimo, de modo que no resulte posible averiguar la relación de un voto con el votante que lo realizó. Finalmente, el esquema ha de evitar que ningún votante pueda demostrar cuál fue su voto, con el objeto de eliminar la posibilidad de extorsión o de compra de votos. La máxima privacidad se logra cuando la privacidad de un votante sólo se puede romper si todas las demás entidades del proceso se ponen de acuerdo para hacerlo.*

- *Verificabilidad (Verf): Un esquema de votación es verificable individualmente (ind) si los votantes pueden comprobar que sus correspondientes votos han sido realmente incluidos en el recuento final. Una variante más rigurosa de este requisito es la verificabilidad universal (máxima), en el sentido de que cada votante (y, en general, cualquier persona) pueda verificar la integridad de todo el conjunto de votos.*

- *Exactitud (Exac): un esquema de votación es exacto si no resulta posible*

alterar o eliminar un voto que ha sido validado, y tampoco resulta posible incluir en el recuento un voto no validado. Pueden haber ciertas condiciones que lo vuelvan un requerimiento condicional (con).

- *Justicia (Just): para lograr una elección imparcial se requiere que no estén disponibles para nadie resultados intermedios antes de que terminen las elecciones. Pueden haber ciertas condiciones que lo vuelvan un requerimiento condicional (con).*

Algunos de estos requerimientos son contradictorios, por ejemplo la de privacidad del votante, que exige que no se pueda ligar al votante con el voto, con la de verificabilidad individual, pues para que el votante pueda confirmar que su voto fue bien contado, debe contar con un recibo que describa la forma en que votó.

El resumen del cumplimiento de los requerimientos anteriores en diferentes esquemas de votación electrónica, desarrollados entre 1981 y el 2004, se presenta en la tabla siguiente [21]:

Esquema/Requerimiento	Eleg	Priv	Verf	Exac	Just
Boyd, 1990 [25]	ok	Max	ind	no	no
Sako and Killian, 1995 [26]	ok	Ok	ok	ok	con
Chaum, 2004 [27]	ok	Ok	max	con	no
Iverson, 1992 [28]	ok	Ok	ind	con	con
Schoenmakers, 1999 [29]	ok	Ok	ok	ok	con
Lee and Kim, 2002 [30]	ok	Ok	ok	ok	con
Kiayias and Yung, 2002 [31]	ok	Max	ok	ok	con
Fujioka et al, 1993 [32]	ok	ok	ind	no	ok
Okamoto, 1997 [33]	ok	ok	ind	no	con
Lee et al, 2003 [34]	ok	ok	ok	ok	con
Kiayias and Yung, 2004 [35]	ok	ok	ok	ok	con
Juels and Jakobsson, 2002 [36]	ok	ok	ind	con	con

***no** =no se cumple, **ok** = se cumple, **max**=máximo, **ind**=individual, **con**=condicional*

Para comprender mejor los requerimientos anteriores se explicarán a continuación algunos de los componentes criptográficos utilizados para la construcción de diferentes esquemas de votación electrónica [21]:

1) *Canales Seguros: se requiere el establecimiento de canales privados usando criptosistemas de clave pública y privada, los cuales son seguros pero observables o asegurados físicamente [26].*

2) *Canales Anónimos: se requieren para garantizar el requerimiento de privacidad y ocultar la identidad del votante a la autoridad o el administrador del proceso. Han habido varios tipos implementaciones que utilizan criptografía y permutaciones denominadas Mixnet, pero éstas no son robustas por el riesgo de mezclas fallidas o corruptas.*

3) *Tablero de anuncios: el cual puede ser accedido públicamente. Puede ser definido como un canal de comunicación tipo broadcast con memoria, que puede ser utilizado también por la autoridad de vigilancia de la votación.*

4) *Firma Ciega: es un protocolo criptográfico que se usa para hacer anónimo al voto. El votante oculta su voto usando una cadena aleatoria y la clave pública de la autoridad, todo esto lo firma con su clave privada, luego la autoridad verifica la validez con la clave pública del votante, y firma con su clave privada, finalmente el votante verifica la firma de la autoridad y quita la cadena aleatoria obteniendo el voto firmado por la autoridad, el cual corresponde al voto con la firma ciega [27]. Al combinarlo con un canal de broadcast anónimo se logra la privacidad máxima.*

5) *Encripción Homomórfica: un algoritmo de encripción es homomórfico si al aplicarlo a una cadena 1 y a una cadena 2 se puede obtener su aplicación a 1 * 2, sin descifrar 1 y 2, para alguna operación *.*

6) *Secreto compartido: se debe distribuir el secreto entre varias autoridades, por lo que éstas deben trabajar en conjunto para poder utilizar la clave privada.*

2.2. Sistema de elecciones en Suiza

Suiza es el país en el mundo con el mayor número de ejercicios democráticos, elecciones, referendos o de otro tipo, sus ciudadanos van a las urnas 4 o 6 veces al año.

Esquema/Requerimiento	*Eleg*	*Priv*	*Verf*	*Exac*	*Just*

Sist Votación Suiza	*ok*	*ok*	*Ind*	*ok*	*ok*

no *=no se cumple,* ***ok*** *= se cumple,* ***max****=máximo,* ***ind****=individual,* ***con****=condicional*

Entre las características claves del sistema de votaciones electrónicas tipo e-vote que usan, como medio complementario al envío de votos por correo y las votaciones manuales, se considera importante mencionar las siguientes [16]:

1) *Los votos electrónicos se cifran con dos claves diferentes.*

2) *La contraseña que desbloquea estas claves sólo es conocida por la comisión que vigila la elección.*

3) *Los votos electrónicos se guardan aparte de la base de datos de votantes y no hay manera de asociar un voto con un votante.*

4) *Los votos son cifrados y para evitar la posibilidad de aplicar técnicas de criptoanálisis sobre texto conocido se completan con un texto arbitrario antes de ser cifrados.*

5) *Para permitirle al ciudadano asegurarse que no es víctima de un IP Spoofing y que él no está votando en un sitio sustituto usan un "código de retorno". La confirmación que se le envía al votante está embebida en una imagen que es diferente para cada votante. Esta imagen puede ser reproducida en la tarjeta de votación y el ciudadano puede verificar que las dos imágenes hagan match.*

6) *El archivo de votos electrónicos es ordenado en forma aleatoria antes de ser desencriptado, lo que ofrece una protección contra el mal uso de los archivos de bitácoras, pues sino sería posible reconstruir el orden de registro de las votos y violar así el anonimato del voto, comparando este orden con la fecha y hora de cada voto.*

7) *El servidor de votaciones es autenticado mediante un certificado digital, debido a que su "huella digital" está impresa en la tarjeta de votación, el votante puede verificar que en realidad está conectado al servidor correcto.*

8) *Para proteger al sistema contra ataques de denegación de servicio se implementaron dispositivos especiales de control, estos dispositivos detectan llamado de páginas desde la misma dirección, intentos sistemáticos de identificación, llamados repetidos, así como fallas de software o hardware en uno de los servidores y la corrupción de cualquier archivo relacionado del sistema.*

9) *Cualquier diferencia entre el número de votos electrónicos guardados en*

la urna y el número de votantes que votaron por Internet, según las bitácoras indicaría una falla en el proceso.

10) *Se usan dos servidores en espejo que corren en una base de datos única, la cual se encuentra en un sistema de disco seguro (dual disk).*

11) *Los votantes reciben una confirmación del registro de su voto.*

12) *La comisión de elecciones posee las dos claves de encripción que se usan para cifrar cada uno de los votos. Sin estas dos claves es imposible descifrar la urna para leer y contar los votos.*

2.3. KOA (Kiezen op Afstand)

Es una nueva plataforma de software tipo e-vote sobre código abierto para la investigación sobre procesos de votación, basados en computador. Está basada en una plataforma de software liberada para el Gobierno Holandés, bajo licencia GPLv2 en el 2004 [22].

Esquema/Requerimiento	*Eleg*	*Priv*	*Verf*	*Exac*	*Just*
KOA	*ok*	*ok*	*ind*	*ok*	*ok*

***no** =no se cumple, **ok** = se cumple, **max**=máximo, **ind**=individual, **con**=condicional*

El proceso se está implementando es el siguiente [23]:

1) *Los votantes se registran previamente en una oficina del gobierno para poder obtener su código PIN.*

2) *El votante inicia una conexión HTTPS con el sitio web de las elecciones.*

3) *El sitio web presenta su certificado el cual es confirmado por el votante.*

4) *El sitio web le ofrece al usuario una página para que realice su respectivo Login.*

5) *El votante se autentica con su identificación y su código PIN, el cual es confirmado por el servidor.*

6) *Una boleta única es enviada por email al votante, dicha boleta sólo puede ser usada por ese votante.*

7) *La boleta tiene un número al lado de cada candidato, el cual es*

diferente para cada votante, lo que se puede considerar como una encripción previa.

8) *Cuando un votante registra su voto, ese número único es encriptado, trasmitido y almacenado en el servidor.*

9) *Aunque ese número único sea interceptado durante su trasmisión a la base de datos, éste carece de significado pues es diferente para cada votante.*

10) *Después de votar, el usuario obtiene un número de recibo que le servirá para verificar que su voto fue contado.*

Entre las características claves del sistema de votaciones electrónicas, se considera importante mencionar las siguientes:

1) *Para asegurar la integridad de la información se usa un número de códigos mayor al número de votantes, los cuales son generados para cada candidato y sólo uno es asignado aleatoriamente a cada votante, por lo que aunque un código malicioso pueda acceder a la boleta, el atacante no podrá modificarlo por otro código que sea válido para otro candidato.*

2) *Los votos son doblemente cifrados y la única forma de descifrarlos es utilizando las contraseñas que permiten acceder a las claves privadas de los miembros de la comisión de vigilancia del proceso, por lo que no se podrán realizar conteos parciales a menos que todos los miembros de la comisión de vigilancia se pongan de acuerdo.*

3) *El sistema cuenta con mecanismos para asegurar que un usuario sólo podrá registrar un voto.*

4) *Los usuarios pueden verificar que su voto fue incluido correctamente en el conteo final mediante un número de recibo que se entrega al votar.*

5) *Se le agrega información aleatoria a los votos antes de que estos sean cifrados, esto asegura que los votos de los votantes del mismo distrito y para el mismo candidato tienen un código encriptado distinto para cada voto, para prevenir el criptoanálisis por texto conocido.*

6) *Los votos son descifrados y contados en un orden aleatorio para evitar la trazabilidad con los votantes mediante las bitácoras del sistema.*

7) *El sistema tiene la capacidad de tomar una instantánea de los candidatos y la lista de votantes denominada "Huella dactilar electrónica" en cualquier momento para verificar que no han sido maliciosamente alterados.*

2.4.SVE - Sistema de Votaciones Electrónicas de la UNAM

Así como los dos sistemas anteriores, el Sistema de votaciones electrónicas de la UNAM se puede tipificar en la segunda clasificación de los sistemas de votación electrónica (e-vote).

El proceso implementando en SVE es el siguiente [39]:

Previo a la votación:

1) *Entrega de la lista final y definitiva de electores, y la generación y entrega de NIPs a cada uno.*

Durante la votación:

2) *Apertura de la votación*
3) *Validación y reconocimiento del derecho del votante.*
4) *Verificación de la validez del voto y dislocación de toda relación: elector – voto.*
5) *Depósito del voto aleatoriamente en la "urna"*
6) *Procesos de monitoreo del proceso, la afluencia de electores, la infraestructura física, entre otros aspectos.*

Entrega de Resultados:

7) *Cierre de la votación*
8) *Conteo de votos.*
9) *Emitir reportes con los resultados de la jornada electoral.*
10) *Integrar paquete electoral.*
11) *Calificar elección y presentar resultados oficiales.*

Entre las características claves del SVE, se considera importante mencionar las siguientes, algunas de las cuales han sido incluidas como posibles mejoramientos del sistema en trabajos futuros:

- *El SVE observa tanto en su código como en sus bases de datos, los estándares internacionales de seguridad informática más actuales y opera con una arquitectura de alta disponibilidad y diseño basada en la lógica de uso con protección contra ataques informáticos.*

- *La seguridad está estructurada por capas, las cuales permiten controlar cada una de las partes del Sistema, bajo un diseño que garantiza los principios básicos inherentes al proceso de votaciones.*

- *Su arquitectura integra mecanismos de protección de datos, monitoreo y detección de actividades de intrusos al Sistema.*

- *Aplicación de certificado digital del sitio web, mediante cifrado SSL (secure Socket Layer)*

- *Hosting del sitio, para garantizar un ininterrumpido proceso lectoral y un nivel de seguridad máximo.*

- *Seguridad vía web, utiliza Internet como medio de comunicación garantizando disponibilidad a la comunidad universitaria aplicando firewalls, monitoreo permanente y bitácora de accesos para evitar intrusiones.*

- *Sistema de detección de intrusos que informa analizando las reglas del comportamiento informático una posibilidad de ataque o sabotaje a la infraestructura de cómputo o al proceso como tal.*

- *Encriptación de la información antes de ser almacenada para proteger la confidencialidad e integración de los datos.*

- *Autenticidad (autenticación) del elector con el objeto de garantizar únicamente los accesos autorizados.*

- *Integridad del Sistema, como parte de su arquitectura, puede ser auditado.*

- *Replicación de datos y mecanismos de redundancia y procedimientos.*

2.5. Conclusiones.

Los sistemas de votaciones KOA y del Gobierno de Suiza, tipificados en la segunda clasificación de los sistemas de votación electrónica (e-vote), se utilizaron como base para la definición del protocolo de votaciones, debido a la información detallada sobre el funcionamiento de los mismos con la que se contó, la utilización de funciones criptográficas relativamente simples que no requieren una alta capacidad de cómputo, el cumplimiento con la mayor parte de los requerimientos de los sistemas de votación analizados en el estado del arte, además de que se tuvo contacto personal con el líder del proyecto KOA en la Universidad de Dublín.

Los requerimientos que se tomaron de estos dos sistemas para el diseño del

protocolo de votación implementado fueron las siguientes (se conserva la numeración de las características usadas en los puntos 2.2 y 2.3):

Sistema de votaciones de Suiza:

1) *Los votos electrónicos se cifran con dos claves diferentes (se aplica tal cual, pues así se exige la participación de al menos un organismo independiente al proceso que asegure que los votos no son alterados después de sacarlos de la "urna").*

2) *La contraseña que desbloquea estas claves sólo es conocida por la comisión que vigila la elección (en realidad se tomó parcialmente porque una de las claves se protege con una contraseña que sólo conoce la comisión y otra con una contraseña que conoce el administrador de la votación).*

4) *Los votos son cifrados y para evitar la posibilidad de aplicar técnicas de criptoanálisis sobre texto conocido se completan con un texto arbitrario antes de ser cifrados (se aplica tal cual, para evitar el criptoanálisis mencionado).*

6) *El archivo de votos electrónicos es ordenado en forma aleatoria antes de ser desencriptado, lo que ofrece una protección contra el mal uso de los archivos de bitácoras, pues sino sería posible reconstruir el orden de registro de las votos y violar así el anonimato del voto, comparando este orden con la fecha y hora de cada voto (se aplica tal cual para evitar asociar el voto al elector).*

9) *Cualquier diferencia entre el número de votos electrónicos guardados en la urna y el número de votantes que votaron por Internet, según las bitácoras indicaría una falla en el proceso (se aplica tal cual, como parte de los resultados publicados del escrutinio se especifica la ocurrencia de estas situaciones).*

11) *Los votantes reciben una confirmación del registro de su voto (se aplica tal cual para que los electores verifiquen que su voto fue apropiadamente contado).*

12) *La comisión de elecciones posee las dos claves de encripción que se usan para cifrar cada uno de los votos. Sin estas dos claves es imposible descifrar la urna para leer y contar los votos (se aplica parcialmente en pues si se requieren dos claves para descifrar la urna y leer los votos, pero una la sabe la comisión y otra el administrador de la votación).*

KOA:

Respecto al proceso se está implementando que:

1) *Los votantes se registran previamente en una oficina del gobierno para poder obtener su código PIN (se aplica parcialmente pues en realidad se requiere un PIN específico para la votación, en lugar de eso se aprovecha la contraseña de autenticación de los usuarios al sitio Plone, la cual requirió un proceso de registro para poder haber sido configurada).*

4) *El sitio web le ofrezca al usuario una página para que realice su respectivo Login (tal cual, se utiliza la página inicial de entrada al sistema Plone que en el caso del IMATE corresponde a la autenticación en el sistema InfoMatem).*

5) *El votante se autentica con su identificación y su código PIN, el cual es confirmado por el servidor (tal cual, pero como se menciona en el punto anterior esta verificación la hace el servidor Plone no el producto de votaciones).*

6) *Una boleta única es enviada por email al votante, dicha boleta sólo puede ser usada por ese votante (se aplica parcialmente, en realidad si se genera una boleta de votación única por cada elector, pero ésta no es enviada por email, se crea como un objeto oculto que sólo se utilizará en el momento de votar y sólo puede ser accedida por el elector correspondiente).*

7) *La boleta tiene un número al lado de cada candidato, el cual es diferente para cada votante, lo que se puede considerar como una encripción previa (se aplica tal cual, y es un aspecto clave de la seguridad del protocolo de votación).*

8) *Cuando un votante registra su voto, ese número único es encriptado, trasmitido y almacenado en el servidor (se aplica tal cual).*

9) *Aunque ese número único sea interceptado durante su trasmisión a la base de datos, éste carece de significado pues es diferente para cada votante (se aplica tal cual, por eso en el escrutinio este tipo de valores corresponderían a votos anulados).*

10) *Después de votar, el usuario obtiene un número de recibo que le servirá para verificar que su voto fue contado (se aplica tal cual, y es un aspecto de confianza de los usuarios hacia el sistema).*

Respecto a las características claves del sistema de votaciones electrónicas:

1) *Para asegurar la integridad de la información se usan 1000 códigos que son generados para cada candidato y sólo uno es asignado*

aleatoriamente a cada votante, por lo que aunque un código malicioso pueda acceder a la boleta, el atacante no podrá modificarlo por otro código válido para otro candidato (se aplica tal cual, sólo con la diferencia de que no son 1000 códigos, sino que se generan tantos como el número de electores del padrón definitivo, con el fin de limitar más la posibilidad de que un atacante ingrese un voto con un código que corresponda a los números aleatorios generados).

2) *Los votos son doblemente cifrados y la única forma de descifrarlos es utilizando las contraseñas que permiten acceder a las claves privadas de los miembros de la comisión de vigilancia del proceso, por lo que no se podrán realizar conteos parciales a menos que todos los miembros de la comisión de vigilancia se pongan de acuerdo (se aplica parcialmente, pues a pesar de que si son doblemente cifrados, uno de los procesos se realiza con la clave pública de la comisión y otro con la del administrador de la votación).*

3) *El sistema cuenta con mecanismos para asegurar que un usuario sólo podrá registrar un voto (se aplica tal cual y es un aspecto clave de cualquier sistema de votación reconocido).*

4) *Los usuarios pueden verificar que su voto fue incluido correctamente en el conteo final mediante un número de recibo que se entrega al votar (se aplica tal cual y como se mencionó antes es clave para darle confiabilidad al sistema).*

5) *Se le agrega información aleatoria a los votos antes de que estos sean cifrados, esto asegura que los votos de los votantes del mismo distrito y para el mismo candidato tienen un código cifrado distinto para cada voto, para prevenir el criptoanálisis por texto conocido (se aplica tal cual con el fin de prevenir el criptoanálisis mencionado).*

6) *Los votos son descifrados y contados en un orden aleatorio para evitar la trazabilidad con los votantes mediante las bitácoras del sistema (se aplica tal cual y básicamente antes del escrutinio se realiza un reordenamiento aleatorio de la urna con los votos).*

CAPITULO 3

3. Descripción Sistema Actual.

En este capítulo se describen las características particulares del proceso de votación que se lleva actualmente en el IMATE con el fin de mostrar algunos aspectos que caracterizan un proceso de votación tradicional, después se hará una descripción general de los requerimientos básicos de los sistemas de votaciones electrónicas existentes en el mercado, finalmente se analizará el cumplimiento de estos requerimientos en el sistema actual, con el fin de determinar los riesgos existentes en un sistema tradicional de votaciones como el del IMATE.

3.1. Sistema Actual

Los participantes en el proceso de votaciones del IMATE son los siguientes:

Cada uno de los participantes tiene las siguientes funciones o características:

- *Consejo Interno: grupo de control del proceso que se encarga de designar la comisión de vigilancia y escrutadores, seleccionar votantes y definir número y ubicación de casillas.*

- *Unidades Foráneas: corresponden a las sedes de Morelia y Cuernavaca donde se encuentran los investigadores que no son de la sede principal Distrito Federal.*

- *Investigadores: pueden ser votantes o candidatos o ambos, dependiendo del cumplimiento de ciertos requisitos.*

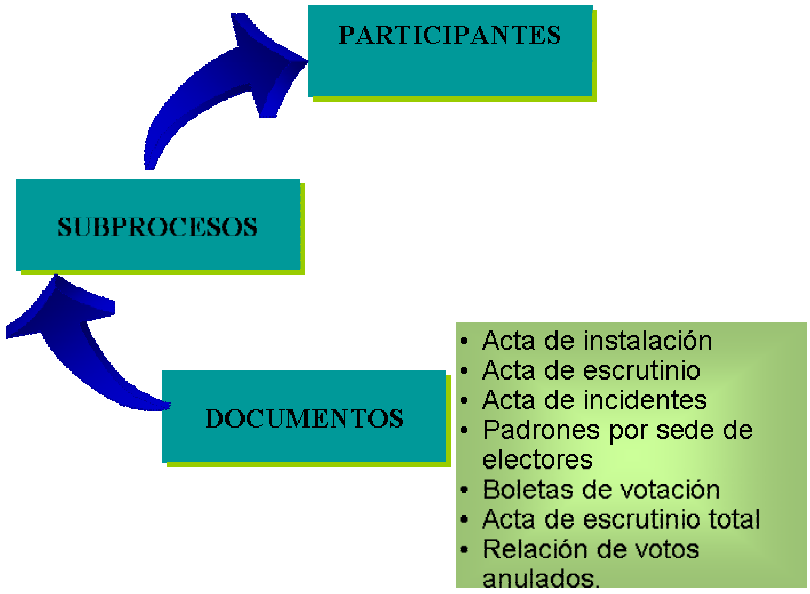
- *Comisión de Vigilancia: grupo de control del proceso que se encarga de seleccionar candidatos, vigilar la elección, cuidar las urnas y recibir votos previos por email.*

- *Secretaría Académica: grupo de control del proceso que se encarga de recibir votos previos al proceso y confirmar candidatos.*

- *Escrutador o Delegado en sedes: grupo de control del proceso que se encarga de vigilar urnas y escrutinio y recibir votos previos de unidades foráneas.*

- *Consejo Técnico de la Investigación Científica: se encarga de declarar el resultado del proceso de votaciones.*

Los documentos y formatos que se manejan en el proceso actual en el IMATE son los siguientes:



Cada uno de los cuales tiene las siguientes propiedades:

- *Acta de Instalación: certificación de que los delegados verificaron la urna vacía, el padrón de electores y el número boletas.*

- *Acta de Escrutinio: registro de cancelación de boletas no utilizadas, conteo de votos en presencia de la comisión de vigilancia, y acta de cierre de la casilla.*

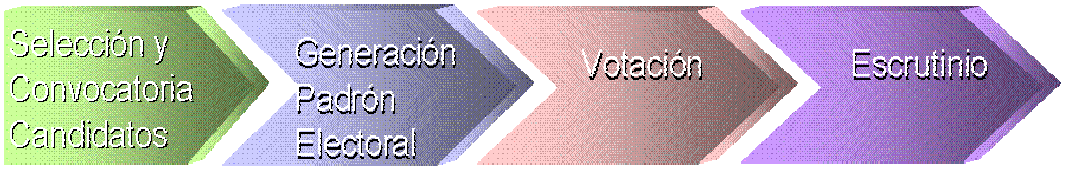
- *Acta de Incidentes: relación de situaciones no previstas en el proceso resueltas por la comisión vigilancia.*

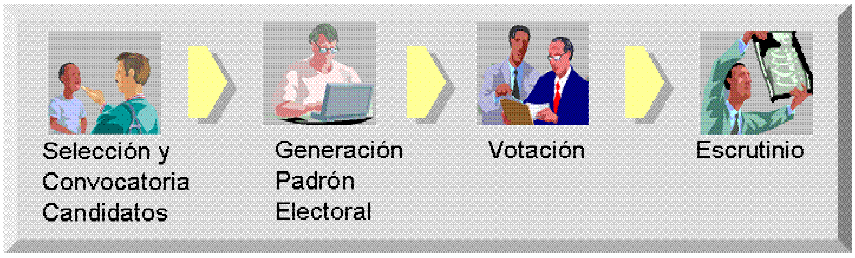
- *Padrón Electoral: relación de posibles votantes por sede, debe ser*

firmado por el votante después de su voto.

- *Boletas de votación: diseñadas por el consejo interno con los nombres de los candidatos para que el votante realice su voto.*
- *Acta de Escrutinio Total: certificación del conteo final de votación y nombre del elegido. Se envía a los académicos y unidades foráneas por email y se informa al Consejo interno.*
- *Relación de votos anulados: votos no válidos por ser depositados en blanco.*

El proceso de votación está compuesto por los siguientes subprocesos:





A continuación se describen en forma general cada uno de los subprocesos:

1. Selección y Convocatoria de Candidatos

Este subproceso tiene como objetivo establecer los candidatos que van a ser elegibles en el proceso de votación.

Según el reglamento del proceso, los investigadores que cumplan con los siguientes requisitos pueden ser candidatos potenciales:

- *Ser investigador definitivo en el instituto*
- *Haber cumplido con su programa de trabajo*

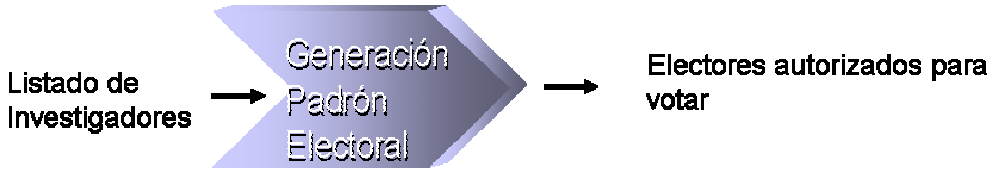
- *No ocupar ni percibir remuneraciones por plaza o asignación en el desempeño de un cargo de carácter académico-administrativo o administrativo.*
- *No pertenecer a alguna comisión dictaminadora*
- *No haber sido sancionado por incurrir en una causa grave de responsabilidad.*

Finalmente, los posibles candidatos aceptan su postulación a la Comisión de Vigilancia antes de la fecha límite que se establezca en el proceso.

En este subproceso deben tenerse en cuenta las siguientes consideraciones adicionales:

- *Cumplimiento de fechas límites tanto en la definición de los candidatos potenciales y en su respectiva confirmación.*
- *Manejo de cancelaciones de postulaciones antes de una fecha límite.*
- *Ajustes en datos que afectan la selección de un posible candidato, por ser parte de los requisitos establecidos en el proceso y reprocesamiento en la determinación de los posibles candidatos.*
- *Sólo se considera un nivel en la elección, es decir de un solo delegado (ej. Se elegirá representante del personal académico).*

2. Generación Padrón Electoral.



Este subproceso tiene como objetivo la definición del padrón electoral.

Según el reglamente del proceso, los investigadores que cumplan con los siguientes requisitos pueden ser ejercer su derecho de voto:

- *Académicos definitivos, interinos o por contrato, cuyo nombramiento haya sido aprobado por el Consejo Técnico de la Investigación Científica.*
- *2 años de antigüedad en el Instituto, calculados al día de la elección.*

- *Incluye investigadores que estén disfrutando de su año sabático, comisión académica o licencias que no interrumpe la antigüedad, o se encuentren adscritos temporalmente a otra entidad*

En este subproceso deben tenerse en cuenta las siguientes consideraciones adicionales:

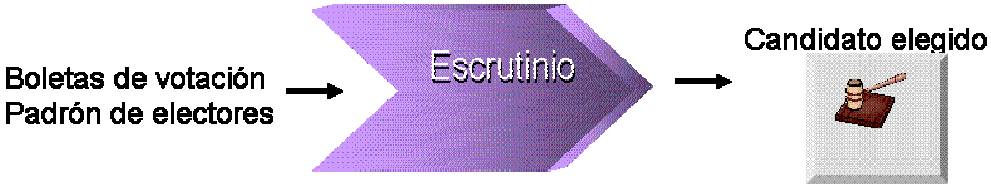
- *Cumplimiento de fechas límites en la definición del padrón electoral inicial y el definitivo después de posibles ajustes en datos que sean requisitos para un posible votante.*
- *Ajustes en datos que afectan la selección de un posible elector en un rango de fechas establecido.*
- *Reprocesamientos en la generación de posibles electores.*

3. Votación.

Las actividades que se realizan dentro de este subproceso son las siguientes:

- *Se establece un rango exacto de fechas y horas en las que podrá realizar la votación.*
- *Se reciben votos previos por email o carta a la Comisión de Vigilancia o la Secretaría Académica.*
- *El proceso se inicia con la firma del acta de instalación.*
- *El votante recibe una boleta de votación, en la cual selecciona candidato de su preferencia.*
- *El votante deposita boleta marcada en la urna.*
- *El votante firma el padrón de electores como constancia de que ya votó (auditoría del proceso).*
- *Se realiza un registro situaciones especiales en el acta de incidentes.*
- *Es importante mencionar que en la normatividad del Instituto no se considera la opción del voto en blanco.*

4. Escrutinio.



Las actividades que se realizan dentro de este subproceso son las siguientes:

- *Apertura de las urnas en una fecha y hora predefinida.*
- *Conteo de boletas de votación y firmas en el padrón de electores por parte de los delegados o escrutadores.*
- *Suma simple de votos por candidato.*
- *Suma de boletas sin selección de candidato y elaboración del acta de votos anulados.*
- *Verificación de votos vs. Número de votantes firmantes en el padrón de electores.*
- *Llenado del acta de escrutinio y de cierre de casilla.*
- *Conteo final de votos y llenado del acta de escrutinio total.*
- *Declaración del resultado por parte del Consejo Técnico de Investigación Científica.*
- *En caso de empate se realizará una segunda elección dentro de los 5 días hábiles siguientes en el mismo horario y "lugar", sólo se incluirán los nombres de los investigadores que hayan empatado en el primer lugar.*
- *No se especifican los posibles incidentes a ser resueltos por la comisión de vigilancia.*
- *Divulgación de los resultados por email y otros medios.*

3.2. Requerimientos de un sistema de votaciones electrónicas

Con base en el estudio de diferentes artículos y documentos de referencia [17] a nivel internacional sobre los requerimientos de los sistemas de votación electrónica y

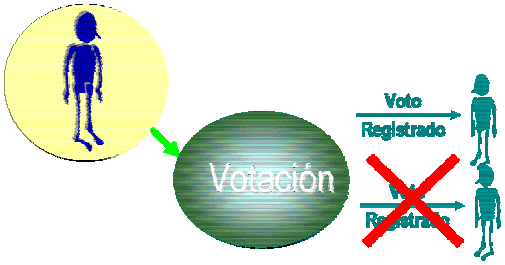
las fallas o limitaciones principales que se han presentado, se extractaron los siguientes requerimientos que se consideran esenciales y cuya revalidación se presenta en el capítulo 5 de este documento:

1. ***Elegibilidad y Autenticación***

Sólo los votantes autorizados deberían ser capaces de votar.

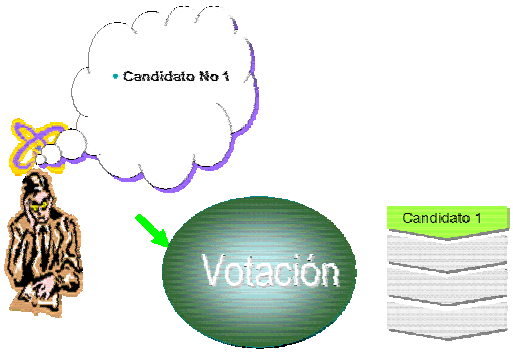
2. ***Unicidad***

Ningún votante podrá votar más de una vez.



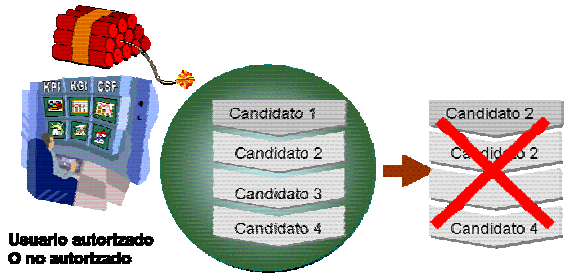
3. ***Exactitud***

Los votos deben ser almacenados correctamente.



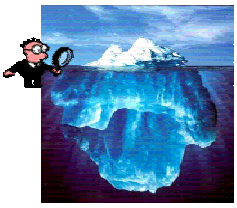
4. ***Integridad***

Los votos no deben ser modificados o borrados sin detección.



5. ***Verificable y Auditable***

Debería ser posible verificar que todos los votos fueron correctamente contados en la elección final.



6. ***Confiable***

Debe trabajar en forma robusta, sin pérdida de votos aunque se presenten numerosas fallas en el sistema o haya pérdida completa de comunicación.

7. ***Secreto y no coercitivo***

Nadie puede determinar como cualquiera votó individualmente. Los votantes no pueden probar como votaron, pues se facilitaría la venta de votos.

8. ***Soporte del voto***

Los votantes podrán ver un soporte de su voto hasta antes de que sea registrado (depositado en la "urna").

9. ***Flexibilidad***

El sistema de elecciones debe permitir definir diferentes tipos de candidatos y votaciones, además de ser compatible con diferentes estándares y tecnologías.

10. ***Conveniencia***

Los votantes no requieren muchas habilidades para poder votar. El acceso al

"sitio" de votación es fácil.

11. ***Transparencia***

Los votantes deberían tener un entendimiento general del proceso de votaciones.

12. ***Escalabilidad***

El sistema puede ser usado por un número pequeño o grande de votantes.

13. ***Velocidad***

El sistema debe ser rápido y conveniente.

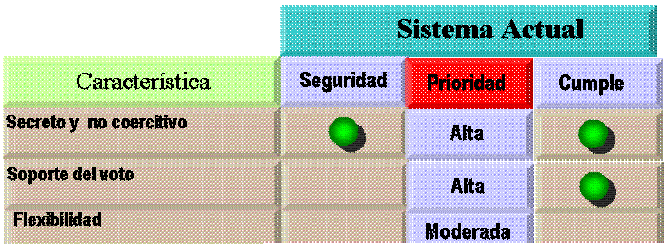
3.3. Cumplimiento de estos requerimientos en el sistema actual.

A continuación se evalúa el nivel de cumplimiento (Satisfactorio o Parcial) de algunos de los requerimientos anteriores en el sistema de votación actual del IMATE, tomando por supuesto en consideración que el proceso actual es manual.

De los requerimientos en consideración se marcan también aquellos relacionados con aspectos de Seguridad y se especifica su nivel de importancia en la columna de prioridad.

	Sistema Actual		
Característica	Seguridad	Prioridad	Cumple
Elegibilidad y Autenticación		Alta	
Unicidad		Alta	
Exactitud		Alta	
Integridad		Alta	
Verificable y Auditable		Alta	
Confiable		Alta	

Satisfactorio Parcial


Sistema Actual
Característica
Seguridad
Prioridad
Cumple
Secreto y no coercitivo
Alta
Soporte del voto
Alta
Flexibilidad
Moderada

CAPITULO 4

4. Configuración de la Seguridad de un producto en Plone.

En este capítulo se describen algunos aspectos técnicos relacionados con la configuración de la seguridad del sistema y algunas de las limitantes del mismo.

Es importante mencionar que lo descrito a continuación parte del supuesto de que el lector conoce las características generales del sistema de administración de contenido Plone [6], la base de datos Zope sobre la que opera y el lenguaje de programación Python.

4.1. Configuración de Seguridad.

Algunos de los aspectos de seguridad que se deben considerar en la construcción de un producto Plone están relacionados con los arquetipos, las vistas y el producto como tal.

4.1.1. Seguridad en Arquetipos

Debido a que las votaciones electrónicas requieren de un buen nivel de seguridad en todas sus etapas. En lo que respecta a la seguridad en los arquetipos se debe considerar la protección de los atributos y de los métodos o funciones, pues:

- *Todos los atributos y funciones dentro de los arquetipos son públicas si no se restringen o si no se utiliza el guión bajo _ .*
- *Todas las funciones pueden ser accesibles desde la URL de la siguiente forma http://ruta/contenido/metodo*

Para controlar el acceso a los métodos o funciones desde un módulo se utiliza la clase ModuleSecurityInfo y para controlar el acceso desde una clase se utiliza la clase ClassSecurityInfo.

Las siguientes funciones se utilizan para controlar el acceso:

- *declarePublic("nombre_del_metodo"). Todos pueden acceder a este método o función.*
- *declarePrivate("nombre_del_metodo"). Es accesible sólo para otras*

funciones de la clase pero no desde el exterior de la clase.

- *declareProtected("permiso","nombre_del_metodo"). Es accesible desde la parte externa de la clase, sólo cuando se tiene el permiso definido.*

A continuación se presenta un ejemplo de la utilización de estas funciones:

```
from AccessControl import ClassSecurityInfo

.....

class arquetipo(BaseContent):

......

    security.declarePrivate('obtenerValor')

    def obtenerValor(self):

        return "devolvio valor"

    vota="X"  ← atributo público

    _vote="1"  ← atributo privado


    security.declarePrivate('funcionPrivada')

    def funcionPrivada(self):

        return self.obtenerValor()


    security.declarePublic('funcionPublica')

    def funcionPublica(self):

        return self.funcionPrivada()


    security.declareProtected("Modify portal content", 'funcionProtegida')

    def funcionProtegida(self):

        return self.funcionPrivada()
```

4.1.2. Seguridad en las Vistas

Todas las vistas son accesibles desde cualquier parte del sitio Plone.

Para acceder a una vista basta con dar una URL válida y al final agregar el nombre de la vista. Ej. http://ruta_valida/NOMBRE_VISTA

Para que no cualquier persona pueda ingresar a una vista se le puede agregar seguridad comprobando si el usuario tiene permisos para ver esa vista.

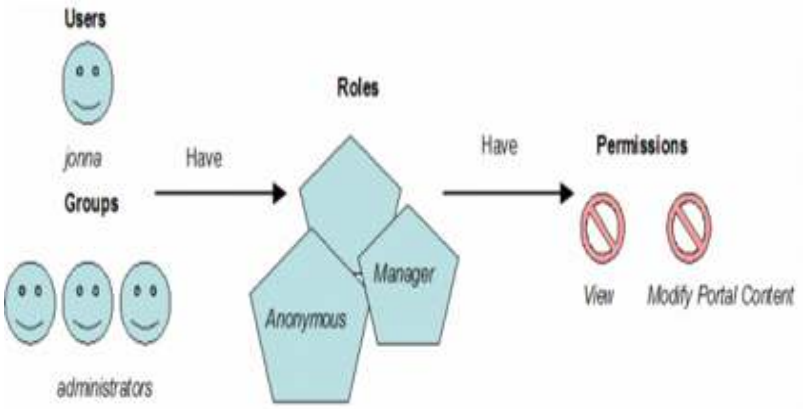
Para comprobar si el usuario que está queriendo acceder a la vista tiene cierto permiso se puede utilizar la función checkPermission de la siguiente forma:

*<**tal:protect** tal:condition="python: not checkPermission('Modify portal content', here)" tal:replace="here/raiseUnauthorized" />*

4.1.3. Seguridad en un producto Plone

Los componentes básicos de la configuración de la seguridad de Plone, utilizados en la solución implementada, son los siguientes [7]:

- *Usuarios: entidad/persona específica que ingresa al sitio.*
- *Roles: permiten la agrupación de permisos. A un usuario se le debe asignar uno o varios roles.*
- *Permisos: autorización para hacer algo, es asignada a roles.*



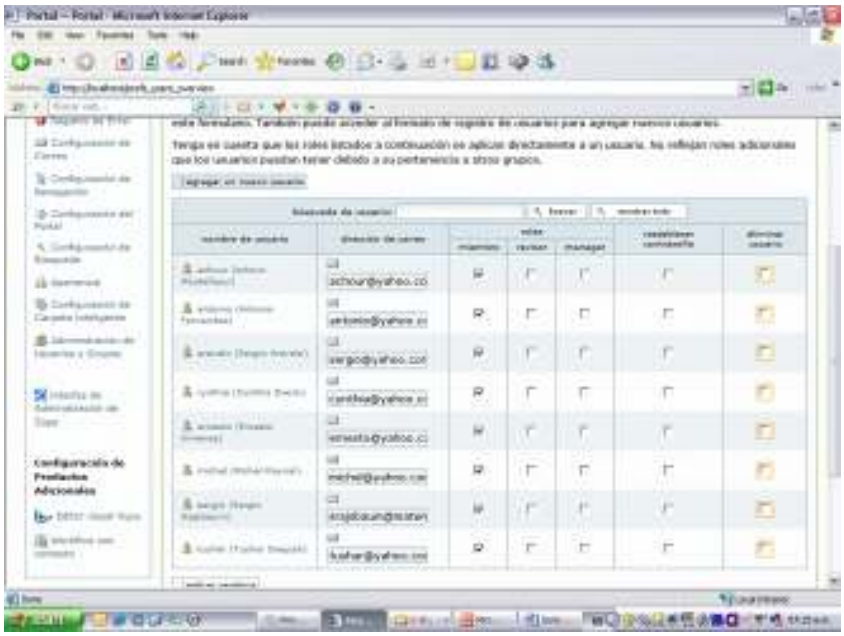
Usuarios

Se encuentran en un "User Folder" llamado "acl_users" dentro de la base de datos de Zope ZODB.

Un usuario Zope está almacenado en una instancia de Zope, por lo que no está relacionado con el usuario del sistema. Tiene las siguientes propiedades: nombre, contraseña y algunos roles asociados.

Cada miembro ingresará al sistema usando una cuenta y contraseña que se le asignará en otro de los sistemas del Instituto.

Interfaz para la actualización de usuarios en Plone:



Roles

Los roles son asignados a los usuarios.

Los roles por defecto son: Anónimo (Zope), Autenticado (Zope), Miembro (Plone), Revisor (Plone), Administrador (Zope) y Propietario (Zope). El rol Miembro es el rol Plone por defecto.

Los usuarios que pertenecen a los roles por defecto tienen las siguientes características:

- *Administrador: tiene permiso total sobre objetos en cualquier estado, puede cambiar la propiedad de cualquier objeto, puede crear y destruir usuarios y modificar sus permisos.*

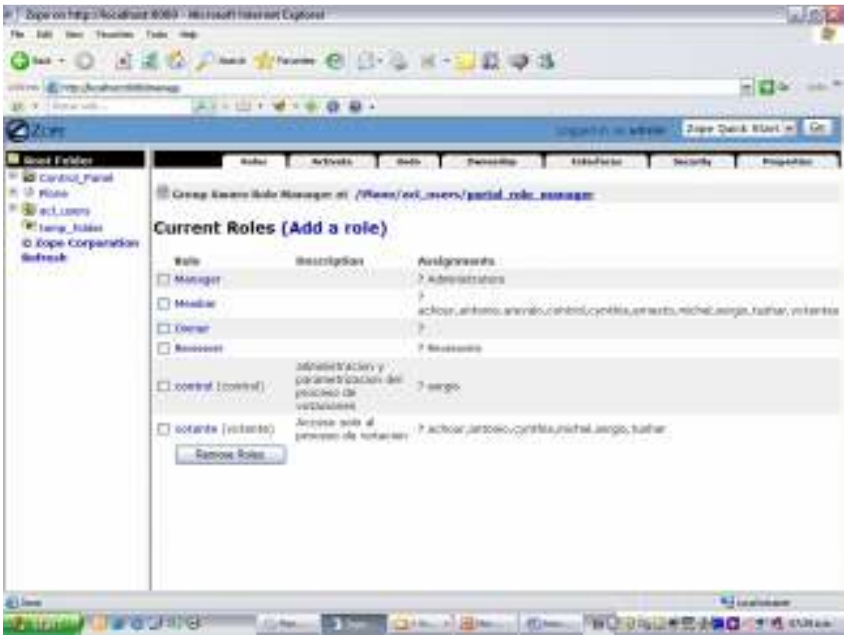
- *Revisor: es un usuario miembro con la potestad de publicar documentos que han sido sometidos a revisión por autores sin permisos para publicar por sí mismos.*

- *Miembro: es un usuario autenticado en el portal, y tiene derecho a ver los contenidos que están en cualquier estado menos privado. No puede publicar.*

- *Propietario: un miembro es propietario de los objetos de los que es autor, pero también puede ser declarado propietario de otros objetos y comparte con su autor los derechos de edición y gestión en igualdad de condiciones.*

*Para el **sistema propuesto** se consideraron dos roles adicionales, elector que se le asignará a todos los miembros y control que se asignará a los funcionarios que pertenezcan a alguna de grupos de control y monitoreo del proceso.*

Es importante mencionar que los miembros del sitio que sean electores o candidatos del proceso sólo deben tener asociado el rol de miembro, pero no se les deben asignar roles de revisor ni de administrador.



Permisos

Nivel más bajo de configuración de la seguridad que corresponde a cadenas que pueden asociarse con objetos o atributos específicos o métodos de objetos.

Se asignan a roles mediante la interfaz de administración de Zope, no pueden asignarse directamente a usuarios.

La única forma real de identificarlos es revisando las opciones que están configuradas en el código Python.

Los 4 permisos básicos que se consideran en la configuración de seguridad del sistema de votaciones son los siguientes:

Permission Name	Description
Access contents information	This grants the ability to access an object's metadata and content. Users must have the View permission to view the object. If a user has the View permission, but not the *Access contents information* permission, on an object, the user can find that object in search results but cannot access the complete content of the object.
Add portal content	This grants the ability to add instances of the existing portal content types to the site (such as Document, News Item, etc.).
Add portal folders	This grants the ability to add folders to the site.
Add portal member	This grants the ability to join the Plone site and become a Member.
Copy or Move	This grants the ability to cut/copy/paste objects from one folder to another.
Delete objects	This grants the ability to delete an object from a folder.
List folder contents	This grants the ability to list the content of a folder.
List portal members	This grants the ability to list all the members of the site.
Modify portal content	This grants the ability to edit and update portal content information.
Reply to item	This grants the ability to reply to discussable content.
Request review	This grants the ability to the content creators to submit the content for approval.

Adicionalmente, a nivel del código Python se deben configurar nuevos permisos para tener una apropiada segregación de funciones en el proceso de votaciones.

Debido a que los permisos se asignan a los roles y a los usuarios se les puede asignar diferentes roles, se pueden considerar que se cumplen con algunas de las propiedades de RBAC Role Based Access Control:

- *Los usuarios no gozan de acceso discrecional a los objetos del sistema, los permisos se otorgan a roles*
- *A los usuarios se les asignan el/los roles apropiados.*
- *A los usuarios puede asignársele un rol dado según sus responsabilidades y capacidades,*
- *A los usuarios se les puede reasignar un rol a otro sin modificar la estructura de control de acceso.*
- *Se pueden otorgar nuevos permisos a un rol.*
- *Se pueden revocar los permisos de un rol, sin tener que ocuparse de todos los usuarios involucrados*

Workflow

Por otra parte debe considerarse otro concepto que se basa en Plone

denominado Workflow, el cual permite manejar los permisos en forma automática para diferentes tipos de contenidos.

Permite manejar diferentes asociaciones de permisos a roles dependiendo de los estados, además de poder programar scripts en transiciones específicas de un estado a otro.

4.2. Limitaciones o Consideraciones de la Seguridad de Plone.

Las principales limitaciones o aspectos de mayor complejidad que deben ser considerados en la seguridad de plone se presentan a continuación:

4.2.1. Definición Granular de Permisos

Los permisos están basados en Zope y deben ser definidos por el desarrollador de una forma muy granulada, lo que hace que este proceso sea complejo.

En el caso de la solución implementada se debieron establecer entre otros los siguientes permisos, lo que quita algo de flexibilidad al sistema al querer ser utilizado en otro tipo de organización con una normatividad diferente, claro que desde otro punto de vista podría ser considerado como una buena opción para refinar la segregación de funciones:

- *Ver la vista principal de la votación.*
- *Editar campos del esquema de la votación.*
- *Ver lista de electores*
- *Ver lista de candidatos*
- *ver bitácoras de votación*
- *Hacer escrutinio*
- *Ver resultados previos (no publicados).*
- *Crear evento*
- *Eliminar evento*
- *Modificar evento*
- *Cerrar padrón de electores*
- *Confirmar padrón de candidatos inicial*
- *Cerrar padrón de candidatos*
- *Crear archivo de candidatos definitivo*
- *Ver url, slogan y comentarios de candidatos confirmados*
- *Modificar url, slogan y comentarios de candidatos confirmados*

- *Ver bitácoras*
- *Ver resultados*
- *Crear usuarios*
- *Modificar resultados de los usuarios*

4.2.2. Diseño con UML Seguro

Las herramientas utilizadas, como argouml, para el diseño de los arquetipos para la versión de Plone 2.5 tienen incorporadas las propiedades del Lenguaje de Modelamiento Unificado UML estándar, pero las propiedades de UMLSec, las cuales deben ser consideradas en trabajos futuros de este proyecto de Tesis.

CAPITULO 5

5. Validación de los Requerimientos de un Sistema de Votaciones Electrónicas.

En este capítulo se describen los resultados de la aplicación del método Delphi con expertos en seguridad y consultoría informática de diferentes países de Ibero América, respecto a la validación de las características claves que deben ser consideradas en la implementación de un sistema de votación electrónica.

Antes de presentar los resultados se describen algunas características generales del método Delphi y las propiedades de dispersión estadística consideradas en la toma de decisión sobre las características claves del sistema de votación electrónica.

5.1. Método Delphi

El método Delphi fue ideado originalmente a comienzos de los años 50 en el Centro de Investigación estadounidense RAND Corporation por Olaf Helmer y Theodore J. Gordon, como un instrumento para realizar predicciones sobre un caso de catástrofe nuclear [14].

Es un método para estructurar el proceso de comunicación de un grupo de personas para que funcionen efectivamente lidiando con problemas complejos [12].

Para iniciar el proceso se debe realizar una selección de un grupo de expertos a los que se les pregunta su opinión sobre cuestiones referidas al problema que se quiere analizar. Las estimaciones de los expertos se realizan en rondas sucesivas, anónimas, con el objetivo de tratar de conseguir consenso, pero con la máxima autonomía por parte de los participantes [14].

La capacidad de obtener conclusiones con el método Delphi se basa en la utilización sistemática de un juicio intuitivo emitido por un grupo de expertos. Es decir, el método Delphi procede por medio de la interrogación a expertos con la ayuda de cuestionarios sucesivos, a fin de poner de manifiesto convergencias de opiniones y deducir eventuales consensos. La encuesta se lleva a cabo de una manera anónima haciendo uso del correo electrónico o mediante cuestionarios web establecidos al efecto para evitar los efectos de "líderes".

El método Delphi pretende extraer y maximizar las ventajas que presentan los métodos basados en grupos de expertos y minimizar sus inconvenientes. Para ello se aprovecha la sinergia del debate en el grupo y se eliminan las interacciones sociales

indeseables que existen dentro de todo grupo. De esta forma se espera obtener un consenso lo más fiable posible del grupo de expertos

Este método presenta tres características fundamentales [12]:

- *Anonimato: Durante un Delphi, ningún experto conoce la identidad de los otros que componen el grupo de debate. Esto tiene una serie de aspectos positivos, como son:*
    - *Impide la posibilidad de que un miembro del grupo sea influenciado por la reputación de otro de los miembros o por el peso que supone oponerse a la mayoría. La única influencia posible es la de la congruencia de los argumentos.*
    - *Permite que un miembro pueda cambiar sus opiniones sin que eso suponga una pérdida de imagen.*
    - *El experto puede defender sus argumentos con la tranquilidad que da saber que en caso de que sean erróneos, su equivocación no va a ser conocida por los otros expertos.*
- *Iteración y realimentación controlada: La iteración se consigue al presentar varias veces el mismo cuestionario. Como, además, se van presentando los resultados obtenidos con los cuestionarios anteriores, se consigue que los expertos vayan conociendo los distintos puntos de vista y puedan ir modificando su opinión si los argumentos presentados les parecen más apropiados que los suyos.*
- *Respuesta del grupo en forma estadística: La información que se presenta a los expertos no es sólo el punto de vista de la mayoría, sino que se presentan todas las opiniones indicando el grado de acuerdo que se ha obtenido.*

En la aplicación del método Delphi existen los siguientes participantes [12]:

- *Los que deciden: el o los individuos que esperan obtener el producto del ejercicio.*
- *Los moderadores: quienes diseñan los cuestionarios, resumen los resultados y conducen el proceso para satisfacer los requerimientos de "los que deciden".*
- *Los expertos : aquellos cuya opinión se necesita y a quienes se les hacen las preguntas*

La metodología que se sigue en forma muy general es la siguiente:

- *Los moderadores preparan una serie de preguntas para los expertos*
- *Se solicita y se evalúa su opinión sin que se comuniquen entre ellos*
- *Se prepara una segunda serie de preguntas*
- *Se solicita y se evalúa su opinión sin que se comuniquen entre ellos*
- *y así sucesivamente hasta lograr consenso o encontrar las causas que no lo permiten.*

En particular el método Delphi numérico tienes las siguientes características [15]:

- *Facilita llegar a un consenso con menor número de iteraciones, pero sin descuidar la calidad del producto.*
- *Se efectúa una agregación estadística de los resultados expresados, a manera de una calificación de aceptación de un enunciado.*
- *El consenso se identifica mediante el promedio de las calificaciones.*
- *Se mide el consenso a través de la dispersión de las respuestas*

5.2. Conceptos estadísticos básicos

Para concluir sobre el nivel de dispersión de los resultados del método Delphi numérico se deben comprender bien los siguientes conceptos estadísticos básicos:

Desviación estándar

Es un índice numérico de la dispersión de un conjunto de datos (o población), mientras mayor es la desviación estándar, mayor es la dispersión de la población. Es un promedio de las desviaciones individuales de cada observación con respecto a la media de una distribución [13].

Cálculo: Primero se mide la diferencia entre cada valor del conjunto de datos y la media del conjunto de datos. Luego, se suman todas estas diferencias individuales para dar el total de todas las diferencias. Por último, se divide el resultado por el número total de observaciones para llegar a un promedio de las distancias entre cada observación individual y la media

Cuando se va a determinar si un grupo de medidas está de acuerdo con el

modelo teórico, la desviación estándar de esas medidas es de vital importancia: si la media de las medidas está demasiado alejada de la predicción (con la distancia medida en desviaciones estándar), entonces consideramos que las medidas contradicen la teoría. Esto es de esperarse ya que las mediciones caen fuera del rango de valores de los cuales sería razonable esperar que ocurrieran si el modelo teórico fuera correcto.

Varianza

Resultado estadístico de dispersión que mide el grado de variabilidad que sintetiza el grado de homogeneidad o heterogeneidad de las diferencias individuales entre los casos de una muestra (o de varias muestras) respecto de una o varias variables numéricas continuas o cuantitativas.

Corresponde al cuadrado de la desviación estándar.

5.3. Requerimientos propuestos para un sistema de votaciones electrónicas

Con base en el análisis de los artículos [1],[2],[3], [4] y [17] se llegó a una primera aproximación de los requerimientos generales que debe cumplir un sistema de votación electrónica, los cuales se tomaron de base para el proceso de decisión basado en el método Delphi y se presentan a continuación.

Id. Aspecto	***Nombre Aspecto***	***Descripción General***	***Detalles de Implementación***
01	*Elegibilidad y Autenticación*	*Sólo los votantes autorizados deberían ser capaces de votar*	*Establecimiento de una contraseña para los votantes que cumplan las condiciones requeridas, la cual debe usarse para poder registrar el voto.*
02	*Unicidad*	*Ningún votante podrá votar más de una vez*	*Mediante una bitácora en el sistema debe controlarse que un usuario no registre más de un voto*
03	*Políticas de notificación*	*Los votantes deben ser informados acerca de su estado y los cambios que afecten su habilidad para votar*	*Se deben notificar con anterioridad al proceso de votación, los requisitos para que un usuario pueda votar y la lista de votantes autorizados.*

Id. Aspecto	*Nombre Aspecto*	*Descripción General*	*Detalles de Implementación*
04	*Exactitud*	*Los votos deben ser almacenados correctamente*	*Mediante una bitácora en el sistema debe registrarse el voto para el candidato que el usuario decidió*
05	*Confiable*	*Debe trabajar en forma robusta, sin pérdida de votos.*	*Aunque se presenten numerosas fallas en el sistema o haya pérdida completa de comunicación no deben alterarse ninguna de las bitácoras*
06	*Secreto y no coercitivo*	*Nadie puede determinar como cualquiera votó individualmente.*	*Los votantes no pueden probar como votaron y las bitácoras en el sistema no permitirán ligar un voto con un votante.*
07	*Soporte del voto*	*Los votantes podrán ver un soporte de su voto hasta antes de que sea registrado*	*En la pantalla del sistema de votación se mostrará una "boleta" con el voto que se va a registrar para que el usuario verifique su selección.*
08	*Verificable*	*Debería ser posible verificar en forma independiente que todos los votos fueron correctamente contados en la elección final.*	*La bitácora en el sistema donde se registrarán los votos permitirá asegurar que todos los votos fueron considerados en el escrutinio final.*
09	*Integridad*	*Los votos no deben ser modificados o borrados sin detección*	*Debe establecerse un mecanismo para garantizar la integridad de las bitácoras del sistema.*
10	*Conveniencia*	*El acceso al ¨sitio¨ de votación es fácil y los votantes no requieren muchas habilidades para poder votar.*	*El acceso al sistema de votación electrónica es muy sencillo y una vez el usuario se autentique el proceso de votación será muy simple.*
11	*Transparencia*	*Los votantes deberían tener un entendimiento general del proceso de votaciones,*	*El sistema de votación electrónica debe implementarse en una plataforma de software libre y su*

Id. Aspecto	Nombre Aspecto	Descripción General	Detalles de Implementación
		incluyendo procedimientos internos e interfaces	*lógica de procesamiento debe ser transparente.*
12	*Escalabilidad*	*El sistema puede ser usado por un número pequeño o grande de votantes*	*La plataforma en la que se implemente el sistema debe ser robusta y usable para procesos masivos de votación.*
13	*Velocidad*	*El sistema debe ser rápido y conveniente.*	*El proceso para registrar un voto no requerirá una espera larga para el usuario.*

5.4. Resultados método Delphi Fase I

Gracias a la colaboración de una funcionaria de ISACA Internacional www.isaca.org se logró la participación de 17 del total de 18 expertos incluidos en este proceso.

Los expertos participantes tenían las siguientes características:

- *Países: México, Colombia, Costa Rica, Estados Unidos, Portugal, España, Argentina y Chile.*
- *Tipos de Organizaciones: Empresas de Consultoría Tecnológica, Auditoría, Entidades Financieras, Telecomunicaciones, Servicios, entre otras.*
- *Cargos: Directores Generales, Consultores, Auditores y expertos en seguridad.*

A continuación se relacionan los nombres de las organizaciones, cargos y países de los 18 expertos participantes, asignando a cada uno un identificador único (id):

Id. Participante	*1*	*2*	*3*	*4*	*5*
Organización	*Comcel S.A*	*Grupo Caixa Geral Depósitos*	*CGD*	*Endesa*	*ISACA: Serving IT Governance Professionals (www.isaca.org)*
Cargo	*Coordinador Quality Assurance*	*Sogrupo SI*	*Internal Information Auditing Manager.*	*Jefe de equipo – Auditoria de Sistemas*	*Manager-Certification Study Program And Educational Development*
País	*Colombia*	*Portugal*	*Portugal*	*España*	*Estados Unidos*

Id. Participante	*6*	*7*	*8*	*9*	*10*
Organización	*Banco de Costa Rica*	*Neoris Consulting*	*BNA*	*Combanc S.A.*	*Grupo Gestor Cumbres*
Cargo	*Gerente de División de Procesos*	*Sr Business Consultant*	*Gerente Departamental Organización y Procesos*	*Auditor General*	*Presidente y Socio Director*
País	*Costa Rica*	*México*	*Argentina*	*Chile*	*Costa Rica*

Id. Participante	*11*	*12*	*13*	*14*	*15*
Organización	*Grupo Salinas*	*Scitum S.A. de C.V.*	*AnyHelp International*	*Independiente*	*Independiente*
Cargo	*Director de Auditoría en Informática*	*Gerente de operaciones UniverScitum*	*General Manager*	*Director*	*Director*
País	*México*	*México*	*España*	*Colombia*	*Argentina*

Id. Participante	***16***	***17***	***18***
Organización	*Asentti*	*Independiente*	*Chispa*
Cargo	*Director*	*Director*	*Director*
País	*México*	*México*	*México*

Lo primero que se realizó fue enviar la siguiente solicitud que a los expertos participantes:

SOLICITUD

"Muchas gracias por su apoyo como experto en este proceso de decisión sobre los aspectos más relevantes a considerar en la implementación de un sistema de votación electrónica.

Como le comenté en el mensaje inicial se va a aplicar el método Delphi, por lo que le pido por favor que califique de 0 a 10 cada uno de los 13 aspectos propuestos, dependiendo del grado de relevancia que usted considere, donde10 es el más alto nivel y 0 el más bajo.

Si usted lo considera relevante puede incluir los comentarios que desee en cada aspecto.

Una vez procese su opinión y la de los demás expertos le pediré por favor que califique de nuevo algunos de los temas para los cuales no se obtenga consenso.

En el caso que considere que la descripción general o los detalles de implementación de alguno de los aspectos se puede mejorar por favor me lo hace saber en la columna de comentarios".

Muchas gracias por su apoyo en este proceso.,

saludos,

Alexander Zapata

Maestría en Ciencias de la Computación

UNAM

****

Posteriormente se procesaron las calificaciones por cada uno de los requerimientos del sistema de votaciones electrónicas y se obtuvo el promedio,

desviación estándar y varianza.

Se procesaron los comentarios de cada uno de los expertos.

Finalmente se seleccionaron los requerimientos con promedio >= 8.0 y desviación estándar <= 2.0. Valores que determinan consenso entre los expertos.

Las calificaciones de los 18 expertos para cada uno de los requerimientos son las siguientes:

Id. Aspecto / Id. Participante	***Nombre Aspecto***	***1***	***2***	***3***	***4***	***5***
1	*Elegibilidad y Autenticación*	*8*	*10*	*8*	*10*	*10*
2	*Unicidad*	*10*	*10*	*10*	*10*	*10*
3	*Políticas de notificación*	*8*	*8*	*8*	*10*	*10*
4	*Exactitud*	*8*	*10*	*10*	*10*	*10*
5	*Confiable*	*7*	*10*	*10*	*10*	*10*
6	*Secreto y no coercitivo*	*8*	*10*	*9*	*10*	*10*
7	*Soporte del voto*	*10*	*7*	*10*	*10*	*3*
8	*Verificable*	*8*	*10*	*10*	*10*	*10*
9	*Integridad*	*8*	*10*	*10*	*10*	*10*
10	*Conveniencia*	*10*	*6*	*8*	*10*	*10*
11	*Transparencia*	*8*	*5*	*10*	*4*	*4*
12	*Escalabilidad*	*8*	*8*	*8*	*10*	*10*
13	*Velocidad*	*8*	*6*	*7*	*10*	*10*

Id. Aspecto / Id. Participante	*Nombre Aspecto*	*6*	*7*	*8*	*9*	*10*
1	*Elegibilidad y Autenticación*	*10*	*10*	*10*	*10*	*9*
2	*Unicidad*	*10*	*10*	*10*	*10*	*8*
3	*Políticas de notificación*	*10*	*5*	*8*	*10*	*8*
4	*Exactitud*	*10*	*10*	*10*	*10*	*8*
5	*Confiable*	*10*	*10*	*10*	*10*	*7*
6	*Secreto y no coercitivo*	*10*	*10*	*10*	*10*	*7*
7	*Soporte del voto*	*10*	*5*	*9*	*6*	*8*
8	*Verificable*	*10*	*10*	*10*	*10*	
9	*Integridad*	*10*	*10*	*10*	*10*	*9*
10	*Conveniencia*	*9*	*5*	*9*	*8*	*7*
11	*Transparencia*	*9*	*7*	*7*	*8*	*6*
12	*Escalabilidad*	*10*	*5*	*8*	*10*	*6*
13	*Velocidad*	*9*	*7*	*9*	*8*	*7*

Id. Aspecto / Id. Participante	*Nombre Aspecto*	*11*	*12*	*13*	*14*	*15*
1	*Elegibilidad y Autenticación*	*10*	*9*	*10*	*10*	*5*
2	*Unicidad*	*10*	*10*	*10*	*10*	*10*
3	*Políticas de notificación*	*8*	*2*	*10*	*10*	*10*

Id. Aspecto / Id. Participante	Nombre Aspecto	11	12	13	14	15
4	Exactitud	10	10	7	10	10
5	Confiable	10	10	10	10	10
6	Secreto y no coercitivo	9	9	10	10	10
7	Soporte del voto	8	3	8	10	10
8	Verificable	10	10	10	10	8
9	Integridad	10	10	10	10	6
10	Conveniencia	9	10	8	8	8
11	Transparencia	9	7	7	10	8
12	Escalabilidad	9	8	10	10	10
13	Velocidad	9	8	9	10	10

Id. Aspecto / Id. Participante	Nombre Aspecto	16	17	18
1	Elegibilidad y Autenticación	10	8	10
2	Unicidad	8	8	8
3	Políticas de notificación	9	5	5
4	Exactitud	9	9	10
5	Confiable	10	9	10
6	Secreto y no coercitivo	10	7	10

Id. Aspecto / Id. Participante	Nombre Aspecto	16	17	18
7	Soporte del voto	8	6	8
8	Verificable	9	8	8
9	Integridad	10	9	10
10	Conveniencia	10	7	5
11	Transparencia	3	7	8
12	Escalabilidad	9	7	8
13	Velocidad	8	8	8

Con base en las calificaciones anteriores el promedio, la desviación estándar y la varianza resultantes son los siguientes:

Id. Aspecto	Nombre	Promedio	Desviación Estándar	Varianza	Respuestas Dispersas Dif > 3
1	Elegibilidad y Autenticación	**9.28**	**1.32**	**1.74**	**1**
2	Unicidad	**9.56**	**0.86**	**0.73**	**0**
3	Políticas de notificación	**8.00**	**2.33**	**5.41**	**4**
4	Exactitud	**9.50**	**0.92**	**0.85**	**0**
5	Confiable	**9.61**	**0.98**	**0.96**	**0**
6	Secreto y no coercitivo	**9.39**	**1.04**	**1.08**	**0**
7	Soporte del voto	**7.72**	**2.32**	**5.39**	**2**
8	Verificable	**9.47**	**0.87**	**0.76**	**0**

Id. Aspecto	Nombre	Promedio	Desviación Estándar	Varianza	Respuestas Dispersas Dif > 3
9	Integridad	**9.56**	**1.04**	**1.08**	**0**
10	Conveniencia	**8.17**	**1.65**	**2.74**	**2**
11	Transparencia	**7.06**	**2.01**	**4.06**	**3**
12	Escalabilidad	**8.56**	**1.50**	**2.26**	**1**
13	Velocidad	**8.39**	**1.20**	**1.43**	**0**

Con base en los resultados anteriores se concluye que 10 de los 13 requerimientos son aceptados por los expertos y 3 (3,7 y11) deben someterse a una segunda ronda de decisión, pues no se obtuvo consenso, al tener un promedio menor a 8.0 o una desviación estándar superior a 2.0

5.5. Resultados método Delphi Fase II

De los 18 expertos que respondieron en la Fase I del método Delphi, se obtuvieron 8 respuestas con las cuales se pudo llegar a una conclusión respecto a la aceptación o rechazo de los 3 requerimientos en análisis.

La descripción de los 3 requerimientos se ajustó con base en los comentarios recibidos en la primera fase, con el fin de que ésta considerara algunos criterios adicionales para que los expertos pudieran emitir una nueva calificación.

Id. Aspec	Nombre	Descripción General	Detalles de Implementación
3	Políticas de notificación	Los requisitos para figurar en los padrones (listas de votantes) electorales deben difundirse ampliamente dentro de la población meta. Debe establecerse una fecha límite para integrar el padrón, luego de la cual generalmente no se permiten inclusiones aunque si exclusiones. Este	Se deben notificar con anterioridad al proceso de votación, los requisitos para que un usuario pueda votar y la lista de votantes autorizados. Esto exige contar con un sistema de registro electoral independiente del sistema de conteo electrónico. Un sitio web con la información requerida por los votantes es el

Id. Aspec	Nombre	Descripción General	Detalles de Implementación
		debe comunicarse oportunamente. *La lista de votantes o padrón debe mantenerse confidencial para evitar que sea usada para fines de coacción*	*ideal.*
7	*Soporte del voto*	*Los votantes podrán ver en la pantalla, la cual debe tener un esquema de seguridad física, un soporte de su voto hasta antes de que sea registrado*	*En la pantalla del sistema de votación se mostrará una "boleta" con el voto que se va a registrar para que el usuario verifique su selección.* *Este mecanismo es requerido para que el votante esté seguro del correcto registro del voto.*
11	*Transparencia*	*Los votantes deberían tener un entendimiento general de la lógica interna del proceso de votaciones, incluyendo procedimientos e interfaces.*	*El sistema de votación electrónica debe implementarse en una plataforma de software libre y su lógica de procesamiento debe darse a conocer a toda la población de votantes.*

Las calificaciones de los expertos en esta segunda fase fueron las siguientes:

Id. Aspecto	Nombre	Cal 1	Cal 2	Cal 3	Cal 4	Cal 5	Cal 6	Cal 7	Cal 8
3	*Políticas de notificación*	*9*	*9*	*9*	*6*	*8*	*8*	*8*	*10*
7	*Soporte del voto*	*9*	*8*	*9*	*5*	*10*	*7*	*9*	*9*
11	*Transparencia*	*5*	*7*	*5*	*1*	*5*	*5*	*7*	*9*

Con base en las calificaciones anteriores el promedio, la desviación estándar y la varianza resultantes son los siguientes:

Num	Nombre Aspecto	Promedio	Desviación Estándar	Varianza	Respuestas Dispersas
3	Políticas de notificación	8.38	1.19	1.41	0
7	Soporte del voto	8.25	1.58	2.50	1
11	Transparencia	5.50	2.33	5.43	2

5.6. Resultados Finales método Delphi

Con base en los resultados de la aplicación del método Delphi en sus dos fases se logró determinar que 12 de los 13 requerimientos propuestos fueron considerados por los expertos como claves en la implementación de un sistema de votación electrónica.

A continuación se presentan en orden de relevancia cada uno de los 12 requerimientos con algunos comentarios sobre aspectos adicionales a considerar según la opinión de los expertos.

1. Confiable

Aspectos adicionales a considerar en la implementación según los expertos:

- *El sistema debe estar en capacidad de mantener persistencia y garantizar las propiedades de las transacciones ACID.*
- *Es recomendable cifrar los archivos de bitácoras.*
- *Se debe tener una bitácora de transacciones con timestamp.*
- *La continuidad del proceso debe garantizarse, incluyendo las opciones para gestionar los fallos, técnicos, administrativos, procedimentales o humanos. Llevar actas de resolución de incidentes y establecer el proceso de cómo deben ser atendidas, cuándo y a quién escalar la resolución del incidente*

2. Unicidad

Aspectos adicionales a considerar en la implementación según los expertos:

- *Además de la bitácora el sistema deberá contar con controles automáticos para evitar la votación múltiple, ya que la bitácora es un control posterior y es necesario prevenir.*
- *El número de identificación una vez se presente el ciudadano y vote debe quedar bloqueado.*
- *Debe usarse una bandera que indica si ya votó o no, debe estar inicializada con una marca de que no ha votado*

3. Integridad

Aspectos adicionales a considerar en la implementación según los expertos:

- *Es recomendable cifrar los archivos de bitácoras.*
- *Este es uno de los elementos cruciales del proceso y puede convertirse en una evaluación sumamente compleja y detallada.*
- *Puede ser a través de transacciones firmadas digitalmente y hashes periódicos de los votos en el tiempo.*
- *Además de detectar, se debe prevenir toda modificación no autorizada.*
- *En realidad hay que confiar que los votos fueron contados correctamente.*
- *La única ayuda que puede haber es una auditoría detallada de la aplicación, aunque ésta, por supuesto, no es infalible.*

4. Exactitud

Aspectos adicionales a considerar en la implementación según los expertos:

- *Cifrar el archivo de votos.*
- *No se debe asociar la identificación del ciudadano con el candidato seleccionado, simplemente debe sumarse un voto más.*
- *Se debería de llevar más de una bitácora, bitácoras locales a los sistemas de votación, mas una centralizada con hashes periódicos de los votos realizados.*

- *Evitar la existencia de una secuencia de votantes en una bitácora que rompa la confidencialidad del voto.*

5. Verificable

Aspectos adicionales a considerar en la implementación según los expertos:

- *Los votos deberían ser registrados simultáneamente en dos sistemas independientes que no se comuniquen entre sí. Al final del proceso ambos deben generar los mismos resultados. (se considerará en versiones posteriores)*
- *Debe diseñarse un mecanismo de auditoría alrededor del computador, pues la urna debe funcionar como una caja negra.*
- *Se podría llevar un serial de cada transacción, tal como se hace cuando se generan certificados por una autoridad certificadora.*

6. Secreto y no coercitivo

Aspectos adicionales a considerar en la implementación según los expertos:

- *El anonimato a la hora de votar debe mantenerse, y esto es lo más difícil de hacer llegar al futuro usuario. Trasmitir esto es difícil, pero a la vez es clave para la aceptación del sistema.*
- *Las bitácoras deben cifrarse.*
- *En caso de interrupciones en medio de la emisión de un sufragio, y restablecer o poder revertir votaciones parciales.*
- *El sistema de autenticación de usuarios y su base de datos deberá estar totalmente desligado de la base de datos de votos.*

7. Elegibilidad y Autenticación

Aspectos adicionales a considerar en la implementación según los expertos:

- *Personas en capacidad de votar, no deben quedar excluidas por problemas de contraseñas.*
- *Contraseña + "otro elemento" que posibilitara una autenticación doble. (se considerará en versiones posteriores)*

- *Firma Digital es un concepto que se puede considerar en un futuro*
- *Una contraseña puede ser crackeable, sería más confiable en un futuro utilizar un medio biométrico u otro sistema más robusto.*
- *Imprescindible que la autenticación evite asociar el contenido del voto del elector.*
- *Implica la construcción de un padrón confiable.*

8. Escalabilidad

Aspectos adicionales a considerar en la implementación según los expertos:

- *Analizar la disponibilidad de comunicaciones en sitios remotos y verificar la no degradación de la capacidad de procesamiento por el incremento de transacciones.*
- *Se puede manejar en varias capas de acuerdo al sistema, pero básicamente se debe contar con un sistema de balanceo en el front-end.*

9. Velocidad

Aspectos adicionales a considerar en la implementación según los expertos:

- *El sistema debe operar en "tiempo-real" o sea que la velocidad debe asegurar que no se produzcan colas de votantes.*
- *Hasta medio minuto es tolerable, lo que en un sistema bien hecho y correctamente dimensionado es muchísimo tiempo.*
- *No obstante lo anterior, este requerimiento puede estar muy relacionado con otros factores exógenos como distribución de electores, horas pico, distribución geográfica y otros que vale la pena considerar en conjunto.*

10. Conveniencia

Aspectos adicionales a considerar en la implementación según los expertos:

- *Es clave, ya que personas mayores "evitan" o tienen una resistencia al cambio fuerte.*
- *Se debe cuidar mucho la interfaz.*

- *En un futuro, desarrollar procedimientos para las personas no videntes o con otras discapacidades motoras o de otra índole.*

11. Políticas de Notificación

Aspectos adicionales a considerar en la implementación según los expertos:

- *Este requerimiento se ubica en el número 11 porque fue ajustado y aprobado en la segunda fase del método Delphi.*
- *Los usuarios deben conocer con antelación su situación para poder votar de modo que exista seguridad sobre la correcta elaboración del padrón electoral, es indispensable también que el usuario conozca los requisitos para poder votar.*
- *El votante debe tener la facilidad de poder consultar su estado en todo momento.*

12. Soporte del voto

Aspectos adicionales a considerar en la implementación según los expertos:

- *Este requerimiento fue ajustado y aprobado en la segunda fase del método Delphi.*
- *Se debe tener cuidado con la técnica de ingeniería social Shoulder Surfing.*
- *Al usuario deberíamos asegurarle la integridad e intención final de su voto, al igual que cuando hacemos una transferencia electrónica de fondos y la máquina nos pregunta "¿está seguro de hacer esta transacción…?".*

Es importante mencionar finalmente que la aplicación del método Delphi fue muy útil para la determinación de los requerimientos que mejoran el nivel de seguridad de un sistema de votación electrónica y para detallar algunos aspectos claves en su implementación.

Algunas sugerencias de los expertos requieren inversión en mecanismos de seguridad que robustecen un sistema de votación electrónica.

Respecto al requerimiento propuesto de transparencia, varios de los expertos comentaron que es mejor considerar que el sistema sea auditado y certificado por un tercero

CAPITULO 6

6. Sistema de Votaciones Implementado

En este capítulo se describe el sistema de votaciones implementado en Plone, empezando por la explicación de las características claves del proceso, continuando con las propiedades de los productos que lo conforman y la descripción del protocolo de votaciones seguido, y finalizando con la especificación de los requerimientos técnicos necesarios para su instalación.

6.1. Características claves del Proceso

Los roles de los usuarios que intervienen en el proceso, las fases del proceso, las bitácoras o archivos históricos manejados y los diferentes estados del proceso que pueden visualizar los usuarios del sistema, se describen a continuación.

6.1.1. Tipos de Usuarios

En el proceso se manejan los siguientes tres tipos de usuarios:

- *Administrador de la votación: corresponde al usuario que agrega el objeto tipo votación, configura todos los parámetros y fechas claves de la misma que sólo pueden ser confirmados por la comisión de vigilancia, registra su clave pública, crea el evento y el foro de preguntas y respuestas de la votación, genera los padrones de electores y candidatos de acuerdo con la convocatoria de la elección pero estos sólo pueden ser confirmados y modificados después de su publicación por la comisión de vigilancia; también puede consultar las bitácoras y apoyar parte del conteo de los votos en lo que respecta a la primera de dos rondas de descifrado de votos de la "urna".*

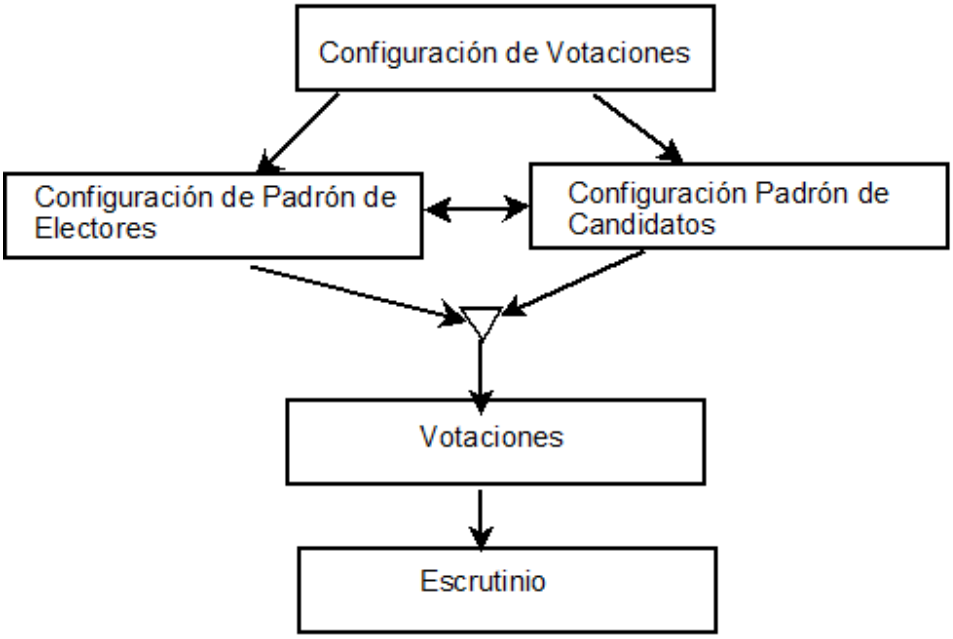
- *Comisión de Vigilancia: es un tipo de usuario que le da mayor confiabilidad al proceso, pues es en forma independiente al administrador, asegura que los parámetros de la votación están de acuerdo con la convocatoria de la misma antes de ser confirmados, registra su clave pública, puede consultar las bitácoras, firma todos los archivos claves de la votación que incluyen el archivo de configuración de la misma, y el padrón definitivo de electores y candidatos. Finalmente realiza el proceso de conteo de votos, descifrando la tabla de códigos aleatorios de candidatos, realizando la segunda ronda de*

descifrado de votos de la "urna", y generando y firmando el documento de resultados definitivos.

- *Electores / Candidatos: corresponde a aquellos que accederán a la votación creada por el administrador ejerciendo el rol de elector o candidato, por lo que podrá verificar si es elegible como candidato o elector y hacer reclamo de caso de alguna inconformidad, rechazar/aceptar/cancelar una candidatura que se le haya propuesto, en el caso de aceptar candidatura registrar su logo y url donde publicará información más detallada, podrá consultar el estado del proceso de votación en cualquier momento, si es elegible como elector podrá ejercer su derecho de voto y verificar los resultados finales de la elección.*

6.1.2. Fases

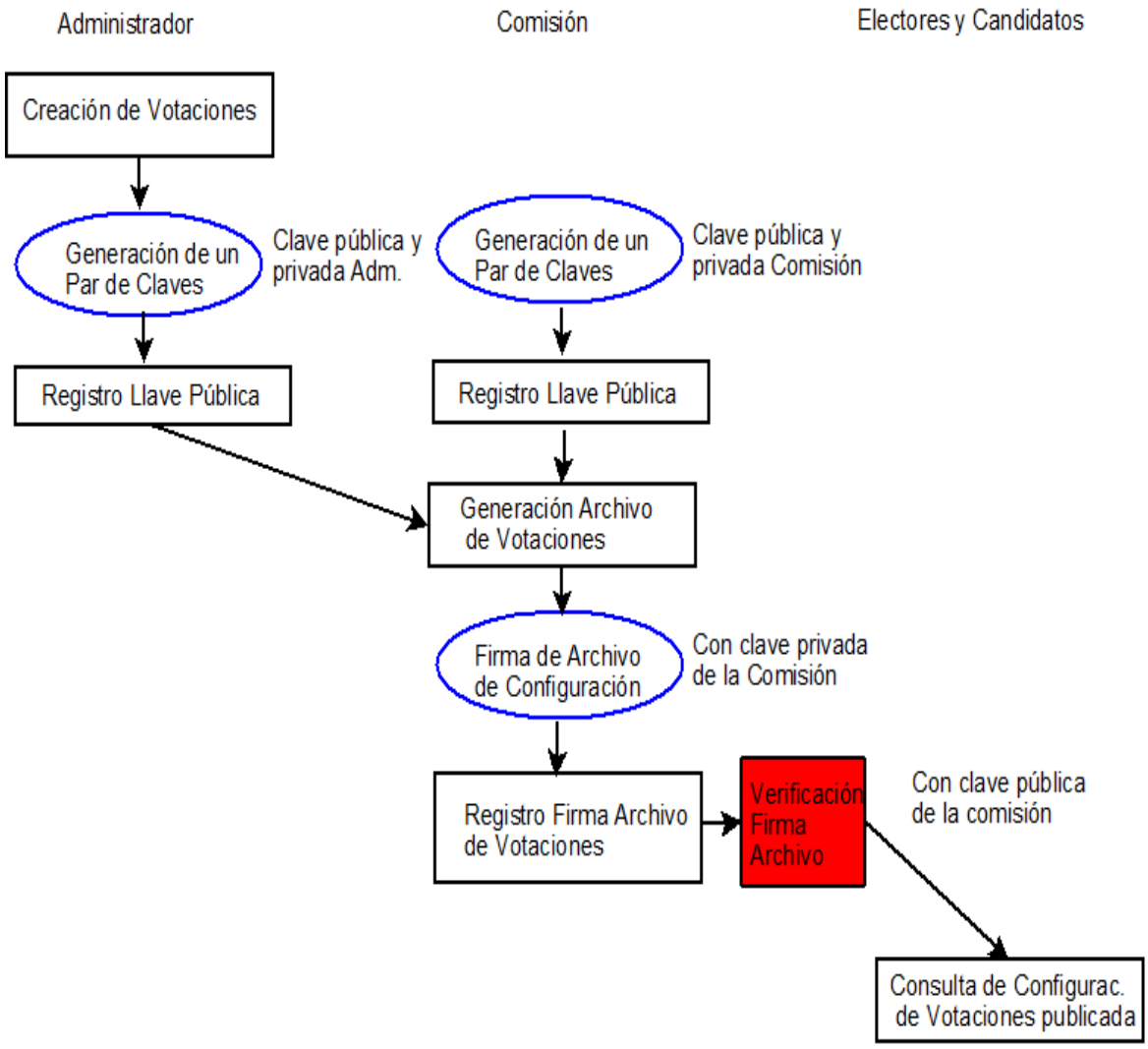
En el diagrama de bloques que se presenta a continuación se especifican las fases generales del proceso de votación.



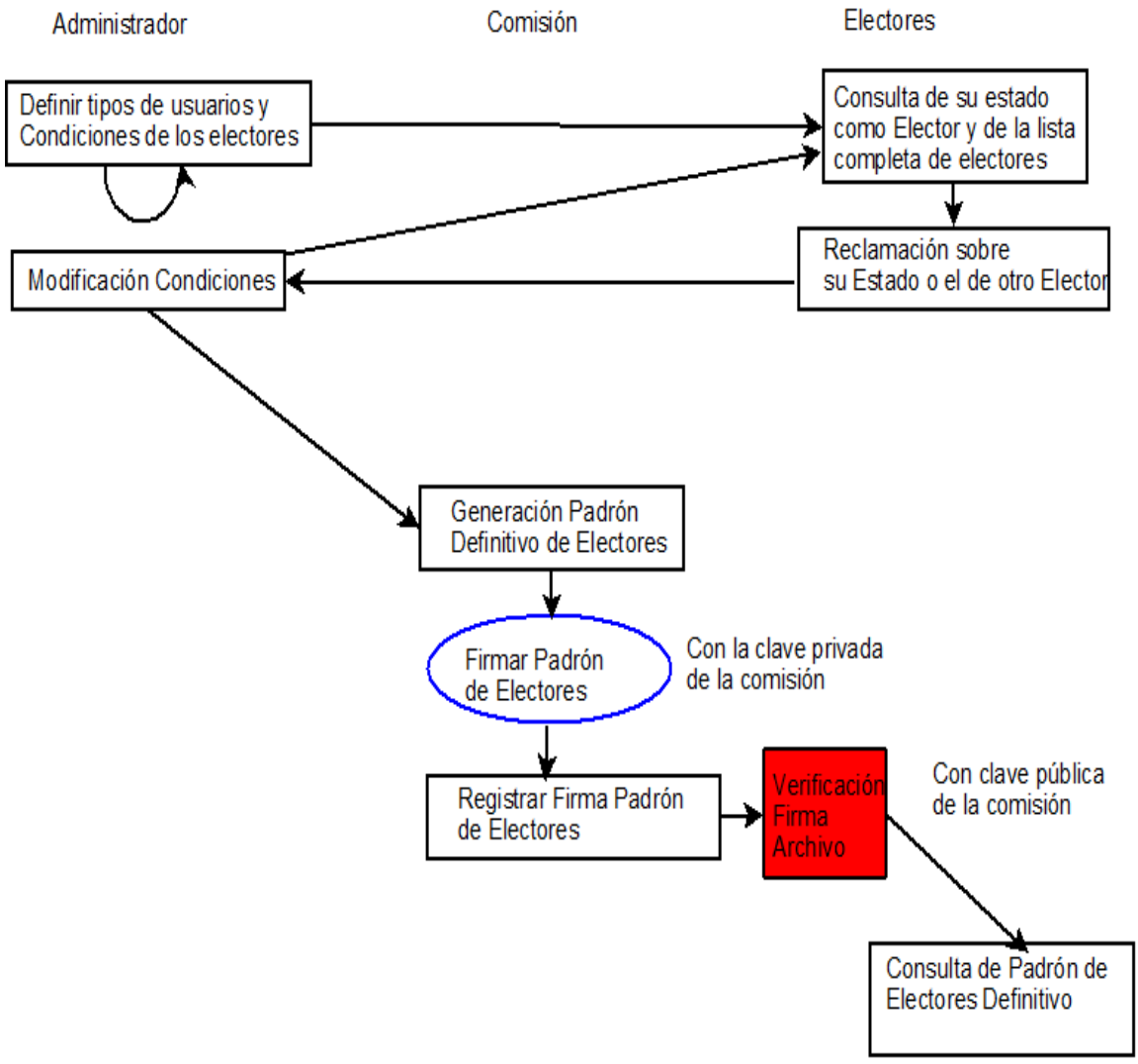
A continuación se presentan los diagramas detallados de cada una las fases principales del proceso, en cada uno se especifica quién participa en las mismas (Administrador, Comisión, Electores y Candidatos) y la plataforma sobre la que

están implementadas, así: las actividades realizadas en Plone se representan con un rectángulo y las que se soportan externamente en GnuPG con una elipse. De otro las actividades realizadas por el administrador o la comisión de vigilancia que usan funciones criptográficas se representan con una elipse, e incluyen un comentario que explica el tipo de clave o función utilizada. Finalmente se colorean con relleno las actividades realizadas internamente por el sistema Plone y que usan funciones criptográficas.

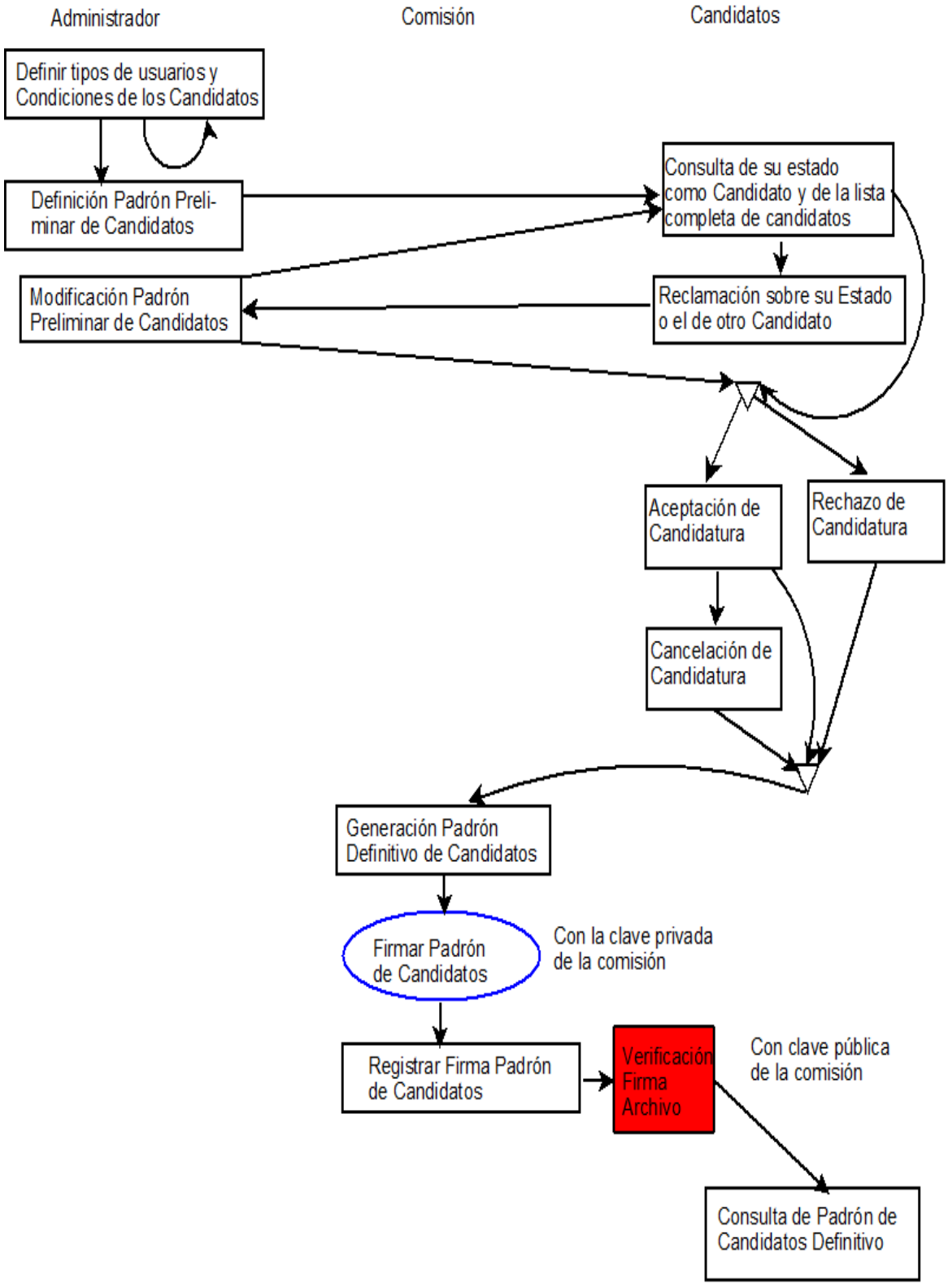
Fase I: Configuración de la Votación



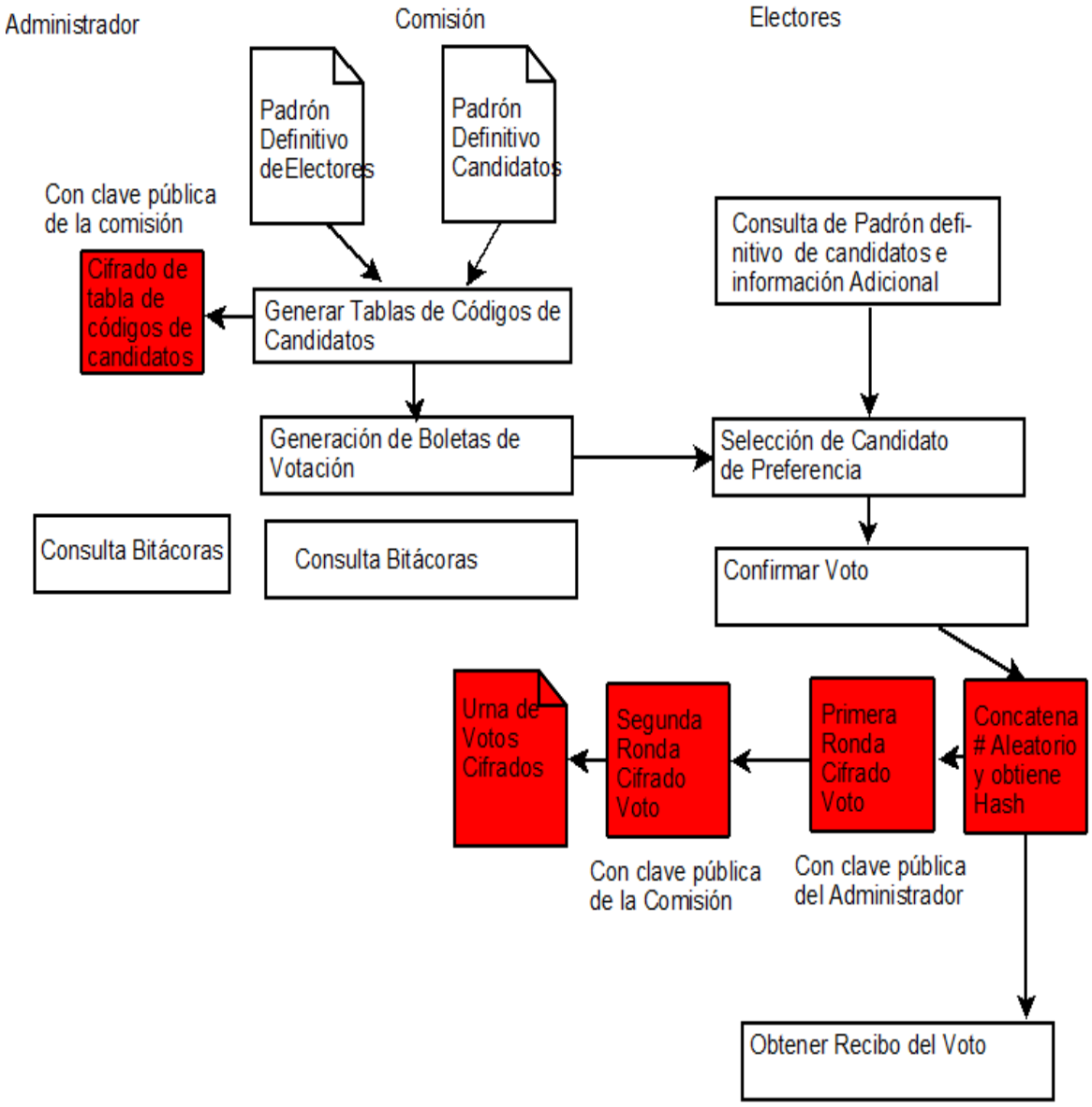
Fase II: Generación del Padrón de Electores



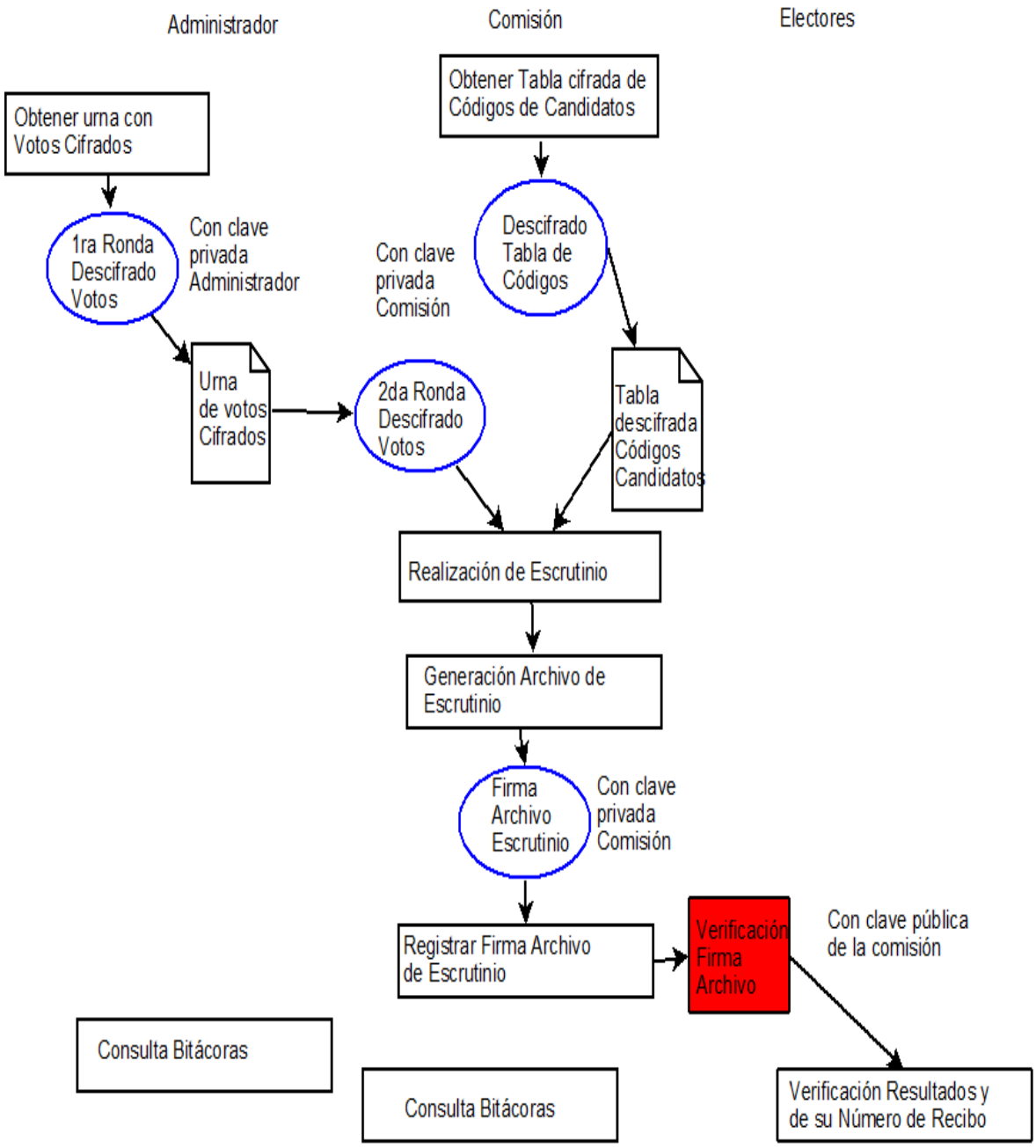
Fase III: Generación del Padrón de Candidatos



Fase IV: Votaciones



Fase V: Escrutinio



6.1.3. Bitácoras

En el proceso de votaciones se manejan tres tipos de bitácoras, las cuales son claves para la confiabilidad del proceso, pues en ellas se registran todas las actividades claves realizadas por el administrador, la comisión de vigilancia y los usuarios del

proceso. Estas bitácoras pueden ser accedidas tanto por el administrador, como por la comisión de vigilancia:

- *Histórico de Electores: en esta bitácora se registran todos los eventos o actividades realizadas para generar el padrón definitivo de electores, que van desde la definición de los tipos de usuarios que podrán votar y las condiciones que deben cumplir, los ajustes realizados por reclamaciones de usuarios ,hasta la generación del padrón definitivo y su respectiva firma y publicación.*

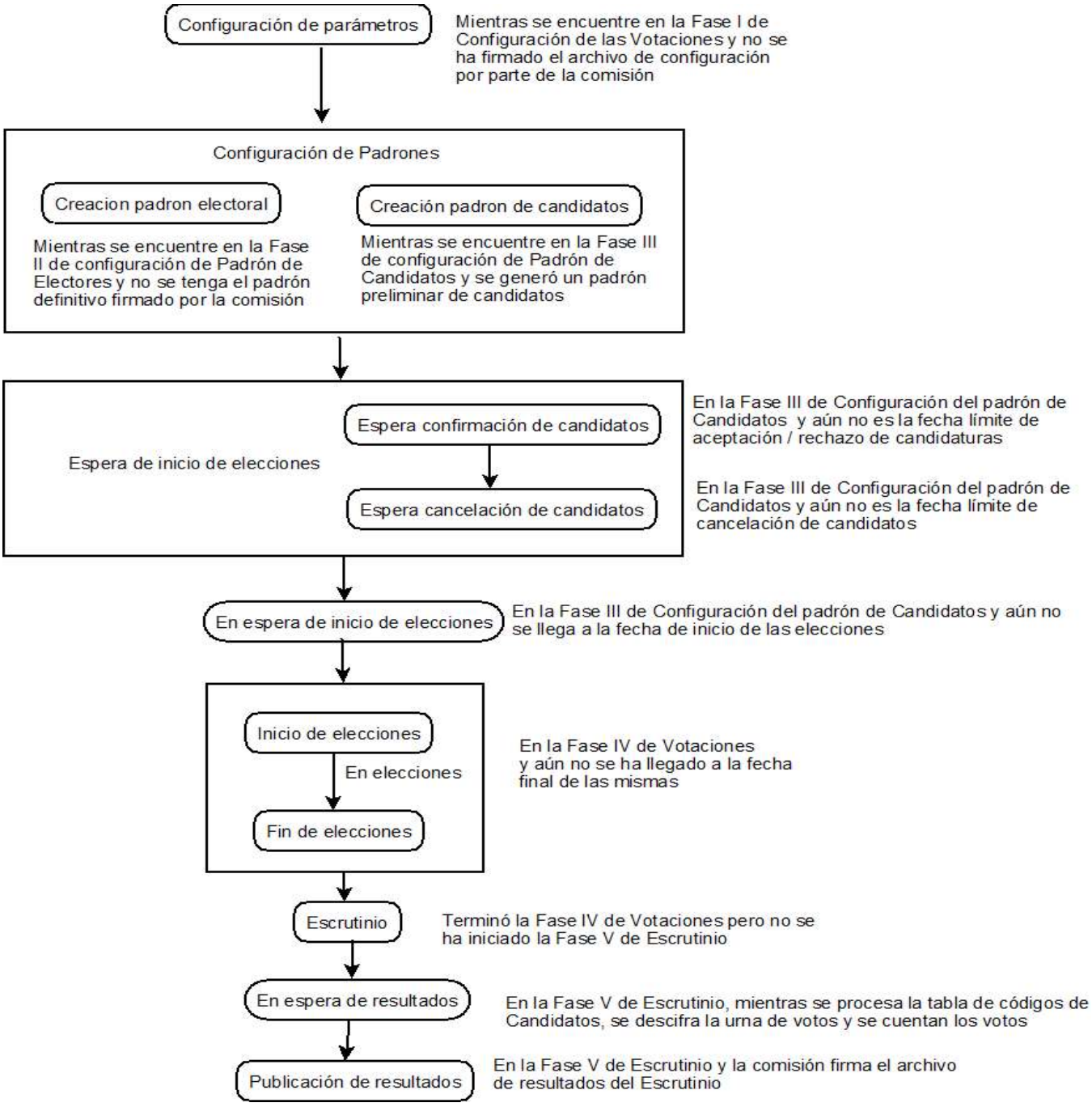
- *Histórico de Candidatos: en esta bitácora se registran todos los eventos o actividades realizadas para generar el padrón definitivo de candidatos, que van desde la definición de los tipos de usuarios que podrán ser elegibles como candidatos y las condiciones que deben cumplir, la generación de un padrón preliminar, los ajustes realizados por reclamaciones de usuarios, la aceptación, rechazo o cancelación de candidaturas, hasta la generación del padrón definitivo y su respectiva firma y publicación.*

- *Histórico de Elecciones: en esta bitácora se registran todos los eventos desde la creación del objeto de la votación la configuración y ajuste de cualquier parámetro, la generación del archivo de configuración de votaciones, su correspondiente firma y publicación, la generación de los padrones definitivos de electores y candidatos, la creación del evento de votaciones, la generación de la tabla de códigos de candidatos y de las boletas de votación, el inicio del proceso de votaciones, el registro de los usuarios que votan, la terminación de las votaciones, el reordenamiento aleatorio de la "urna" con los votos, la realización del proceso de escrutinio incluyendo el descifrado de la tabla de códigos de candidatos y de la "urna" con los votos, el conteo de los votos y la publicación y firma de los resultados finales.*

6.1.4. Estados

El proceso de votaciones tiene los siguientes estados, los cuales serán desplegados una vez el usuario ingrese al sistema y dependerán de la fase en que se encuentre el proceso.

Diagrama de Estados del Proceso



6.2. Productos en Plone

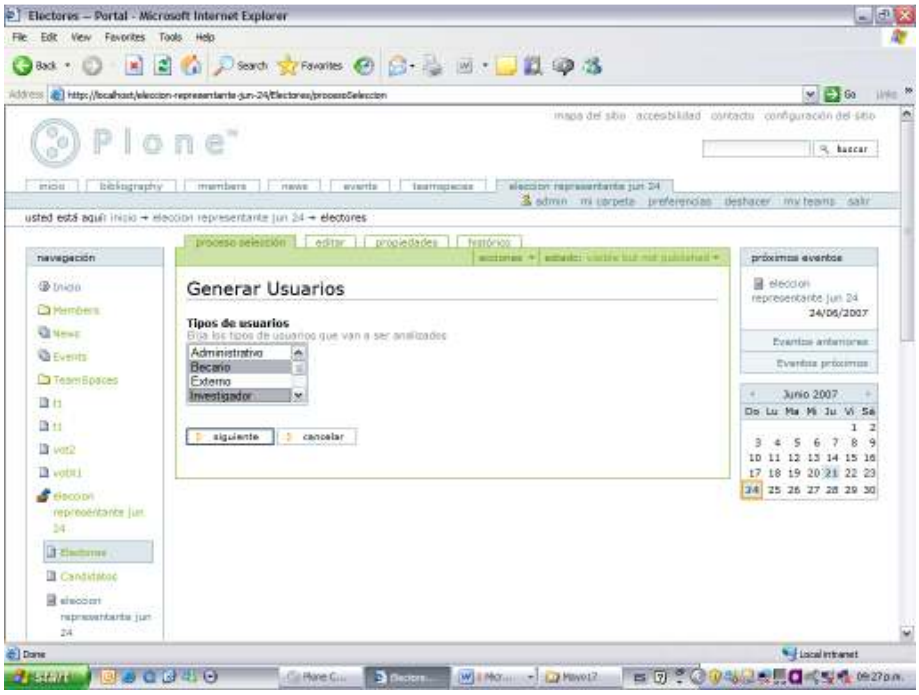
Se decidió implementar dos productos uno de selección de usuarios y otro de votaciones, con el fin de que el primero pueda ser utilizado en forma independiente del proceso de votaciones, pues se consideró que podría ser útil para procesos donde se requiera seleccionar un conjunto de usuarios de un sitio Plone que cumplan con un grupo de condiciones predeterminadas:

6.2.1. Producto de Selección de Usuarios

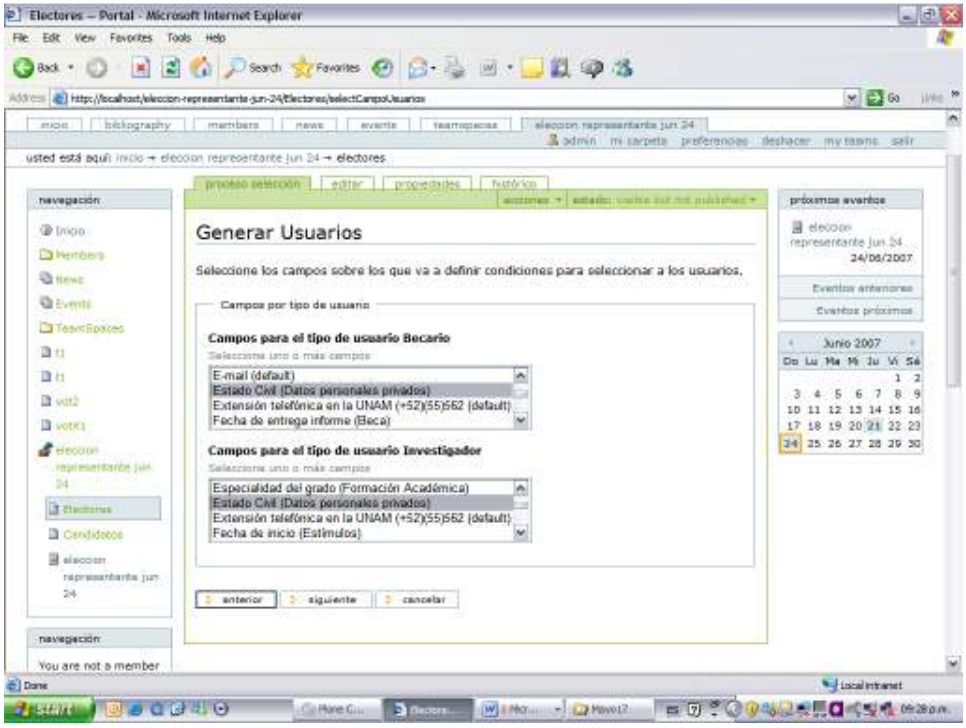
Es utilizado por el producto de votaciones para generar una lista con un subconjunto de miembros del sitio Plone que cumplen los criterios establecidos por el administrador de las votaciones y que es verificado por una comisión de vigilancia, en este producto primero se debe escoger el tipo de miembros que se quieren incluir en la selección, luego los campos que se van a utilizar para definir los criterios de selección, luego los criterios a cumplir en forma específica para cada uno de los campos utilizados. Una vez generada una lista de miembros se pueden hacer las modificaciones que se requieran, se pueden redefinir los criterios a partir de una nueva base o tomando los ya seleccionados antes. Este producto es usado para los procesos de generación del padrón de electores y el padrón de candidatos.

Lo primero que se debe realizar es agregar un ítem de tipo Selección Usuarios y después realizar los siguientes pasos:

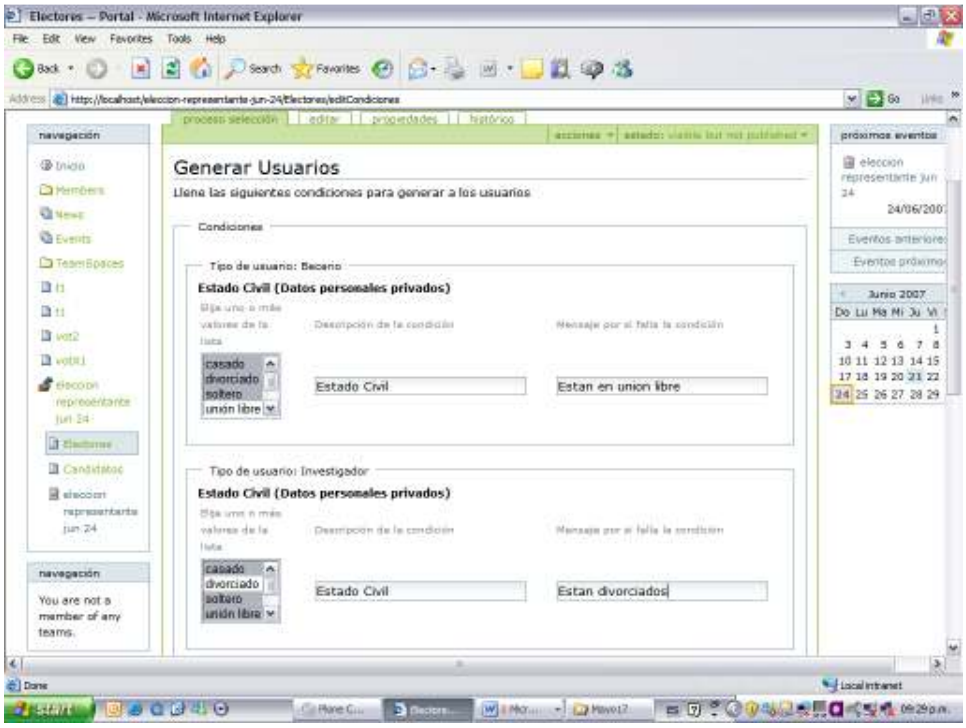
- *Definir los tipos de miembros del sitio Plone que se quieren seleccionar:*



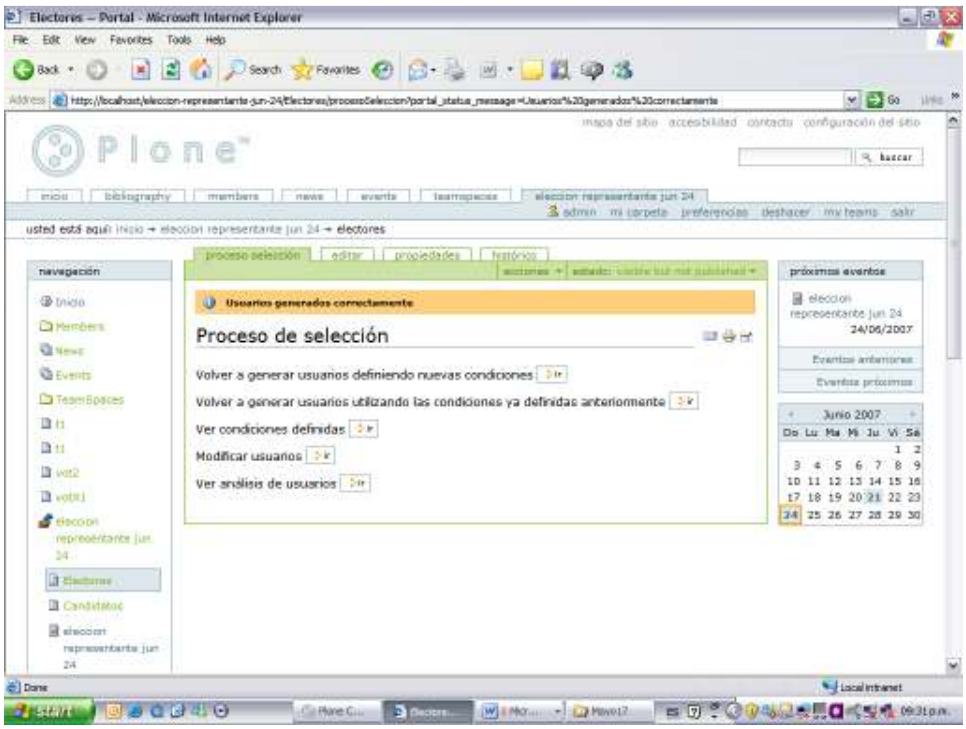
- *Por cada tipo de miembro seleccionado deben escogerse los campos requeridos para definir los criterios que deben cumplir:*



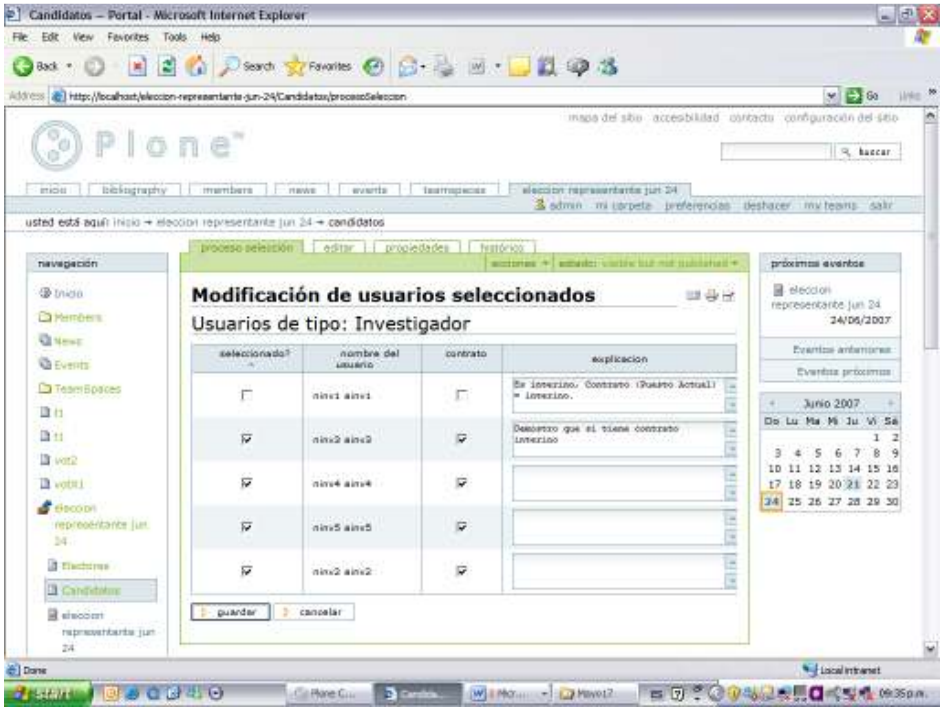
- *Por cada campo seleccionado se debe definir el criterio específico a cumplir:*



- *Finalmente puede ajustarse la lista de miembros seleccionados utilizándose las siguientes opciones:*



- *Por ejemplo en el caso de utilizar la opción de "Modificar usuarios", se pueden agregar usuarios no considerados inicialmente o descartar otros preseleccionados, especificando la explicación o justificación de esta acción.*



Este producto permite verificar todos los cambios realizados en la el proceso de selección de usuarios, incluyendo las modificaciones manuales a los registros seleccionados y la correspondiente explicación, mediante la opción "Ver Histórico sobre la selección de Usuarios":

Una vez se tenga la lista definitiva de usuarios seleccionados se puede generar un archivo en formato pdf.

Como se mencionó antes, el producto de selección soporta las fases de selección y convocatoria de candidatos y generación del padrón electoral, dentro del proceso de votaciones así:

Fase del proceso	***Funcionalidad del producto***
Selección y convocatoria de Candidatos	1. *Configuración de las reglas de selección de los posibles candidatos con base en ciertos valores de los campos de la base de datos, es importante mencionar que para darle flexibilidad al sistema cualquier tipo de miembro del sitio Plone puede ser elegible como candidato, con lo que podrán ser soportadas elecciones diferentes a la descrita en el sistema actual.* 2. *Ejecución de un proceso automático para marcar los posibles candidatos en la base de datos.* 3. *Atención de los reclamos y realización de los ajustes requeridos en forma manual sobre la lista base generada en el proceso automático.* 4. *Generación del padrón de candidatos preliminar.*

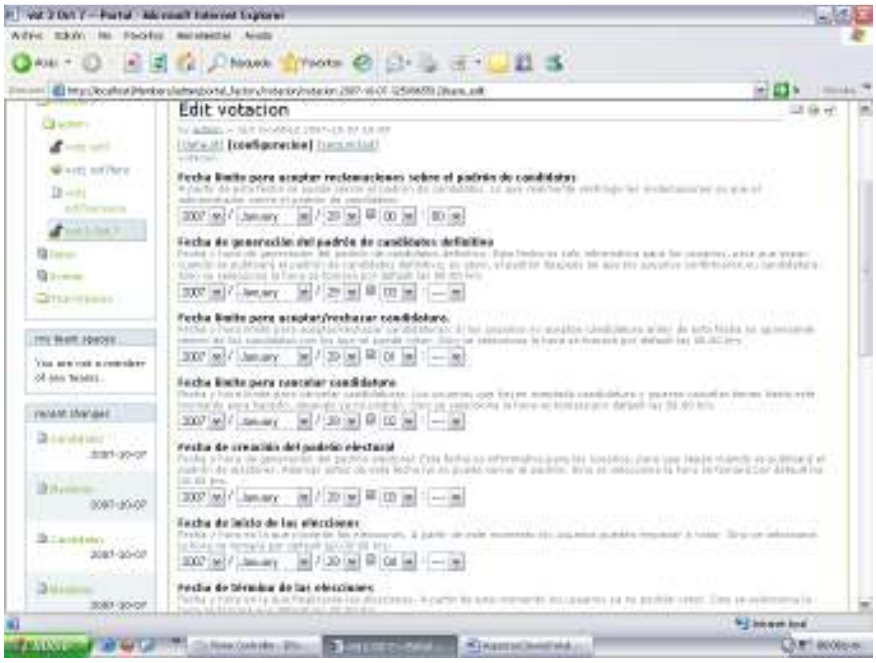
Fase del proceso	*Funcionalidad del producto*
	5. *Generación de una bitácora con todos los cambios realizados en el padrón de candidatos.* 6. *Generación del padrón definitivo de candidatos.*
Generación Padrón Electoral	1. *Configuración de las reglas de selección de posibles electores con base en ciertos valores de los campos de la base de datos, es importante mencionar que para darle flexibilidad al sistema cualquier tipo de miembro del sitio Plone puede ser elegible como elector, con lo que podrán ser soportadas elecciones diferentes a la descrita en el sistema actual.* 2. *Ejecución de un proceso automático para marcar los electores posibles.* 3. *Atención de los reclamos de los miembros no seleccionados como electores y considerarse ajustes en la base de datos en campos que afectan los criterios de selección de un posible elector en un rango de fechas establecido* 4. *Reprocesamientos en la generación de posibles electores.* 5. *Generación de una bitácora con todos los cambios realizados en el padrón de electores.* 6. *Generación del padrón de electores definitivo en una fecha determinada en la configuración del proceso de votaciones.*

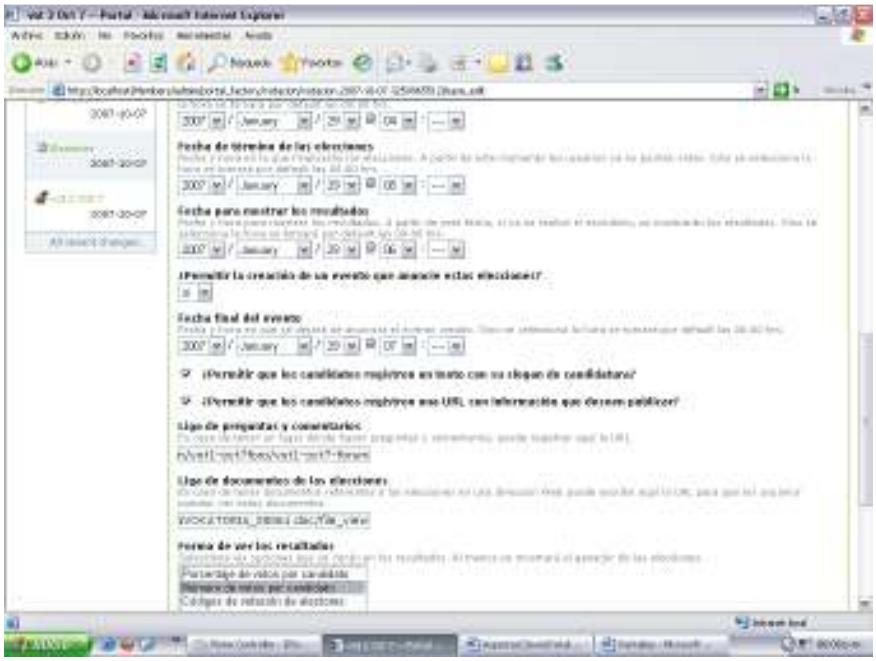
6.2.2. Producto de Votaciones

Este producto se utiliza para configurar un proceso de votaciones, el registro de votos y la realización del escrutinio final:

Una vez se halla instalado el producto de votaciones y el de selección de usuarios, lo primero que debe realizarse es agregar un objeto del tipo "votación" y definir su identificación, descripción y cuerpo.

Después de darle la opción "Siguiente" deben definirse las fechas claves del proceso de elecciones en las cuales se creará el padrón electoral, se generará el padrón de candidatos definitivo, finalizará el evento del proceso y otros parámetros generales:

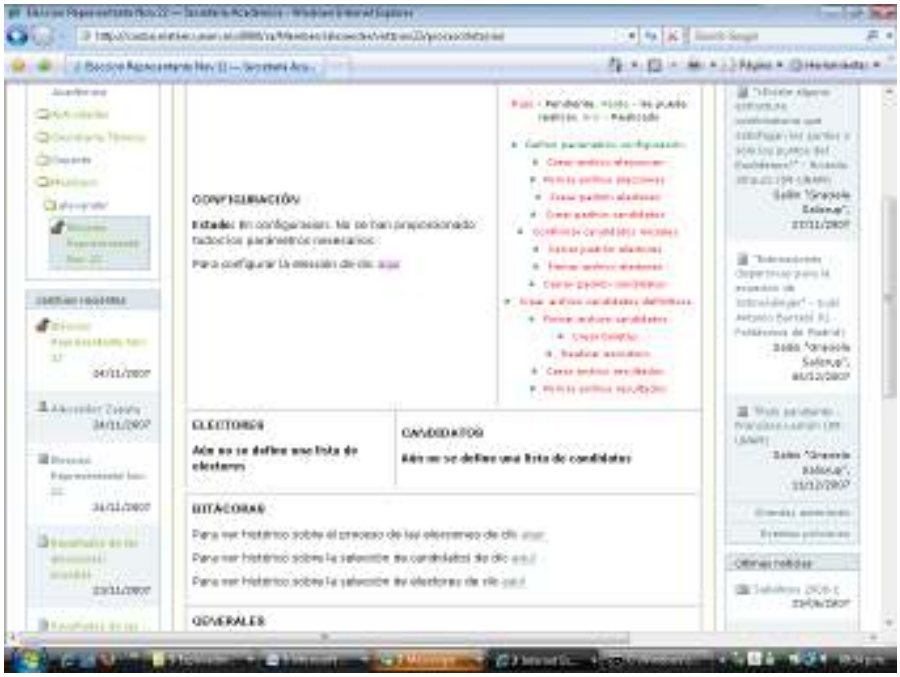




Al dar clic en el botón "siguiente" se debe registrar tanto la llave pública de la comisión de vigilancia que se tomará como base para verificar los documentos firmados (padrones, resultados y configuración de la votación) y cifrar los votos, como la llave pública del administrador de la votación que se utilizará para una ronda adicional de cifrado de los votos. Es importante mencionar que antes del proceso se debieron generar el par de claves privada y pública con Gnu PG por fuera del sitio Plone tanto para la comisión como para el administrador.

Todos los pdf generados (padrón de electores, padrón de candidatos y descripción de la votación, más adelante explicados) sólo se mostrarán cuando ya hayan sido firmados por la comisión de vigilancia.

Una vez se configuren los parámetros básicos de la votación se puede utilizar el botón "proceso de votación" para utilizar alguna de las siguientes opciones, las cuales se describen en los manuales del administrador (Apéndice I), la comisión de Vigilancia (Apéndice II) y de los usuarios finales (Apéndice III).



6.3. Protocolo de Votación

El protocolo implementado para el proceso de votación, considera algunas de las características de los dos esquemas analizados en el Estado del Arte Votaciones Suiza [16] y KOA [23] y cumple en una medida importante las condiciones requeridas según el análisis comparativo incluido también en ese apartado y el proceso de decisión mediante el método Delphi presentado en el capítulo anterior.

Tal como se especifica en el capítulo 2.4 de Conclusiones del Estado del Arte, el sistema implementado no cumple" con las siguientes características (se conserva su numeración de acuerdo con lo especificado en los capítulos 2.2 y 2.3) del sistema de Votaciones de Suiza y del KOA, algunas de las cuales se han establecido como posible trabajo futuro en el capítulo de Conclusiones:

Sistema de Votaciones de Suiza:

3) *Separación de la base de datos donde se encuentra la información de los votantes y la de los votos.*

5) *Mecanismos de protección contra ataques de suplantación tipo IP Spoofing.*

7) *Autenticación del servidor de votaciones mediante un certificado digital.*

8) *Mecanismos de protección contra ataques de Denegación de Servicio.*

10) *Uso de dos servidores en paralelo.*

KOA (Proceso):

2) *El votante inicia conexión vía HTTPs*

3) *Certificado digital del sitio web de votaciones.*

KOA (Características Claves):

7) *Huella dactilar electrónica de los candidatos y la lista de votantes.*

6.3.1. Descripción

El proceso es el siguiente (todos los eventos del mismo quedan registrados en la bitácora de las elecciones):

1) ***Identificación de los códigos de candidatos que registrarán los electores como votos***

Esquema más simple: asignar un código único a cada candidato, lo cual facilitaría el criptoanálisis por texto conocido, pues habría valores de votos que se repetirían cuando varios votantes hagan la misma elección.

Esquema implementado (acorde con el estado del arte propuesto en el sistema KOA): antes de iniciar el proceso de votación pero después de haber determinado los padrones definitivos de electores y de candidatos se generan un conjunto de números aleatorios únicos de tamaño configurable entre 0 y 10^n para los diferentes candidatos. La idea es que se generen tantos números aleatorios como votantes para cada uno de los candidatos. Por lo que en el caso de que por ejemplo hubiesen X candidatos y Y Electores válidos, por cada uno de los X candidatos se generarán Y números aleatorios, que no pueden repetirse ni en un mismo ni en diferentes candidatos, asegurando que dos votos no sean iguales y se dificulta la técnica de criptoanálisis por texto conocido, además si hay un proceso suplantación de elector al momento de votar se dificultaría escoger un código válido que corresponda a otro candidato pues son números aleatorios, además en el caso que alguien pudiera acceder a la tabla de códigos de candidatos y buscar un valor válido, al asignar exactamente tantos códigos como número de electores ayuda a determinar el caso de una suplantación al tratar de registrar dos votos con el mismo valor. Si por ejemplo tuviéramos un proceso de votaciones con 3 candidatos y 5 electores válidos, se podría generar una tabla de códigos asignados a los candidatos así (el tamaño real de los números aleatorios se configura como un parámetro de seguridad de la votación que se recomienda sea mayor o igual a 40 pues 10^{40} es un poco mayor que 2^{128}):

Candidato	***Num1***	***Num2***	***Num3***	***Num4***	***Num5***
Luis	*27*	*90*	*14*	*25*	*8*
María	*18*	*10*	*45*	*91*	*33*
Carlos	*15*	*2*	*99*	*85*	*72*

Esta tabla sólo puede ser descifrada por la comisión de vigilancia del proceso una vez termina el tiempo de la votación, pues está cifrada con la clave pública de la misma y sólo se habilita la opción de obtener el archivo al inicio del escrutinio.

2) Generación de Boletas de Votación

Esquema más simple: al asignar un código único a cada candidato, no se generarían boletas de votación y simplemente se desplegarían estos cuando el

usuario decida votar, esto facilitaría el ingreso de votos no reales, pues al no quedar asociado el voto a un elector por el criterio de privacidad no habría forma de identificar cuáles se ingresaron en forma no autorizada y cuales sí.

Esquema más seguro: corresponde al esquema implementado que se explica a continuación con la variación de que la boleta se envía por email [23] (sistema KOA) con todas las condiciones de seguridad requeridas.

Esquema implementado: se debe genera una boleta de votación para cada elector, la cual incluye los códigos asignados para cada candidato que podrá utilizar solamente ese elector al momento de que decida votar, la selección de los códigos para cada candidato es aleatoria y se asegura que ningún código se repita en dos boletas diferentes. Esta boleta de votación es creada como un dato privado que tiene asociado un método que garantiza la privacidad, de tal manera que el objeto sólo podrá ser visualizado por parte del elector al que se le asignó después de haberse autenticado en el sitio Plone y de seleccionar la opción de votar, por lo que no podrá ser visualizada por ningún usuario distinto, incluyendo el administrador de las votaciones y la comisión de vigilancia, y en ningún momento distinto al del evento de votar. Este esquema cumple en realidad todas las condiciones de seguridad del estado del arte, sólo que no depende de la seguridad y disponibilidad de servicios distintos como el de correo electrónico

En el ejemplo anterior una posibilidad de las 5 boletas de votación de cada uno de los electores podría ser la siguiente, en la cual se puede notar que la asignación de códigos de candidatos es aleatoria, pues por ejemplo al elector 1 le corresponde el segundo código posible del candidato Luis, el tercer código definido para el candidato María y el quinto código asignado al candidato Carlos; estos códigos de candidatos no podrán ser asignados a ningún otro elector. En el caso del elector 2 le corresponde el primer código del candidato Luis, el quinto del candidato María y el cuarto del candidato Carlos. Como se ve no se sigue ningún patrón reconocible en la asignación de códigos y podemos decir, que este método le da mayor aleatoriedad a la definición de las boletas de votación:

Elector 1 (E_1)

Luis	*90*
María	*45*
Carlos	*72*

Elector 2 (E_2)

Luis	*27*
María	*33*
Carlos	*85*

Elector 3 (E_3)

Luis	*8*
María	*18*
Carlos	*2*

Elector 4 (E_4)

Luis	*14*
María	*91*
Carlos	*15*

Elector 5 (E_5)

Luis	*25*
María	*10*
Carlos	*99*

3) Registro de Voto por parte de un elector autenticado

Esquema más simple: el usuario selecciona su candidato a elegir y se actualiza el contador de votos para ese candidato. En este caso no se maneja el concepto de base de datos de urna de votos, pero este esquema no garantiza Justicia pues se pueden conocer resultados parciales antes de terminar la votación.

Otra posibilidad, es guardar el voto en una urna pero sin encripción, lo cual no impide los conteos parciales del caso anterior. Una alternativa adicional un poco más segura es cifrar el voto con la clave pública de la comisión de vigilancia, pero estos podrían ponerse de acuerdo para leer o alterar los votos guardados.

Esquema más seguro: al voto se debe concatenar una cadena aleatorio de longitud fija y después cifrarse con una llave que representa un secreto compartido [37] entre el administrador de las elecciones y n-1 miembros de la comisión de vigilancia. Para poder realizar el conteo de los mismos se requerirán al menos k de los mismos para reconstruir el secreto compartido [37], donde k<n, y se evita el riesgo que si alguno de los involucrados no quiere aportar su parte del secreto el proceso quede bloqueado.

Esquema implementado: una vez un elector válido se autentique en el sitio Plone y decida registrar su voto, se activa un evento que presenta la boleta de votación, que en realidad no muestra los códigos de candidato asignados y sólo presenta sus nombres y fotos (en el caso que hayan sido guardadas).

En el momento que el elector seleccione el candidato por el cual quiere votar el sistema le pide confirmar su selección, por ejemplo si el Elector 1 (E_1) decide votar por María E_1(M) y confirma su voto, el voto en un inicio corresponde al número 45 (E_1(M)=45) .

A continuación se genera un número aleatorio de tamaño fijo, con tantos dígitos como el parámetro establecido en la configuración de la votación, y se le concatena al valor inicial del voto, quedando por ejemplo 4578, donde 78 corresponde al número aleatorio generado.

En este momento se le aplica una función hash MD5 al número conformado por el código del candidato seleccionado y el número aleatorio generado, en el caso del ejemplo sería Hash(4578) y el resultado corresponde al recibo del voto que se explica en el paso siguiente del protocolo.

A continuación se realiza un proceso de doble cifrado al número 4578, primero con la clave pública de la comisión de vigilancia y luego con la clave pública del administrador de las elecciones, éste valor final corresponderá al voto que se guarda en la base de datos (urna de votos doblemente cifrados). El resto de códigos de candidatos no seleccionados en la boleta del usuario se guardan para verificar que no sean utilizados en forma no autorizada como un voto fraudulento, en el caso del ejemplo presentado serían los códigos E_1(L)=90 y E_1(C)=72, los cuales no se repiten para ningún otro elector y no podrían corresponder a un voto válido. Este esquema es tan seguro como el estado del arte pero se requiere que tanto el administrador como la comisión estén presentes para descifrar los votos, por lo que se requieren los n participantes y no k<n como el esquema que utiliza el algoritmo de secreto compartido [37].

4) Recibo del voto para el elector

Esquema más simple: no generar ningún recibo de voto que le permita al usuario verificar que su voto fue debidamente contado en el escrutinio final. Un siguiente nivel sería la generación de un recibo de voto que aparezca al final del escrutinio pero no asociado a ningún candidato, de tal manera que el elector pueda verificar que su voto fue contabilizado pero no pueda saber si para el candidato de su preferencia.

Esquema implementado: tal como se explicó en el paso anterior del protocolo, antes de cifrar el voto, se le saca el hash, el cual corresponde al recibo del voto, éste número es incluido en un pdf que el elector podrá imprimir o guardar en su carpeta, incluyendo su identificador, la fecha y hora, y el título y descripción de la votación en la que está participando.

Este esquema garantiza la verificabilidad individual analizada en el estado del arte, pues el elector puede confirmar al final del escrutinio si su voto fue contado apropiadamente, pues para cada candidato se especifica la lista de los recibos de los votos que fueron tenidos en cuenta, dándole mucha confiabilidad en el proceso a los electores.

Lo único es que se contradice en parte el anonimato porque el usuario puede demostrar por quien votó, pues como se acaba de mencionar ese número de recibo aparece al final en la lista de votos detallada del candidato que seleccionó y si el elector lo imprimió al momento de votar, aparece su respectiva identificación, esto sucede sólo si es la voluntad del elector hacerlo público, pues él es único que puede acceder al recibo de su voto y cualquier otro participante en el proceso no puede conocer quién efectuó un voto a partir de un número de recibo, pues una función hash es unidireccional.

5) Escrutinio

Esquema más simple: lectura del contador de votos en caso de que no se guarden los votos. Otra opción es que en el caso de que los votos se guardaron en el orden en que llegaron, estos se podrían asociar al elector tomando el mismo orden de los eventos de voto en la bitácora de las elecciones, además en el caso de que se hayan alterado los votos o se ingresen nuevos se podrá detectar la inconsistencia al comparar el total con los eventos de la bitácora pero no se podrán determinar cuáles son los votos que deben ser anulados.

Esquema implementado: antes de que se inicie el proceso de escrutinio se reordena en forma aleatoria la base de datos de votos (urna de votos doblemente cifrados) para que no exista una forma de asociarlos a los electores de acuerdo con el orden de los eventos de votos registrados en la bitácora de las elecciones.

Se requiere que tanto el administrador como la comisión de vigilancia utilicen sus claves privadas, en ese mismo orden, para poder leer los votos de la base de datos ("abrir la urna sellada"), después de quitarles la cadena aleatoria de tamaño fijo que se le concatenó.

Posteriormente, se requiere que la comisión inicie el conteo de votos por cada candidato, pues se toma cada voto descifrado y se le agrega al candidato que tenga dicho número asignado, éste número se marca en el objeto de códigos de candidatos como ya utilizado, en el caso que se repita se anula el voto anterior, el nuevo no es contado y se agrega al objeto de votos anulados con la descripción de "Voto Doble".

En el caso que se registre un voto con un código de candidato no asignado o que corresponda a una boleta cuyo voto por otro candidato fue ya contado, se desplegarán en el escrutinio como "Voto Inválido". Es importante aclarar, que la posibilidad de que se ingrese en forma no autorizada un voto doble o un voto inválido es baja debido a que la urna de los votos es un objeto privado que sólo puede ser accedido por la función de votar, la cual sólo se le presenta a los usuarios debidamente autenticados, que no hayan votado y sólo durante el

periodo de la votación, además se encuentra doblemente cifrados, lo que dificulta la realización de una modificación íntegra de un voto espurio que se pudiese tomar como válido.

Al finalizar este proceso se tiene el número total de votos por cada candidato con el detalle de los mismos y se compara con el de eventos contados de la bitácora, cualquier diferencia debe quedar registrada para ser analizada por la comisión de vigilancia.

El paso final del escrutinio corresponde a la generación, firma por parte de la comisión de vigilancia y publicación de un archivo pdf con los nombres de los candidatos, el número de votos, incluyendo datos de identificación de la votación, fecha y hora, los votos anulados con su descripción (doble/inválido), y el detalle de los números de hash (recibos de voto) contados para cada uno de los candidato

Una vez los resultados de la votación han sido publicados, los electores podrán consultar la lista de los candidatos con su número total de votos y el detalle de los números de recibo contados para cada uno, de tal manera que podrá verificar que su voto fue efectivamente considerado para el candidato que seleccionó.

6.3.2. Requerimientos del esquema propuesto

A continuación se sustenta el cumplimiento de las condiciones analizadas en el estado del arte de los sistemas de votación electrónica por parte del sistema implementado en Plone:

Esquema/Requerimiento	*Eleg*	*Priv*	*Verf*	*Exac*	*Just*
EVotaMatem	*Ok*	*Parc*	*Ind*	*ok*	*ok*

***no** =no se cumple, **ok** = se cumple, **max**=máximo, **ind**=individual, **con**=condicional*

- *Eligibilidad (Eleg): **OK***
    - *Una vez se configure la elección de acuerdo con las reglas establecidas en la convocatoria de la misma, las cuales especifican las condiciones que deben cumplir los posibles electores, se publican listas preliminares, para que estos puedan realizar cualquier tipo de reclamo en el caso de que no sean considerados en las mismas, a pesar de cumplir todas las condiciones necesarias. Después de que se llegue a la fecha de*

generación del padrón definitivo de electores, la comisión de vigilancia lo revisará y firmará en el caso de estar de acuerdo. Después de esto, sólo los usuarios que se encuentren dentro del padrón definitivo de electores podrán votar.

- *Privacidad (Priv):* ***PARCIALMENTE***

    - *El esquema es anónimo, pues no es posible que el administrador de la votación, comisión de vigilancia o cualquier otro participante en el proceso pueda averiguar el voto de algún votante, pues cada boleta de votación sólo puede ser visualizada por el usuario al cual se le generó y sólo en el momento de votar. De otro lado, al ingresar el voto no queda información del usuario y la "urna" con los votos es reordenada en forma aleatoria antes del escrutinio, con el objetivo de que el orden de sus registros no corresponda al mismo que la bitácora de las elecciones y se pueda realizar la asociación de un voto a un elector.*

    - *Al agregar a los votos (número aleatorio de candidato seleccionado) una cadena aleatoria adicional y cifrar doblemente tanto con la clave pública del administrador como con la de la comisión de vigilancia, se dificulta el criptoanálisis del mismo por parte de un atacante para poder encontrar el valor del voto y además dicho valor no puede ser asociado directamente a un candidato, pues se debería tener acceso a la tabla de códigos aleatorios por candidato, la cual está cifrada con la clave pública de la comisión de la vigilancia y es la única que podría conocerla.*

    - *La privacidad no se cumple en forma total, pues al generarse un recibo del voto que le permitirá al elector confirmar el apropiado conteo del mismo dentro del escrutinio de la votación, éste podría demostrar ante algún tercero por quien votó en el caso de aquí lo desease, pues en el escrutinio salen los números de recibo por candidato y en el recibo que sólo posee el elector aparece su identificación, pudiendo demostrar que él votó por un candidato específico.*

- *Verificabilidad (Verf):* ***INDIVIDUAL***

    - *El esquema de votación es verificable individualmente, pues cada votante puede comprobar mediante el número recibo que su correspondiente voto ha sido realmente incluido en el conteo final del candidato de su selección, consultando el detalle publicado al final del escrutinio de los números de*

recibo de dicho candidato.

- *Exactitud (Exac):* ***OK***
    - *El esquema de votación es exacto pues no resulta posible ingresar un voto no válido, pues después de descifrarlo se verifica que corresponde a un código válido de candidato y no está incluido en una boleta de votación ya utilizado, esto se logra así:*
        - *El que las boletas de votación queden protegidas para que sólo el usuario correspondiente las pueda ver en el momento de votación, asegura que sólo el usuario debidamente autenticado pueda seleccionar un número de candidato válido*
        - *Al guardar cifrado con la clave pública de la comisión de vigilancia el archivo de códigos aleatorios de candidatos dificulta a un atacante o al mismo administrador el seleccionar un código válido de candidato que pueda contar como voto y no sea anulado.*
        - *Al agregar a los votos (número aleatorio de candidato seleccionado) una cadena aleatoria adicional y cifrar doblemente tanto con la clave pública del administrador como con la de la comisión de vigilancia dificulta el criptoanálisis del mismo por parte de un atacante para encontrar las claves y registrar votos no válidos.*
- *Justicia (Just):* ***OK***
    - *Se logra una elección imparcial pues nadie puede obtener resultados intermedios antes de que terminen las elecciones, pues se requeriría la colusión del administrador de la votación y la comisión de vigilancia, pues los votos están doblemente cifrados con las claves públicas de estas entidades.*

Finalmente, se sustenta el cumplimiento de un importante número de los requerimientos de los sistemas de votación electrónica obtenidos de la aplicación de método Delphi con expertos en seguridad:

- *Confiable*
    - ***SI*** *- El sistema está en capacidad de mantener persistencia y garantizar las propiedades de las transacciones ACID.*

- ***NO*** *- Mejoramiento Futuro - El archivo de bitácora de las elecciones no está cifrado*.

- ***SI*** *- Se cuenta con una bitácora de transacciones que incluye el registro de la fecha y hora de cada evento.*

- ***NO*** *– Mejoramiento Futuro - La continuidad del proceso debe garantizarse, incluyendo las opciones para gestionar los fallos, técnicos, administrativos, procedimentales o humanos. Llevar actas de resolución de incidentes y establecer el proceso de cómo deben ser atendidas, cuándo y a quién escalar la resolución del incidente*

- *Unicidad*

  - ***SI*** *- Además de la bitácora el sistema cuenta con controles automáticos para evitar la votación múltiple, pues los códigos aleatorios que se establecen para cada uno de los candidatos no se repiten y cada número sólo puede ser contado una vez en el escrutinio, en caso contrario tanto el voto original como el segundo intento con el mismo número es anulado y reportado al final del escrutinio.*

  - ***SI*** *– Una vez un usuario autenticado y validado registre su voto, el sistema deshabilita el evento de votar para ese usuario.*

  - ***SI*** *– En la base de datos de electores se usa una bandera que indica si ya votó o no, ésta es inicializada con una marca de que no ha votado*

- *Integridad*

  - ***SI*** *– El archivo de bitácoras de la elección se cifra con la clave pública de la comisión de vigilancia.*

  - ***NO*** *– Por factores de funcionalidad y facilidad de uso no se manejan transacciones firmadas digitalmente y hashes periódicos de los votos en el tiempo, pues los proceso de cifrado y firma se realizan por gnuPG y requeriría que los usuarios manejaran bien este software.*

  - ***SI*** *- Además de detectar, se previene toda modificación no autorizada, mediante los códigos aleatorios asignados a los candidatos, la doble encripción de los votos y la exigencia de registro de votos sólo por usuarios debidamente autenticados y validados.*

- ***SI*** *- Los votos son contados correctamente y sólo por parte de la comisión de vigilancia, pues es la única que puede conocer los códigos aleatorios asignados a los candidatos, el conteo de votos es comparado con el número de eventos de votos de la correspondiente bitácora de la elección.*

- ***SI*** *– se cuenta con una bitácora de auditoría detallada de la aplicación.*

- *Exactitud*

  - ***SI*** *– Los votos son cifrados doblemente, con la clave pública del administrador de las votaciones y de la comisión de vigilancia.*

  - ***SI*** *- No se asocia la identificación del elector con el candidato seleccionado, pues en el voto sólo va el código del candidato y una cadena aleatoria que se le agrega.*

  - ***NO*** *- Se debería de llevar más de una bitácora, bitácoras locales a los sistemas de votación, mas una centralizada con hashes periódicos de los votos realizados.*

  - ***SI*** *– Se evitar la existencia de una secuencia de votantes en una bitácora que rompa la confidencialidad del voto, pues la base de datos (urna) se reordena antes del escrutinio para evitar cualquier asociación.*

- *Verificable.*

  - ***NO*** *– Mejoramiento Futuro - Los votos deberían ser registrados simultáneamente en dos sistemas independientes que no se comuniquen entre sí. Al*

  - *final del proceso ambos deben generar los mismos resultados. (se considerará en versiones posteriores).*

  - ***SI*** *– Se tiene diseñado un mecanismo de auditoría alrededor de los eventos de la elección, además la urna funciona como una caja negra pues los votos son doblemente cifrados con la clave pública del administrador de la elección y la comisión de vigilancia.*

  - ***PARCIAL*** *– Mejoramiento Futuro - Llevar un serial de cada transacción, tal como se hace cuando se generan certificados por una autoridad certificadora. El sistema genera un recibo del voto que corresponde al hash del voto + cadena aleatoria doblemente encriptado con la clave pública de la comisión de*

vigilancia y el administrador de las votaciones que fungiría como un identificador de la transacción del voto para el usuario.

- *Secreto y no coercitivo*
    - ***SI** – El sistema garantiza el anonimato a la hora de votar lo que es clave para la aceptación del sistema, esto se logra, separando el voto del elector, reordenando base de datos de votos y asegurando que las boletas de votación sólo puedan ser vista por el usuario que le corresponde en el momento de la votación.*
    - ***SI** - Las bitácoras se cifran.*
    - ***NO** – Mejoramiento Futuro - En caso de interrupciones en medio de la emisión de un sufragio, y restablecer o poder revertir votaciones parciales. En este momento se base en las características de Plone respecto a la garantía de integridad de las transacciones.*
    - ***PARCIAL** - El sistema de autenticación de usuarios y la base de datos está desligado de la base de datos de votos, pues se manejan en componentes diferentes de la base de datos de Zope, pero debería buscarse la manera de distribuir esto entre diferentes servidores.*
- *Elegibilidad y Autenticación*
    - ***SI** – Todas las personas en capacidad de votar, no quedan excluidas por problemas de contraseñas, pues se utiliza el mismo sistema de autenticación del sistema Plone y no se requiere enviar PINs adicionales que puedan perderse o no ser recibidos por los electores potenciales.*
    - ***NO** – Mejoramiento Futuro – además de la Contraseña no se utiliza "otro elemento" que posibilite una autenticación doble.*
    - ***NO** - Mejoramiento Futuro – Firma Digital es un concepto que se puede considerar en un futuro y requeriría la participación de una autoridad certificadora o algún mecanismo para generar confianza en las claves públicas de los electores. En la versión actual podría manejarse con el gnuPG pero le daría mucha complejidad al uso del sistema y rechazo respecto a su utilización.*
    - ***NO** - Mejoramiento Futuro – Una contraseña puede ser*

crackeable, sería más confiable en un futuro utilizar un medio biométrico u otro sistema más robusto.

- ***SI***- *debido a que en la información del voto no se guarda información del votante y la base de datos de votos se reordena se dificulta ligar a un voto con el usuario autenticado.*

- ***SI*** - *La construcción del padrón de electores es robusto, pues se definen los electores elegibles mediante criterios, posteriormente se pueden aceptar reclamos y realizar los ajustes respectivos, para que al final se generé un padrón definitivo que debe ser firmado por la comisión de vigilancia después de su correspondiente verificación.*

- *Escalabilidad*

  - ***NO*** - *Mejoramiento Futuro – Analizar la disponibilidad de comunicaciones en sitios remotos y verificar la no degradación de la capacidad de procesamiento por el incremento de transacciones.*

  - ***NO*** - *Mejoramiento Futuro – Se puede manejar en varias capas de acuerdo al sistema, pero básicamente se debe contar con un sistema de balanceo en el front-end.*

- *Velocidad*

  - ***NO*** - *Mejoramiento Futuro – - El sistema opera en "tiempo-real" o sea que la velocidad debe asegurar que no se produzcan colas de votantes. Actualmente depende del desempeño del servidor Plone, el cual es razonable para procesos de elección pequeños y medianos.*

  - ***PARCIAL*** - *Hasta medio minuto es tolerable, lo que en un sistema bien hecho y correctamente dimensionado es muchísimo tiempo. Aplica la misma justificación del punto anterior.*

  - ***NO*** - *Mejoramiento Futuro – Este requerimiento puede estar muy relacionado con otros factores exógenos como distribución de electores, horas pico, distribución geográfica y otros que vale la pena considerar en conjunto.*

- *Conveniencia*

  - ***PARCIAL*** - *Mejoramiento Futuro – Es clave, ya que personas mayores "evitan" o tienen una resistencia al cambio fuerte. En*

este caso se usa una plataforma de internet con buena facilidad de uso, pero no se ha hecho un trabajo para refinar este aspecto en la implementación.

- ***PARCIAL** - Mejoramiento Futuro – Se maneja una interfaz estándar en el proceso, de acuerdo con los demás sistemas de información del sitio Plone, pero no se ha realizado un trabajo de refinación a este respecto.*

- ***NO** - Mejoramiento Futuro – desarrollar procedimientos para las personas no videntes o con otras discapacidades motoras o de otra índole.*

- *Políticas de Notificación*

  - ***SI** - Los usuarios conocen con antelación su situación para poder votar de modo que existe seguridad sobre la correcta elaboración del padrón electoral, también conoce los requisitos para poder votar, esto se logra con información en línea cuando ingresa al sitio Plone y mediante documentos oficiales con esta información que firma la comisión de vigilancia.*

  - ***SI** - El votante tiene la facilidad de poder consultar su estado en todo momento.*

- *Soporte del voto*

  - ***NO** - Mejoramiento Futuro – Se debe tener cuidado con la técnica de ingeniería social Shoulder Surfing.*

  - ***SI** - Al usuario se le asegura la integridad e intención final de su voto mediante el recibo del voto que puede confirmar al final del escrutinio,*

  - ***SI** - Al igual que cuando hacemos una transferencia electrónica de fondos y la máquina nos pregunta "¿está seguro de hacer esta transacción…?", cuando el usuario registra su voto se pide confirmación por parte del mismo.*

6.4. Requerimientos para la instalación del producto de votaciones

El requerimiento básico para poder instalar este producto es contar con el administrador de contenido Plone y la base de datos Zope. El proceso de instalación no se incluye en este documento.

Además debe contarse con los siguientes programas o herramientas de software:

Latex : *Miktex para Windows o Tetex para Linux, software utilizado para generar los archivos oficiales de la votación en formato pdf. Su directorio de instalación debe agregarse a la variable del sistema PATH. En el caso de Windows debe ejecutarse antes un index con el comando texhash en el subdirectorio bin.*

Python 2.5.1: *se utiliza para ejecutar el script descifrador de Votos con el objetivo de que sólo se requiera digitar una vez la contraseña asociada a la clave privada para descifrar los votos.*

gnuPG*: Software utilizado para todo el manejo criptográfico por fuera del producto de Plone, que incluye generación de un par de claves privada y pública, encripción de votos, desencripción de votos, firma de documentos y verificación de firma de documentos.*

Su directorio de instalación debe agregarse a la variable del sistema PATH.

De otro lado, para complementar la funcionalidad del producto de votaciones se requiere la instalación de los siguientes productos:

- *Ploneboard: producto requerido para la creación de foros, cuya funcionalidad que es utilizada para que los usuarios puedan enviar preguntas, comentarios o reclamos al administrador del proceso de votaciones, en el caso que tengan dudas sobre el mismo, o sobre su status como posibles electores o candidatos. Dentro de la configuración del objeto de votaciones se puede configurar si se va a utilizar o no esta funcionalidad, en el caso que así sea, se debe incluir la URL respectiva, la cual les aparecerá a los usuarios y al administrador cuando ingresen al objeto de la votación.*

- *MasterSelectWidget: el cual es utilizado por el producto de selección de usuarios con el fin de referenciar diferentes tipos de widgets.*

- *CMFMember: producto utilizado para agregar información adicional a los miembros del sitio Plone, incluyendo datos como tipo de usuario (investigador, becario, externo, etc) y otros atributos. En sitio del IMATE se configuró un producto que es una extensión de CMFMember que se denomina ImatemUser, el cual requiere de productos como AddRemoveWidget y ATCountryWidget.*

Desde el punto de vista de seguridad del servidor y la red deben considerarse los siguientes requerimientos:

- *El servidor debe contar con un certificado digital válido que le permita a la comisión de vigilancia, los electores y los candidatos verificar la*

autenticidad del sitio web de las votaciones al cual se están conectando y evitar ataques de suplantación de IP (IP Spoofing)

- *La comunicación de los usuarios con el servidor debe basarse en un protocolo tipo SSL/TLS que permita que la información viaje en forma cifrada, incluyendo el usuario/contraseña con el que se autentican.*

CAPITULO 7

7. Conclusiones

Para finalizar este documento se presentan las conclusiones del trabajo realizado, las cuales resumen las lecciones aprendidas del mismo, y los posibles trabajos a futuro para continuar con el mejoramiento de la solución implementada para el Instituto de Matemáticas, la cual como se comentó antes puede ser considerada en otras instituciones y organismos de diferente nivel de complejidad, claro está, sólo en el caso de que aspectos sociales, técnicos y premisas especiales sean atendidas, pues estos son factores críticos de éxito en los procesos de votación electrónica.

7.1. Conclusiones Generales y Lecciones Aprendidas

La utilización de la solución de votación electrónica, implementada en este proyecto, en el proceso de elección de la comisión dictaminadora en diciembre de 2007 en el Instituto de Matemáticas, fue una prueba de fuego respecto al cumplimiento de los requerimientos claves del sistema que se han discutido en varios capítulos de esta tesis:

- *Aceptación por parte de usuarios que siempre habían participado en procesos de elección con urnas físicas en lugares específicos, al contar con casi 50 votos. Lo que indica que el sistema es relativamente fácil de usar.*
- *No se presentaron reclamos respecto a los resultados, pues todos los votantes tuvieron la posibilidad de verificar que su voto fue considerado apropiadamente en el conteo final, mediante los números de recibo de voto.*
- *La comisión de vigilancia contó con los mecanismos necesarios para confirmar los parámetros de configuración de la votación, el padrón definitivo de electores, el padrón de candidatos y los resultados definitivos de la votación.*

La solución de votaciones electrónicas implementada tiene una funcionalidad amigable y completa para ser utilizado en procesos de elecciones de pequeña y mediana escala, donde los aspectos más relevantes sean los de exactitud, elegibilidad, justicia, verificabilidad y sobre todo transparencia y confianza por parte de los usuarios en el proceso y los resultados del mismo.

Es importante mencionar que toda la implementación se basó en el esquema

más simple en lo que respecta al registro un solo voto para alguno de los candidatos del proceso, y el extenderlo a permitir que un elector vote por diferentes candidatos al mismo tiempo, sería un trabajo futuro que exigiría la revisión y ajuste del protocolo de votación, de tal manera que se asegure que no se afecte el cumplimiento de los requerimientos de los sistemas de votación del estado del arte, ya mencionados previamente.

Al soportarse la implementación de todos los aspectos criptográfico relacionados con el protocolo durante la votación y el escrutinio en una herramienta ampliamente usada como es Gnu PG, la cual ha sido suficientemente evaluada por el público en general, se logra un mayor nivel de confianza en el Sistema. Esto le da mayor flexibilidad al sistema implementado, pues se podrían cambiar las funciones criptográficas utilizadas sin mayor impacto.

La implementación del producto sobre Plone fue uno de los aspectos más complejos del proyecto, requiriendo un porcentaje muy importante del tiempo en el análisis y comprensión de las características del mismo, no obstante el resultado es muy satisfactorio pues se logró construir un Producto fácilmente instalable, transportable y configurable de acuerdo con la necesidad de diferentes públicos usuarios del administrador de contenidos Plone.

Uno de los aspectos más relevantes de la solución implementada es que se integró a InfoMatem con una gran facilidad de utilización para los usuarios que participan como electores o candidatos, por lo cual basta con que los usuarios se autentiquen como regularmente lo hacen, accedan al evento donde se publiquen las votaciones, seleccionen al candidato de su predilección, confirmen su voto, impriman un recibo de su voto y verifiquen una vez que se publiquen los resultados que su voto fue apropiadamente contado (esto fue probado en el proceso de elección de la comisión dictaminadora en diciembre de 2007). En el caso del administrador de las votaciones y del organismo de control o comisión de vigilancia el proceso es un poco más engorroso, pues además de configurar la votación, establecer el padrón de electores y candidatos, monitorear el proceso y realizar el conteo de los votos, deberán utilizar funciones externas con GnuPG para generar un par de claves pública y privada, firmar documentos oficiales, descifrar la tabla de códigos de candidatos y la urna con los votos, claro está que todos estos aspectos se explican en forma detallada en los manuales de ayuda diseñados para apoyar el buen uso del sistema.

Finalmente es importante mencionar, que la solución implementada puede adaptarse a ajustes en el protocolo de votación pues está construida sobre código abierto, claro está que si se requeriría modificar el código del producto, pues en realidad no se tienen configurados diferentes esquemas de votación que puedan ser utilizados de acuerdo a los parámetros que establezca el administrador.

7.2. Trabajos a Futuro

El sistema de votaciones electrónicas implementado puede seguir un proceso de optimización en varios aspectos, de los cuales resaltan los siguientes:

Fortalecimiento del mismo para la resistencia de diferentes tipos de ataques de Denegación de Servicio (DoS), incluyendo algunas de las propiedades del sistema SVE de DGSCA explicado en el estado del arte:

- *Arquitectura de alta disponibilidad y diseño basada en la lógica de uso con protección contra ataques informáticos.*
- *Integración de mecanismos de protección de datos, monitoreo y detección de actividades de intrusos al Sistema.*
- *Hosting del sitio, para garantizar un ininterrumpido proceso lectoral y un nivel de seguridad máximo.*
- *Detección de intrusos que analice de acuerdo con las reglas del comportamiento informático una posibilidad de ataque o sabotaje a la infraestructura de cómputo o al proceso como tal.*
- *Replicación de datos y mecanismos de redundancia y procedimientos*

Análisis de estrategias para la distribución de la base de datos y la capacidad de procesamiento del mismo, para soportar diferentes esquemas de operación distribuida.

Mejoramiento en el protocolo de votación y escrutinio con el fin de minimizar la posibilidad de que el votante demuestre ante terceros (posible coerción) por quién realizó su voto, pero garantizando al tiempo el principio de verificabilidad universal.

Cifrado de la bitácora del proceso de votaciones de tal manera que sólo puede ser accedida por la comisión de vigilancia, además ésta debe guardarse en un servidor diferente al del registro de los votos. Por otra parte, establecimiento de rutinas de verificación periódica de su integridad que aseguren que ningún evento es eliminado.

Uso de Firma Digital con la participación de una autoridad certificadora o algún mecanismo para generar confianza en las claves públicas de los electores y del mismo servidor de votaciones.

Manejo de un esquema de autenticación fuerte que utilice un mecanismo adicional al de "Algo que se sabe" como es el caso de una contraseña, con "Algo que se tiene" o "Algo que se es", que corresponde a un tipo biométrico.

Parametrización de diferentes protocolos de votación que puedan usarse de acuerdo a configuración que establezca el administrador de las votaciones.

Registro de un número variable de votos por cada elector.

Diseño de un módulo de calificación de la votación que facilite el análisis de inconsistencias en el proceso por parte de la comisión de vigilancia, además de presentar los votos inválidos y repetidos que la solución actual incluye.

Manejo de un esquema de secreto compartido [37] para la definición de la contraseña que protege la clave privada de la comisión de vigilancia, de tal manera que se requiera un conjunto de miembros mínimo para poder realizar los procesos de firma y descifrado, pero no necesariamente deben participar todos.

Manejo de información relacionada con los partidos participantes y aseguramiento de la participación de un miembro de cada uno en la comisión de vigilancia.

Utilización de métodos seguros y criptología para el desarrollo de software como el recomendado en INST SP800-64.

Utilización de UML Seguro desde el diseño del software de tal manera que se puedan incorporar formalmente diferentes enfoques de control de acceso como RBAC y MAC, así como chequeos de integridad.

Considerar todos los aspectos requeridos para cumplir con los principios de "Seguridad en el Ciclo de Vida del Software" [40], de tal manera que se asegure:

- *Confiabilidad: no existen vulnerabilidades maliciosas o no intencionales explotables.*
- *Ejecución predecible: confianza justificable de que el software funcionará como debe en el momento de ejecutarse.*
- *Conformidad: conjunto de actividades que aseguran que el software cumple con los requerimientos, estándares y procedimientos aplicables.*

Apéndice I – Manual Administrador

Funcionalidad Incluida

1) *Configuración proceso de votación y registro de su clave pública.*
2) *Generación y administración del padrón de electores*
3) *Generación y administración del padrón de Candidatos*
4) *Creación del evento y foro de preguntas de la votación*
5) *Despliegue en forma continua del estado actual del proceso de votaciones.*
6) *Escrutinio de Votos*
7) *Consulta de bitácoras de actualizaciones en el proceso de votación, electores y candidatos.*

Requerimientos mínimos:

El sistema puede funcionar en equipos con sistema operativo Linux, Windows o Mac.

Con el fin de asegurar el apropiado funcionamiento del sistema y evitar que se generen errores en el proceso se requiere tener instalados las siguientes herramientas:

- *Gnu PG para el proceso de firma digital y su correspondiente verificación. Debe descargarse la versión correspondiente al sistema operativo de su máquina. (Linux, Windows o Mac)*
- *Python 2.5.1 para poder ejecutar el script de desencripción de votos con sólo digitar una vez la contraseña.*

Tipos de usuarios en el sistema:

En la versión actual del sistema existen tres tipos de usuarios:

- *Administrador de la votación: corresponde al usuario del grupo de Managers que agrega el objeto tipo votación, configura todos los parámetros y fechas claves de la misma, registra su clave pública, crea el evento y el foro de preguntas y respuestas de la votación, genera y administra los padrones de electores y candidatos de acuerdo con la convocatoria de la elección; también se encarga de la verificación de bitácoras y apoyar parte del conteo de los votos en lo que respecta a la primera de dos rondas de descifrado de la "urna" de votos.*

- *Comisión de Vigilancia: es un tipo de usuario que le da mayor confiabilidad al proceso, pues es en forma independiente al administrador, asegura que los parámetros de la votación están de acuerdo con la convocatoria de la misma, registra su clave pública, firma todos los archivos claves de la votación que incluyen el archivo de configuración de la misma, y el padrón definitivo de electores y candidatos. Finalmente realiza el proceso de conteo de votos, descifrando la tabla de códigos aleatorios de candidatos, realizando la segunda ronda de descifrado de la "urna" de votos, y generando y firmando el documento de resultados definitivos.*

- *Usuario general: corresponde a aquellos que accederán a la votación creada por el administrador ejerciendo el rol de elector o candidato, por lo que podrá verificar si es elegible como candido o elector y hacer reclamo de caso de alguna inconformidad, rechazar/aceptar/cancelar una candidatura que se le haya propuesto, en el caso de aceptar candidatura registrar su logo y url donde publicará información más detallada, podrá consultar el estado del proceso de votación en cualquier momento, si es elegible como elector podrá ejercer su derecho de voto y verificar los resultados finales de la elección.*

Productos incluidos

El sistema EVotaMatem está conformado por dos productos que deben ser instalados, votaciones y selección de usuarios:

1) *Selección de usuarios: producto que es utilizado por el producto de votaciones para generar una lista con un subconjunto de miembros del sitio plone que cumplen los criterios establecidos por el dueño del objeto, en este producto primero deben seleccionar el tipo de miembros, luego los campos que se van a utilizar, luego los criterios a cumplir. Una vez generada una lista de miembros se pueden hacer las modificaciones que se requieran, se pueden redefinir los criterios a partir de una nueva base o tomando los ya seleccionados antes. Este producto es usado para los procesos de generación del padrón de electores y el padrón de candidatos.*

2) *Votaciones: producto clave del sistema que incluye toda la configuración del proceso y ajustes al mismo, registro y consulta de información de bitácoras, control de aceptación, rechazo y cancelación de candidaturas, el registro de votos y conteo, entre otros.*

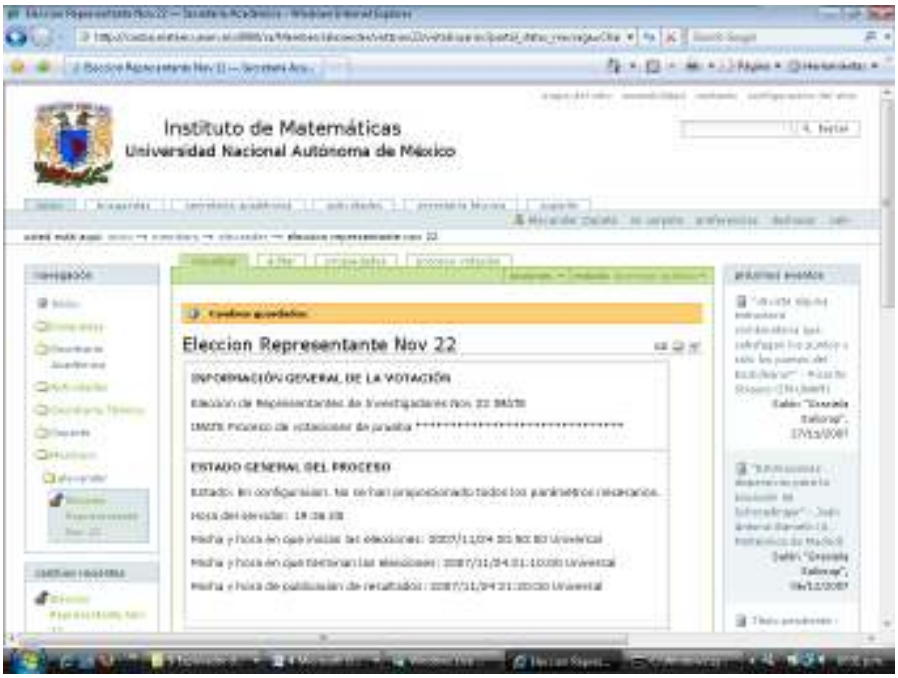
Creación de un objeto tipo votación

Una vez se haya instalado el producto de votaciones y el de selección de usuarios, lo primero que debe realizarse es agregar un objeto del tipo "votación" y definir su identificación, descripción y cuerpo, además de especificar el nombre del usuario que va a tener la función de Comisión de Vigilancia del proceso.

Proceso de Configuración básica por parte del dueño de la votación

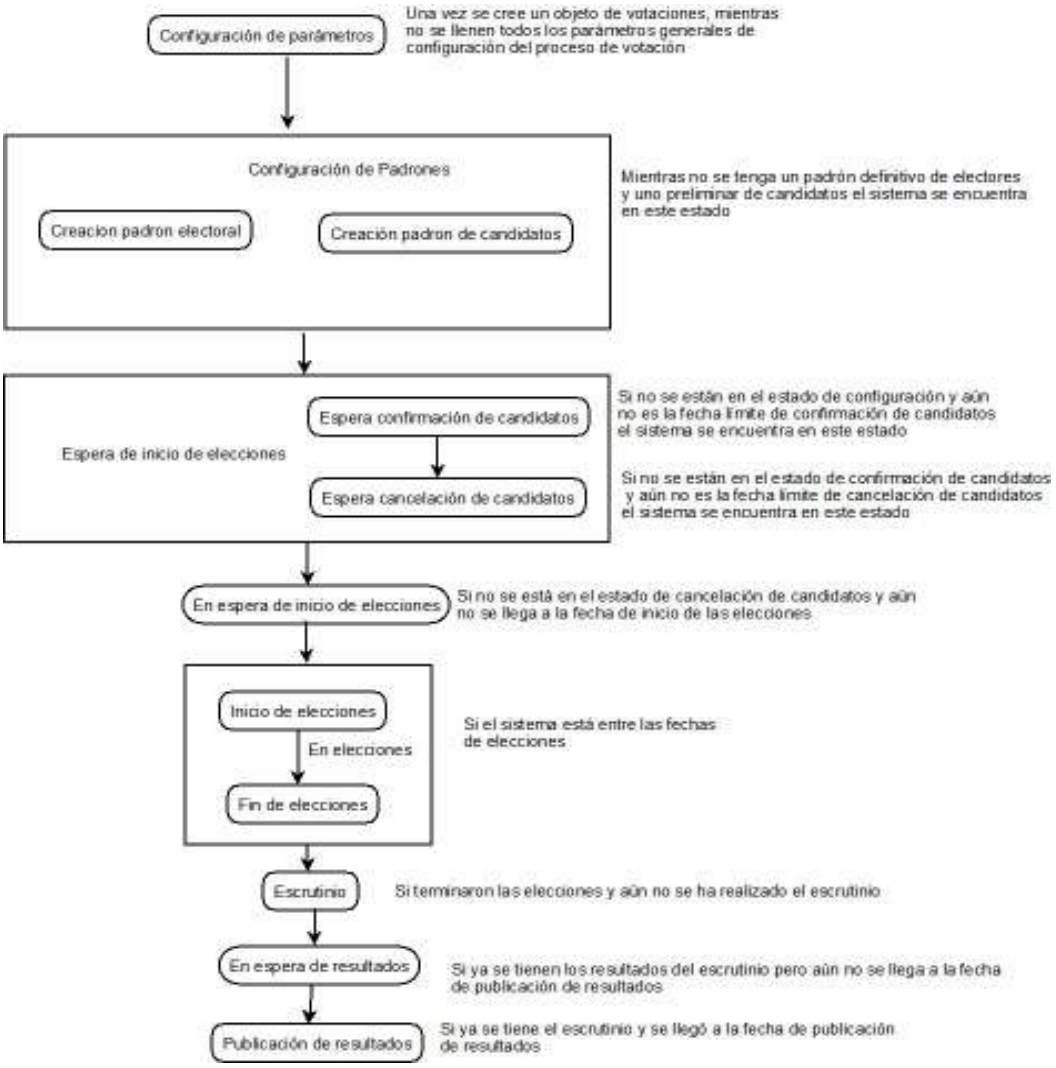
1. Verificación del estado general del proceso

Lo primero que se recomienda hacer es consultar el estado general del proceso, mediante la opción de Visualizar:



En esta pantalla se puede verificar el estado actual de la votación, el cual podría ser uno de los que se presenta en el diagrama de la página siguiente.

Se puede también verificar la hora actual del servidor del sitio plone para determinar el tiempo que resta para el inicio y la culminación de la votación, para presentar los resultados, y para los tiempos límites en algunas partes del proceso relacionadas con aceptación, rechazo o cancelación de candidaturas y publicación de padrones que se visualizan en el Estado como Elector y Estado como Candidato (estos datos se desplegarán una vez se hayan configurado todas las fechas del proceso).



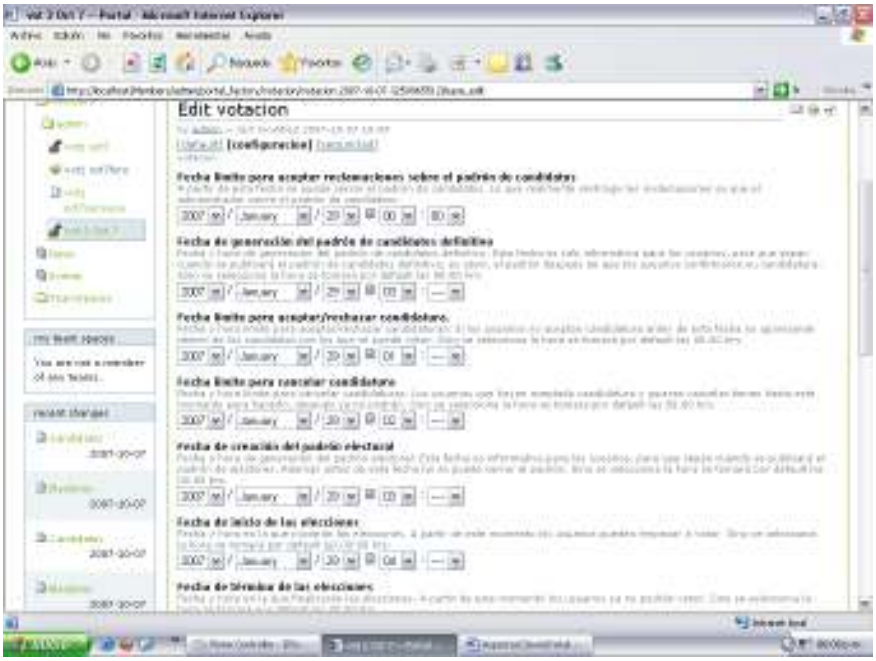
De otro lado, para facilitar la realización de las diferentes actividades del proceso se incluye una pantalla que puede ser visualizada en cualquier momento mediante la opción de "proceso de votación" , en la cual se presenta la lista de actividades del

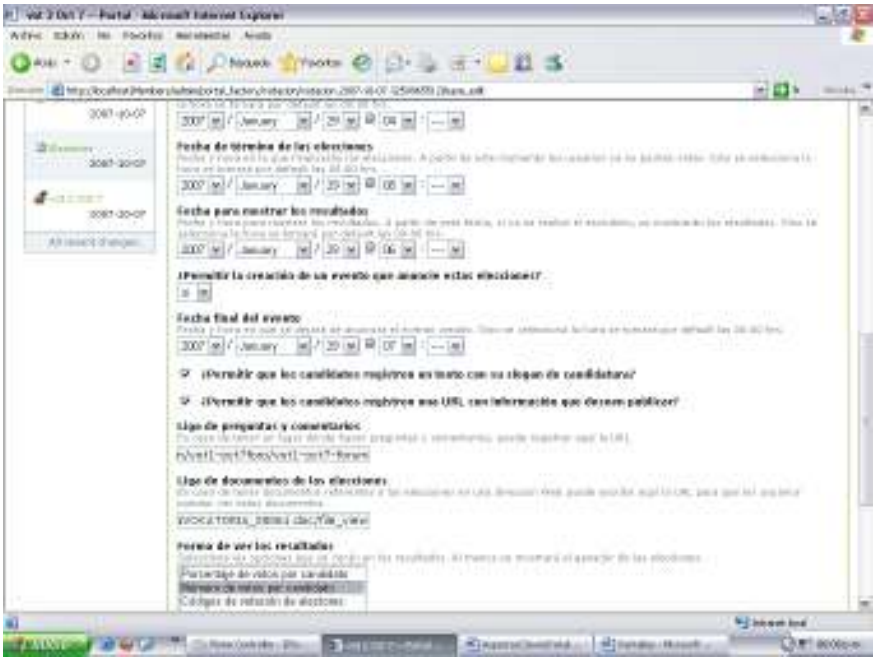
proceso, presentando en color gris las ya realizadas, en color verde las que pueden ser realizadas y en color rojo las que se encuentran pendientes pero que aún no pueden ser realizadas porque el estado del proceso no lo permite aún.

2. Configuración de las fechas de la votación

Después de darle la opción "Siguiente", la configuración de fechas también puede realizarse desde la opción Proceso de Votación, dando clic en Para configurar la elección de clic aquí.

Deben definirse las fechas claves del proceso de elecciones en las cuales se creará el padrón electoral, se generará el padrón de candidatos definitivo, finalizará el evento del proceso y otros parámetros generales, las cuales deben tener un orden lógico como el que se presenta en el campo "hora" de las dos pantallas siguientes:



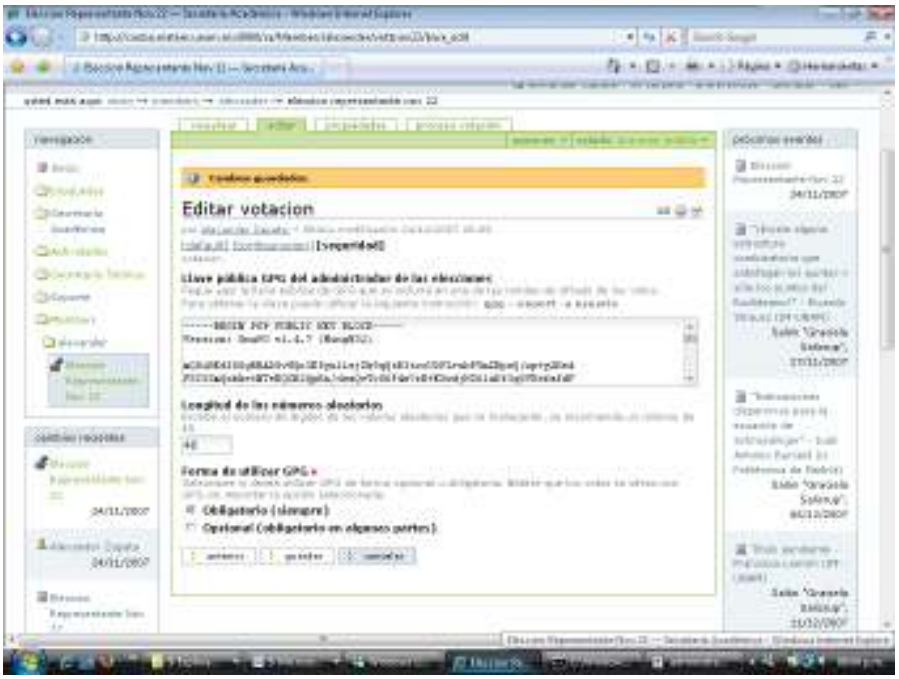


Al dar clic en el botón "siguiente" se debe registrar la llave pública del administrador de la votación que se tomará como base para una ronda de cifrado de los votos.

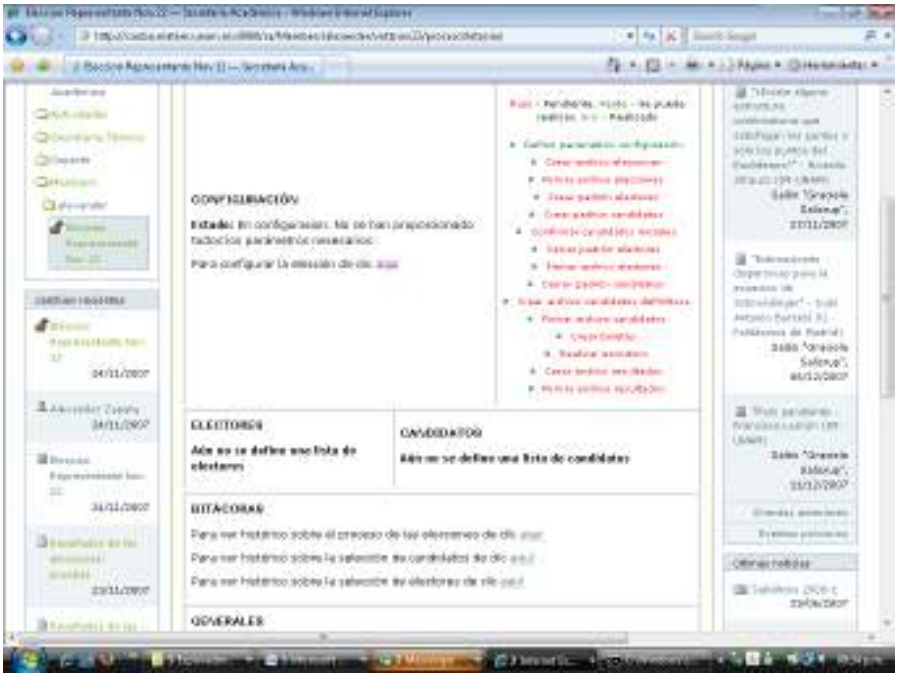
En forma independiente al producto de votaciones el administrador del proceso debe generar en un equipo propio un par de claves con GnuPG siguiendo las recomendaciones del Apéndice IV - Manual de funciones claves GnuPG Punto 1, con el fin de que su clave privada quede protegida.

Una vez se haya generado el par de claves debe exportarse la llave pública siguiendo las recomendaciones del Apéndice IV - Manual de funciones claves GnuPG Punto 2.

Posteriormente debe editarse el archivo donde se exportó la llave pública y pegarse en el sitio plone tal como se presenta en la pantalla siguiente, además de establecerse la longitud de los números aleatorios y de los códigos de candidatos que se van a configurar para generar las boletas de votación (se recomienda un valor mínimo de 40).



Una vez se configuren los parámetros básicos de la votación se puede utilizar el botón "proceso de votación" para utilizar alguna de las siguientes opciones:



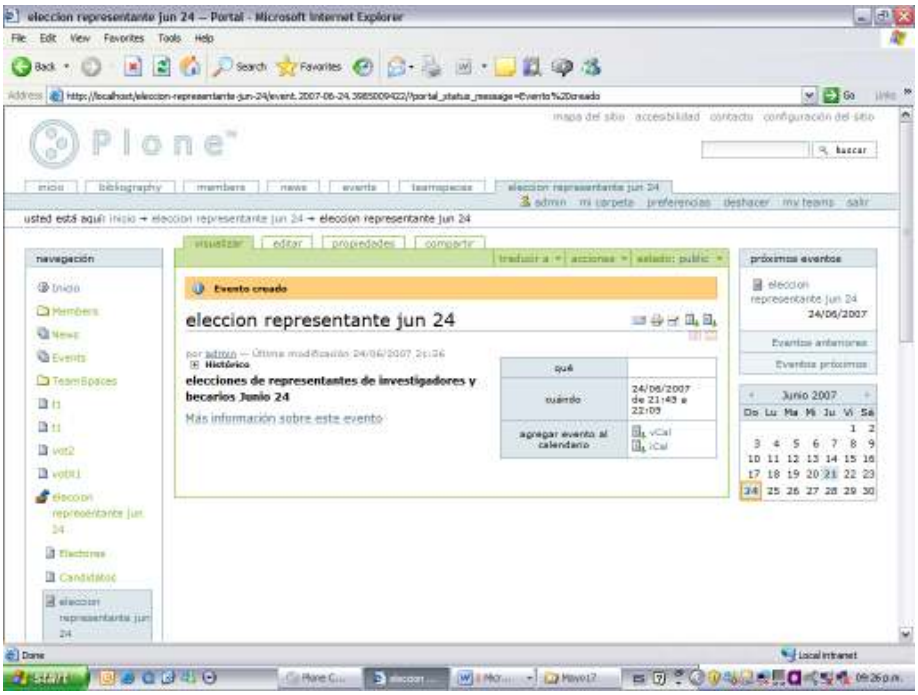
3. Consulta del archivo de las votaciones

Una vez la comisión firme el archivo de configuración éste puede ser consultado por la opción de "Para obtener el archivo de configuración de las elecciones de clic" en CONFIGURACION.

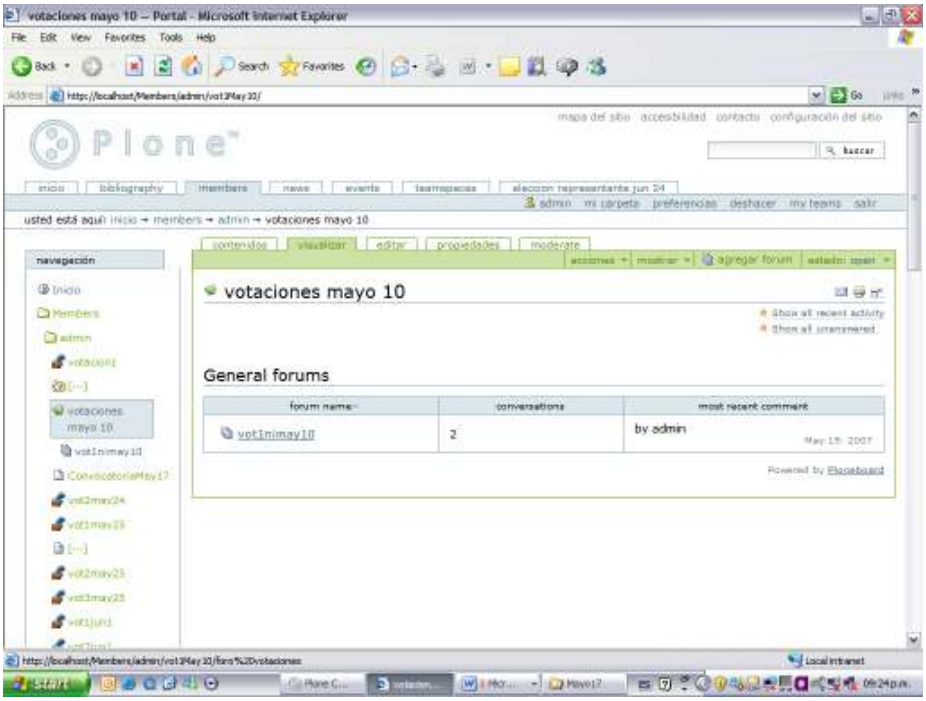
4. Creación del evento del proceso de votaciones.

Mediante la opción de "Crear evento anunciando estas elecciones" en GENERALES

Una vez se cree el evento, éste es publicado en forma automática y todos los usuarios del sitio plone podrán acceder a la votación, desde "Próximos eventos":

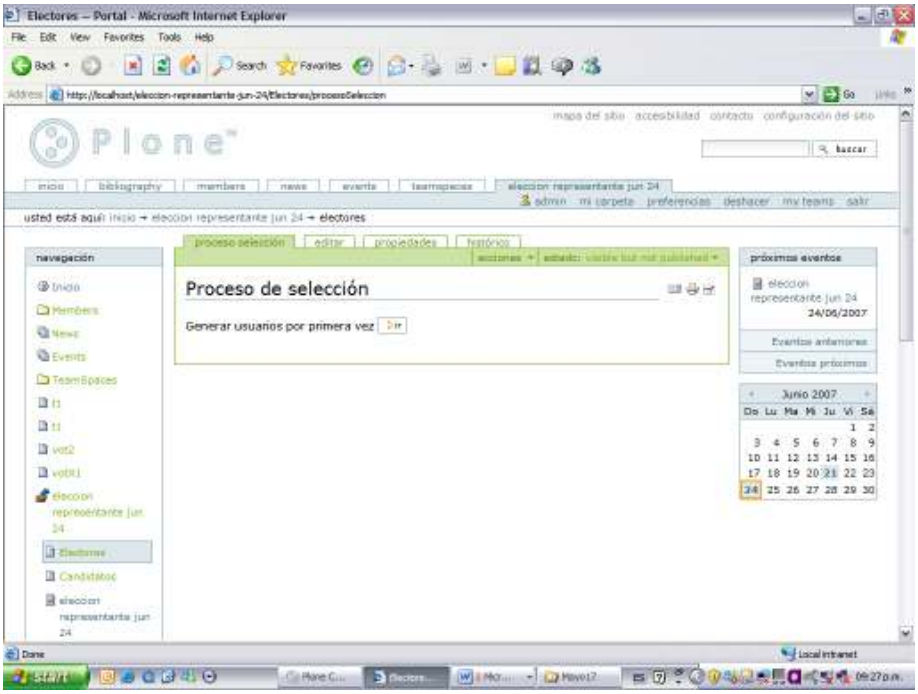


5. Creación del foro de preguntas y respuestas para resolver dudas y reclamos en el proceso, agregando un ítem tipo Message Board, luego seleccionar la opción de agregar Forum – su url se debe incluir en la configuración de la votación.

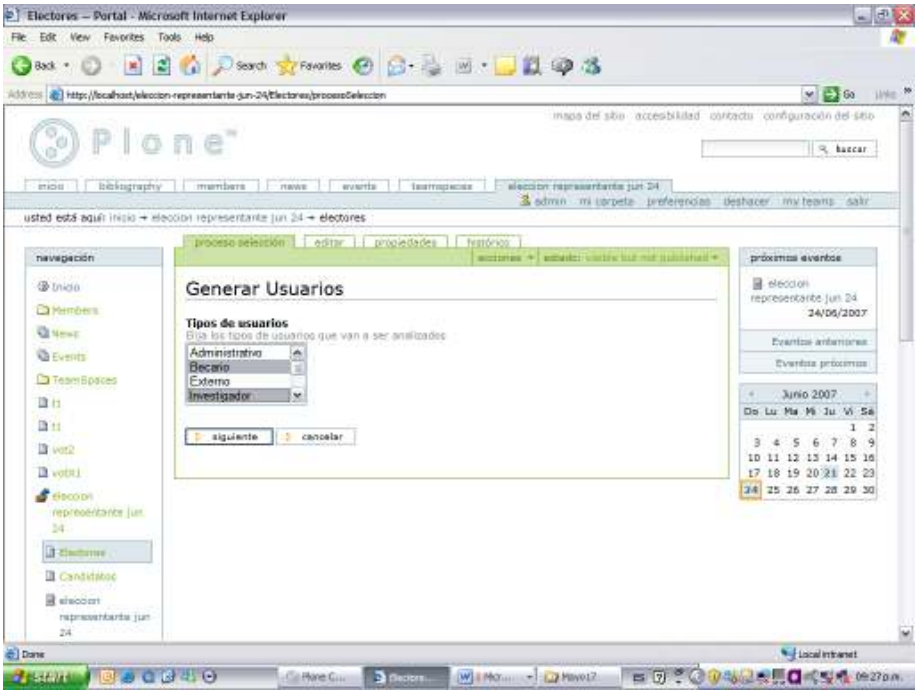


6. Configuración del padrón de electores

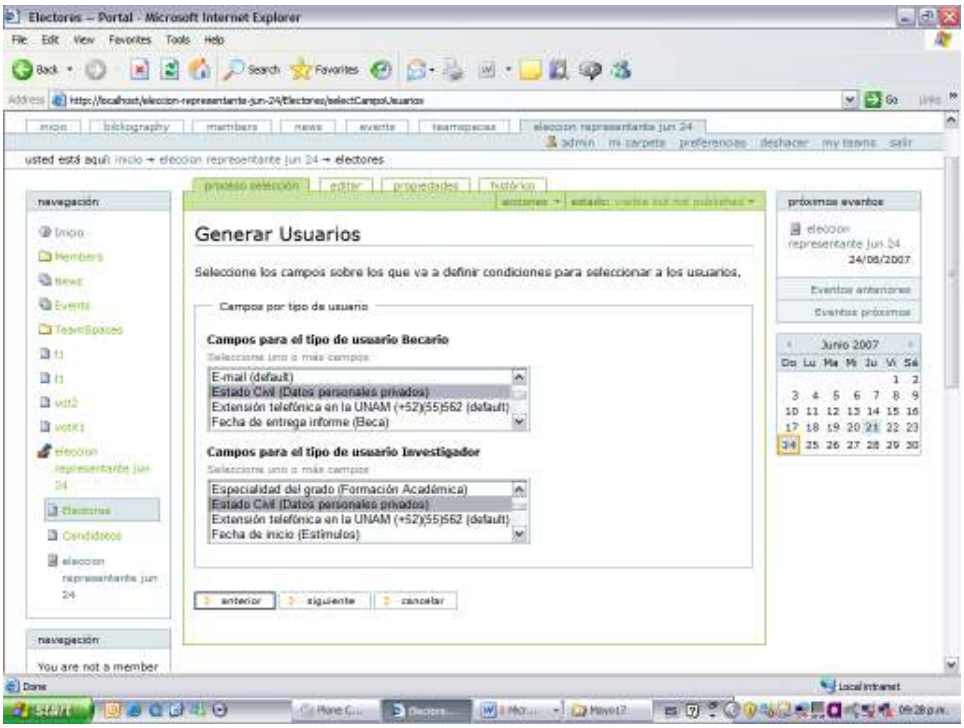
Al ingresar por la opción de "Ir al módulo de selección de electores"



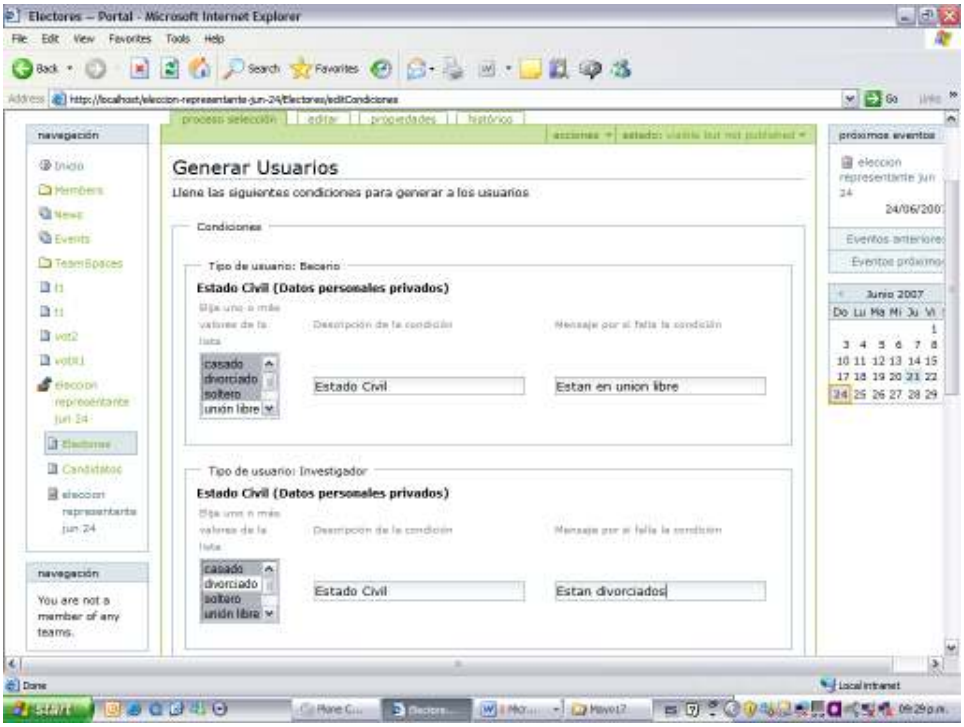
Después deben definirse son los tipos de miembros que podrán ser electores:



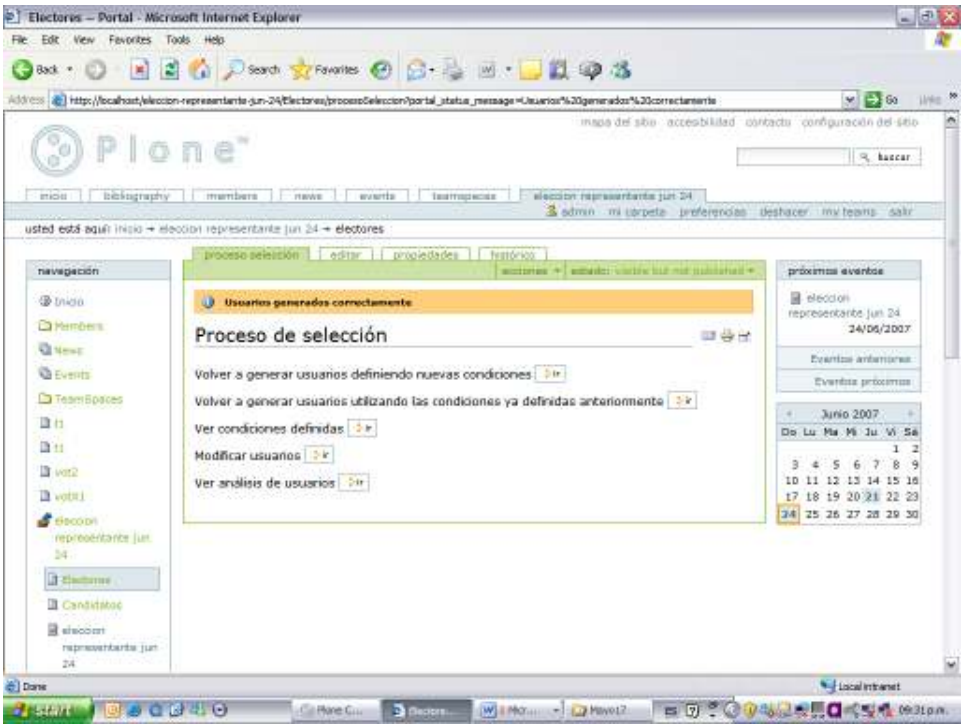
Posteriormente, por cada tipo de miembro seleccionado deben escogerse los campos requeridos para definir los criterios que deben cumplir los posibles electores de la votación:



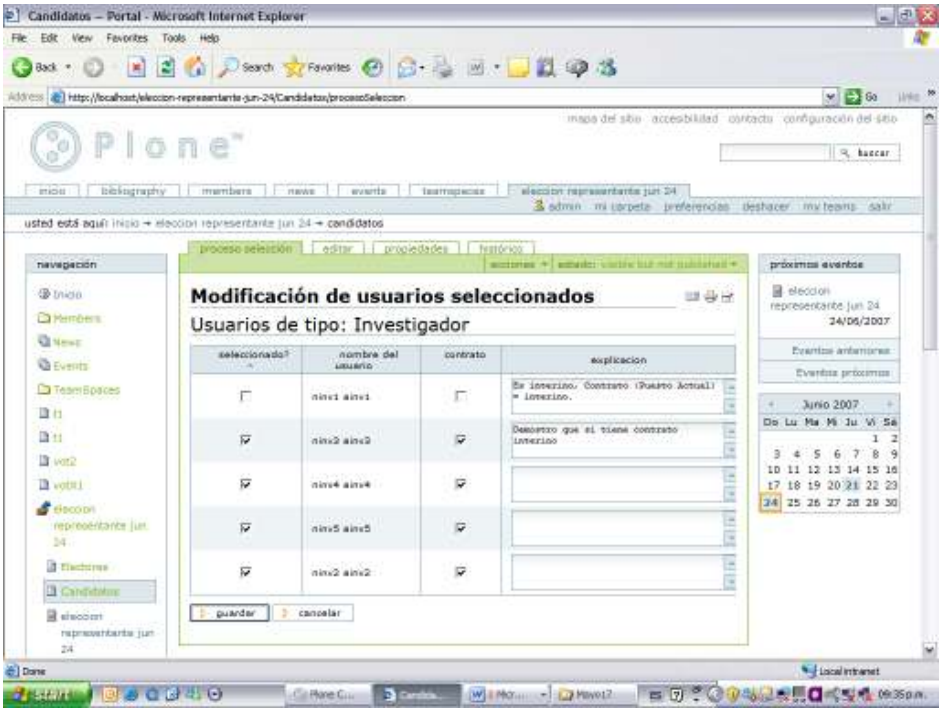
Por cada campo seleccionado deben definirse los criterios a cumplir:



Finalmente puede ajustarse la lista de usuarios seleccionados utilizándose las siguientes opciones:

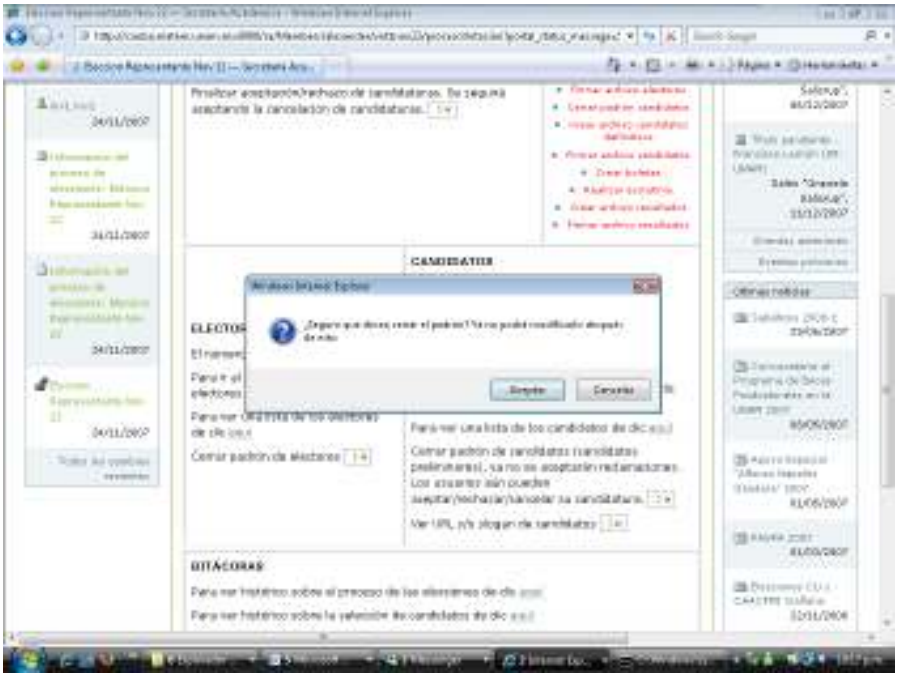


Por ejemplo en el caso de utilizar la opción de "Modificar usuarios":



Para consultar todos los cambios realizados se puede utilizar la opción de "Ver Histórico sobre la selección de electores":

Una vez se tenga la lista definitiva de electores y se llegue a la fecha respectiva que se configuró en el proceso de votaciones se puede cerrar el padrón de electores:



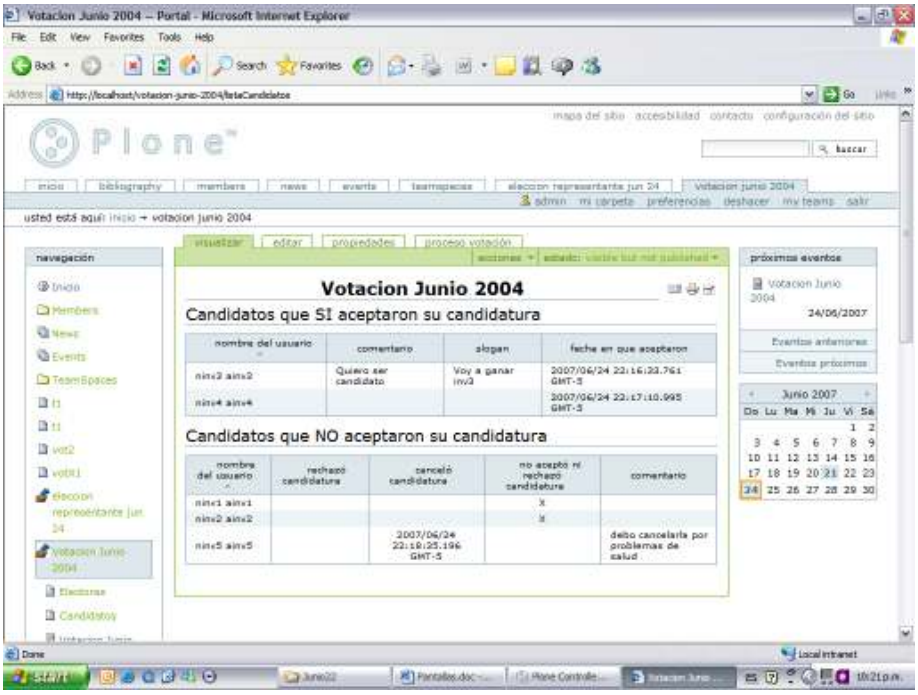
7. Configuración del padrón de candidatos

El proceso es similar al de electores pues se utiliza el mismo producto de selección de usuarios. Una vez se llegue a la lista base de los mismos debe utilizarse la opción de Configurar padrón de candidatos inicial

Posteriormente los usuarios que pueden ser candidatos deberán aceptar o rechazar sus candidaturas en las fechas límites configuradas. En el caso de aceptar una candidatura un usuario puede cancelarla si se encuentra antes de la fecha límite de dicho parámetro. (Ver manual de usuarios generales del proceso)

El administrador puede consultar el status de los candidatos en la opción de "Ver lista de candidatos" en la opción del Proceso de votación:

A partir de este momento los usuarios que pueden ser candidatos deberán aceptar, rechazar o cancelar (después de aceptar) sus candidaturas, lo cual puede ser verificado con la opción de "ver lista de candidatos"



Una vez se llegue a la fecha de generación de padrón definitivo de candidatos se puede seleccionar la opción de "Cerrar padrón de candidatos (candidatos preliminares), ya no se aceptarán reclamaciones. Los usuarios aún pueden cancelar su candidatura" en CANDIDATOS.

Como resultado aparecerá el mensaje "Padron de candidatos cerrado", el cual deberá ser firmado, externamente con gnuPG por la comisión de vigilancia.

En cualquier momento que se quiera configurar otra opción del proceso de votación debe buscarse en "navegación" el objeto de la votación y escoger la opción de "proceso de votación".

Finalmente se debe utilizar la opción de "Crear archivo de candidatos definitivo" en CANDIDATOS.

Debe aparecer la siguiente pantalla confirmando el proceso:

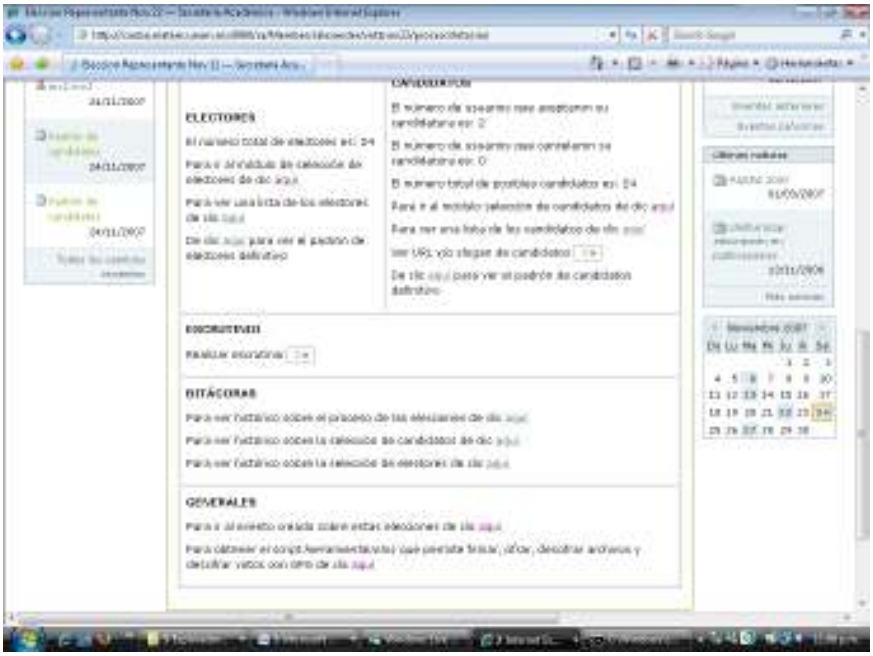
9. Monitoreo del proceso de votación

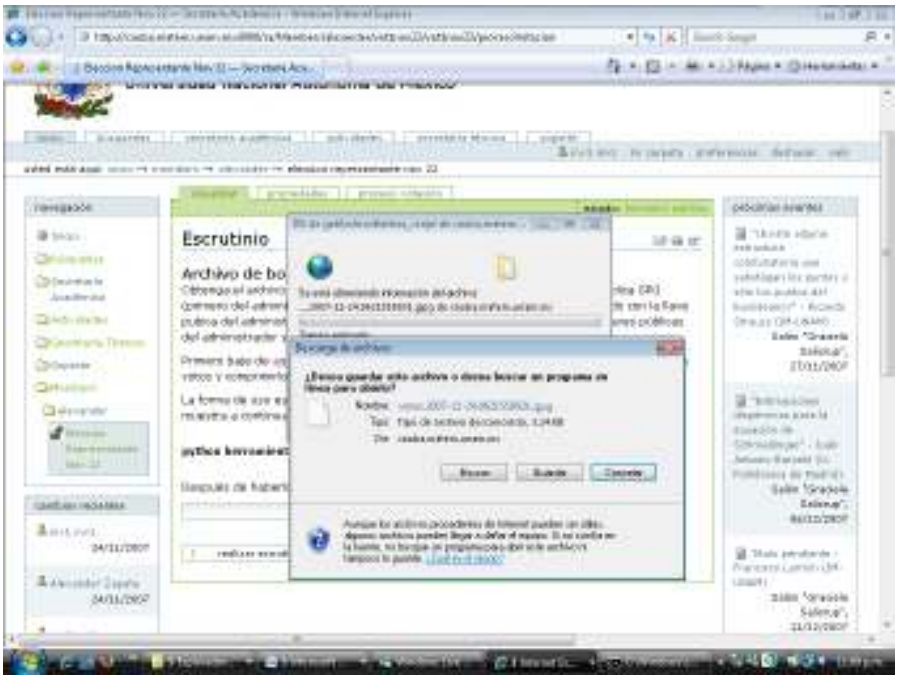
En cualquier momento se puede monitorear el proceso consultando los archivos históricos de las elecciones, los candidatos o los electores.

El histórico de candidatos y electores ya se presentó antes, el histórico del proceso de votaciones presentará una pantalla como la siguiente, con todos los eventos registrados antes, durante la votación y al final el conteo de los resultados

10. Escrutinio

Lo primero que se debe realizar es dar clic en la opción de "Realizar Escrutinio" en ESCRUTINIO y obtener el archivo de votos





Este proceso debe ser realizado por la comisión de vigilancia, pero el administrador participará en la primera ronda de descifrado del archivo de los votos, por lo que debe obtener el archivo de votos y mediante la herramientaVotos utilizar la opción descifrar votos con su clave privada mediante Gnu PG. Referirse al Apéndice 4 – Manual de funciones claves de GnuPG.

También es posible utilizar la opción 4 de descifrar votos de la herramientaVotos que se puede descargar en la pantalla siguiente, y ejecutar con Python herramientaVotos en la carpeta donde se haya generado la clave privada del administrador.

La salida de este proceso debe entregar a la comisión de Vigilancia para que realice el segundo proceso de descifrado y culmine el escrutinio.

Apéndice II – Manual Comisión Vigilancia

Funcionalidad Incluida

1) *Generación y firma del archivo de configuración del proceso de votación.*
2) *Generación y firma del padrón de electores*
3) *Generación y firma del padrón de Candidatos*
4) *Despliegue en forma continua del estado actual del proceso de votaciones.*
5) *Registro de llave pública y Escrutinio de Votos*
6) *Consulta de bitácoras de actualizaciones en el proceso de votación, electores y candidatos.*
7) *Generación y firma del archivo de resultados definitivos del escrutinio*

Requerimientos mínimos:

Ver Apéndice I Manual Administrador – Página 108

Tipos de usuarios en el sistema:

Ver Apéndice I Manual Administrador – Página 108

Productos incluidos

Ver Apéndice I Manual Administrador – Página 109

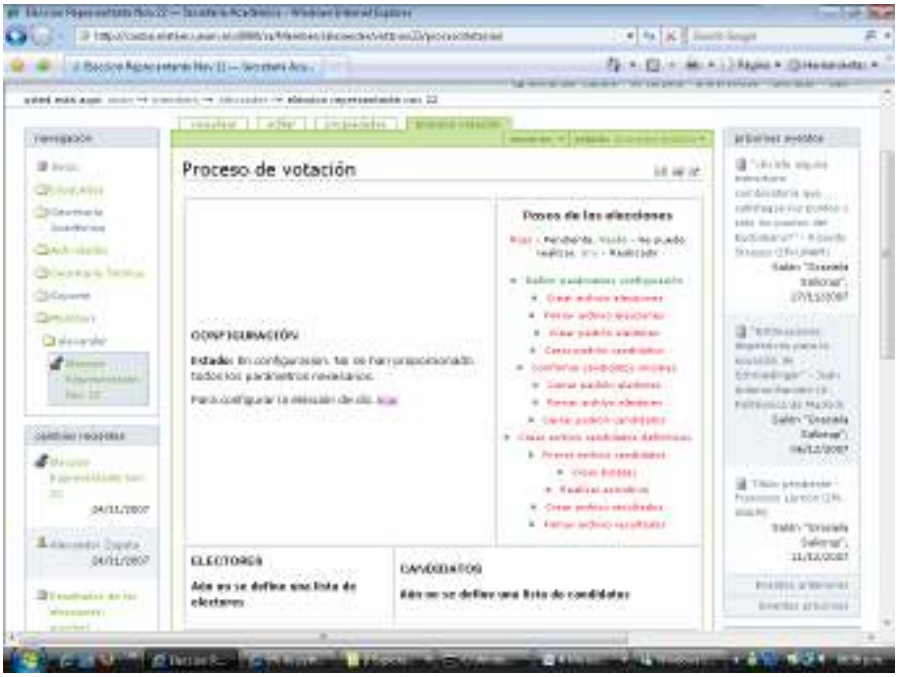
Proceso de Configuración

1. Verificación del estado general del proceso

Ver Apéndice I Manual Administrador – Página 110

2. Configuración de la clave pública y la información general de la comisión

Para facilitar la realización de las diferentes actividades del proceso se incluye una pantalla que puede ser visualizada en cualquier momento mediante la opción de "proceso de votación" , en la cual se presenta la lista de actividades del proceso, presentando en color gris las ya realizadas, en color verde las que pueden ser realizadas y en color rojo las que se encuentran pendientes pero que aún no pueden ser realizadas porque el estado del proceso no lo permite aún.



Al dar clic en en la opción de "Para configurar la elección de clic aquí" en CONFIGURACION, se pueden registrar los nombres de los miembros, comentarios y la clave pública de la comisión de vigilancia que se tomará como base para verificar los documentos firmados (padrones, resultados y configuración de la votación) y cifrar los votos.

En forma independiente al producto de votaciones la comisión de vigilancia del proceso debe generar en un equipo propio un par de claves con GnuPG siguiendo las recomendaciones del Apéndice IV - Manual de funciones claves GnuPG Punto 1, con el fin de que su clave privada quede protegida.

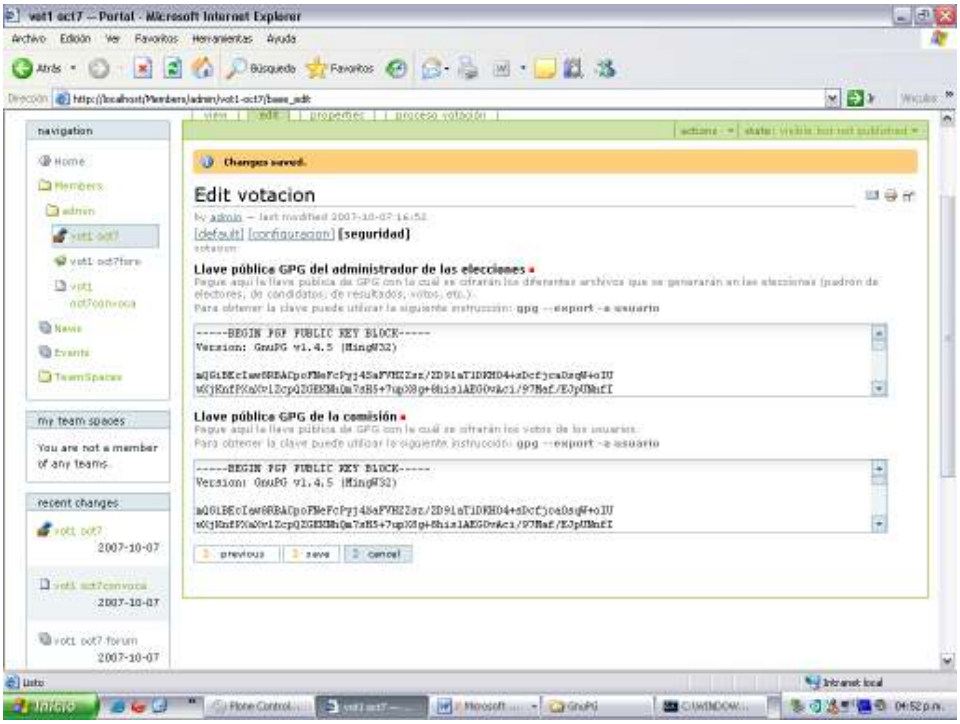
Una vez se haya generado el par de claves debe exportarse la llave pública siguiendo las recomendaciones del Apéndice IV - Manual de funciones claves GnuPG Punto 2.

Posteriormente debe editarse el archivo donde se exportó la llave pública y pegarse en el sitio plone tal como se presenta en la pantalla siguiente, además de registrar la información adicional de la comisión de vigilancia.

Todos los pdf generados (padrón de electores, padrón de candidatos y descripción de la votación, más adelante explicados) sólo se mostrarán cuando ya hayan sido firmados.

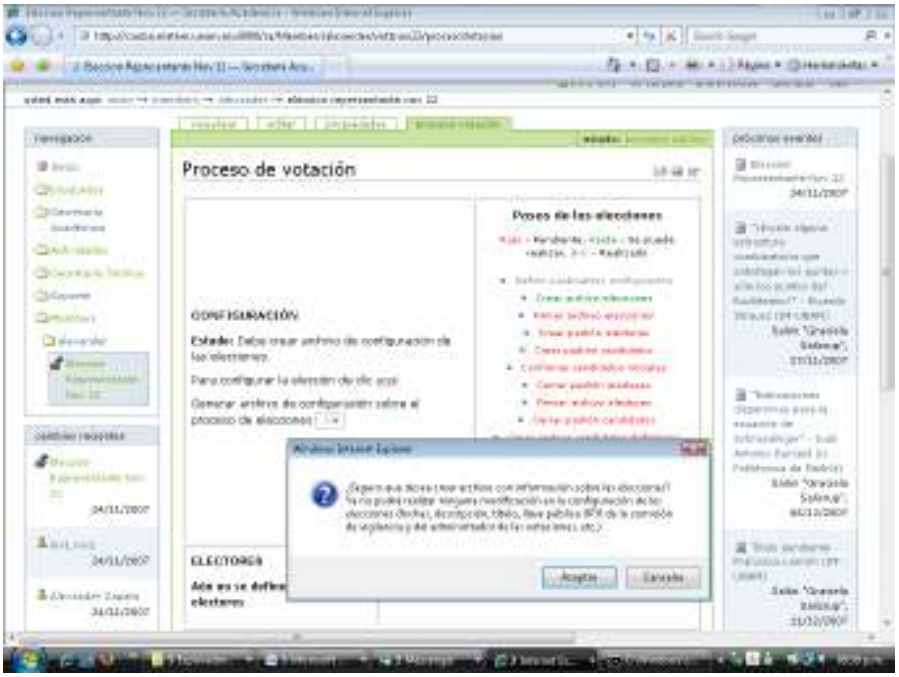
Para generar el par de claves pública y privada. Referirse al Punto 1 del Apéndice 4 – Manual de funciones claves de GnuPG.

Posteriormente debe exportarse la llave pública y pegarse en el sitio plone tal como se presenta en la pantalla siguiente. Referirse al Punto 2 del Apéndice 4 – Manual de funciones claves de GnuPG.



3. Creación del archivo de las votaciones

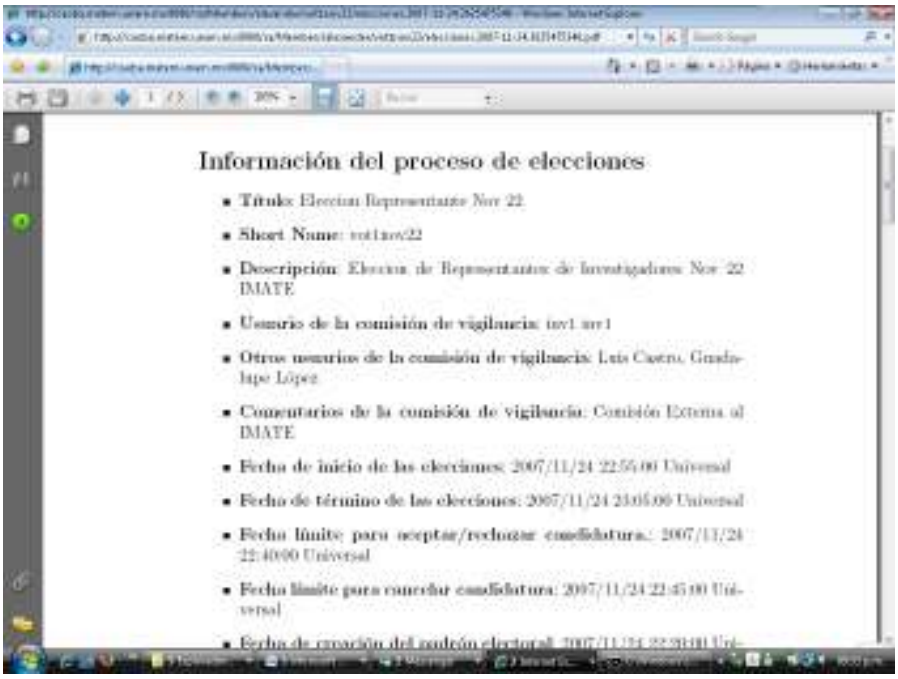
Además de todos los documentos que se pueden referenciar del proceso de votaciones se puede generar un pdf con toda la configuración de las mismas, el cual debe ser firmado por la comisión de vigilancia por fuera del sitio Plone. Después de generar este documento no se pueden realizar cambios en los parámetros de configuración por la opción "Generar archivo con información sobre el proceso de elecciones"



Debe aparecer el mensaje de que se creó el archivo de información de elecciones y se debe firmar el archivo:

Este archivo debe ser firmado externamente por la comisión de vigilancia con Gnu PG por lo que debe guardarse en alguna carpeta y copiarse a la máquina donde se encuentre la clave privada de la misma (la cual generada en un proceso previamente explicado).

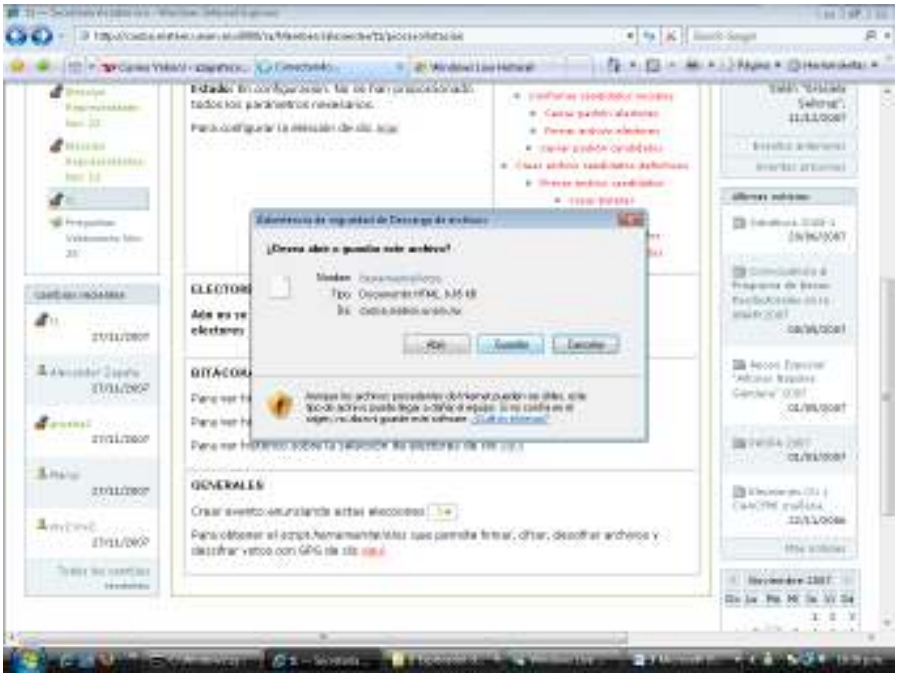
Lo primero que debe hacerse es guardar el archivo generado en la carpeta donde se encuentre la llave privada de la comisión de vigilancia (firmante), por lo que se debe dar clic en "Para obtener el archivo de configuración sobre las elecciones" en CONFIGURACION y luego guardarlo.



Para `poder firmar este archivo y los que se mencionan en pasos siguientes del proceso, como el padrón de electores oficial, el padrón de candidatos definitivo y los

resultados oficiales de las votaciones, se debe descargar el script herramientaVotos con la opción "Para obtener el script herramientaVotos que permite firmar, cifrar, descifrar archivos y descifrar votos con GPG de clic aquí" en GENERALES.

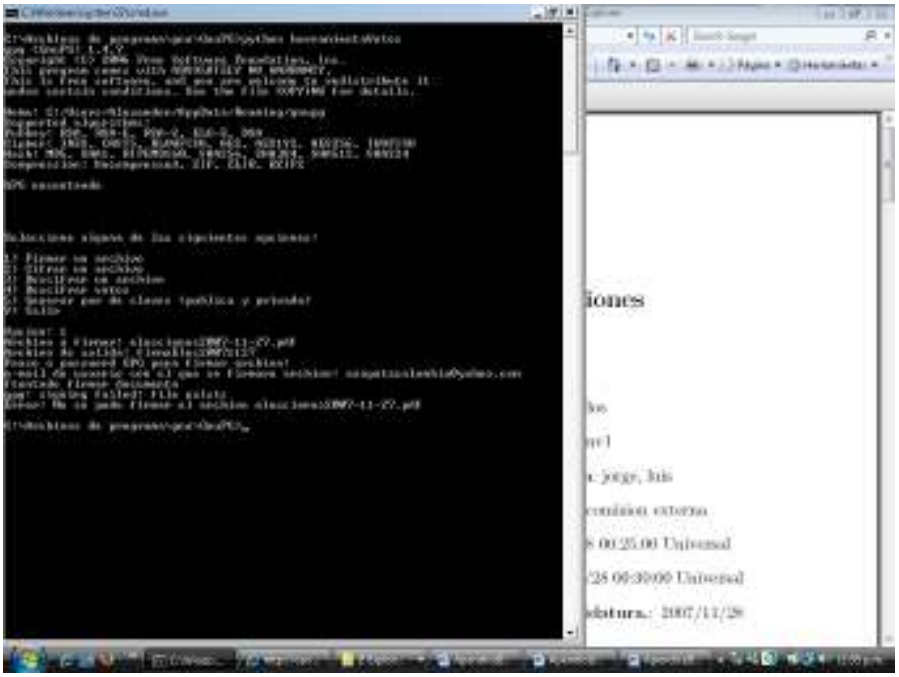
Se recomienda que se guarde en el mismo directorio o carpeta donde se generó el par de claves en un paso previo.



Una vez se haya descargado el archivo se debe ejecutar desde la línea de comando sobre el directorio donde se generaron las llaves y se bajó este archivo el comando:

Python herramientaVotos

Se debe seleccionar la opción 1 para Firmar un archivo, luego digitar el nombre del archivo a firmar (ruta completa), el archivo donde va a quedar la firma, la contraseña y el email asociado a la clave privada que está protegida con la contraseña.



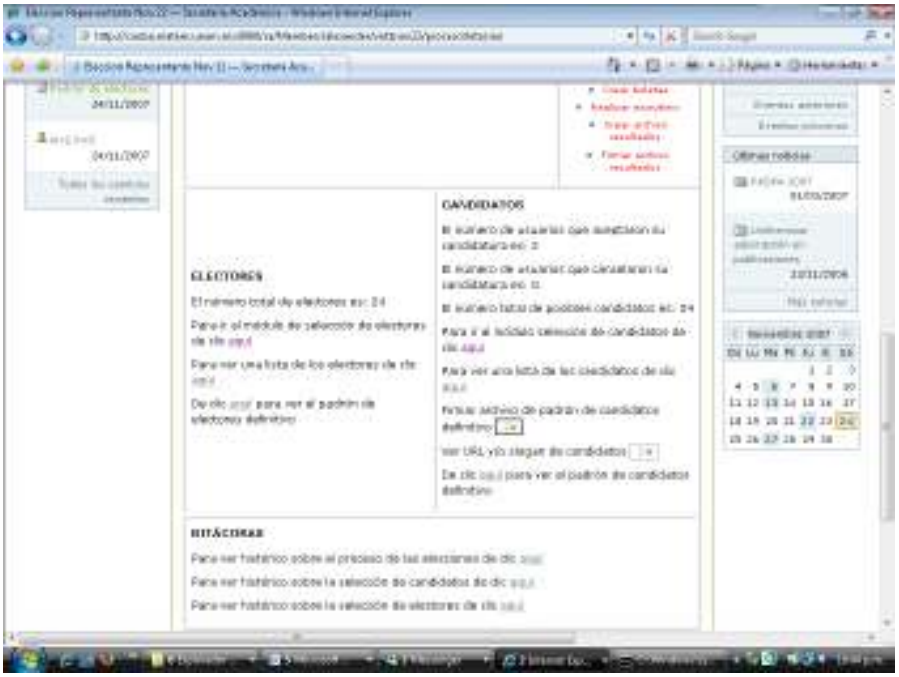
Una vez se firme externamente por parte de la comisión de vigilancia debe editarse la firma con la opción "Firmar archivo de configuración sobre las elecciones" y pegarse en la siguiente pantalla.

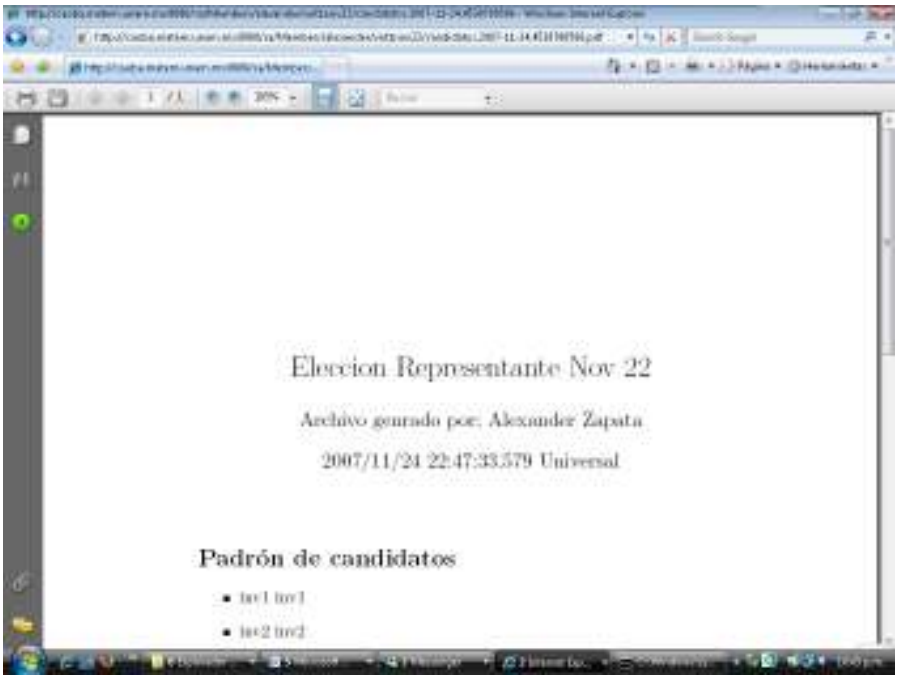
Debe aparecer un mensaje que confirme la firma del archivo, como se mencionó en el caso de alguna diferencia en el contenido del archivo, la llave pública o la firma, no aparecerá el mensaje.

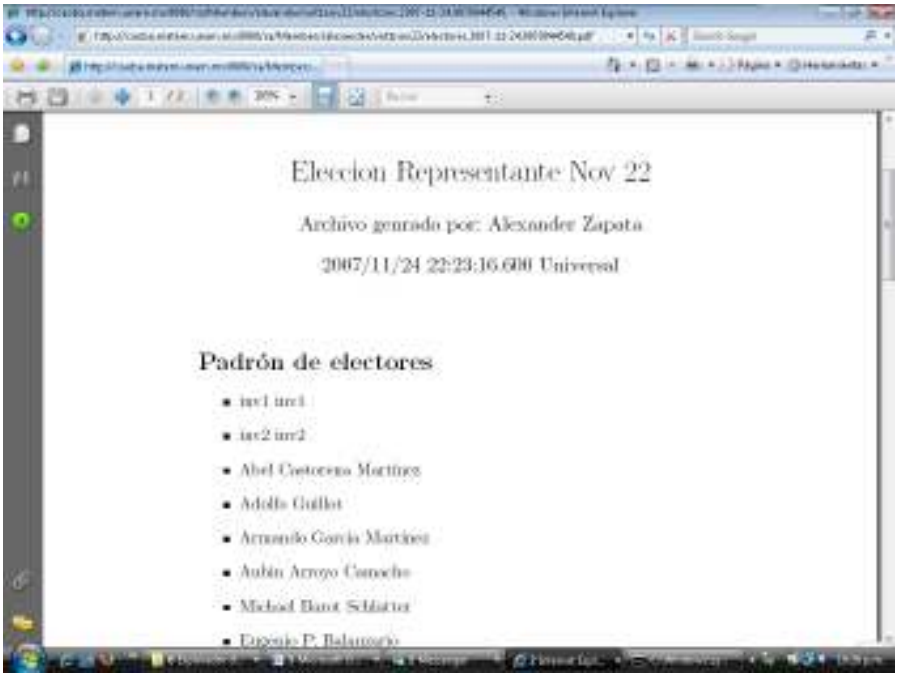
Una vez se obtenga este archivo ya no se podrá realizar ningún cambio a la configuración de las votaciones y los usuarios podrán consultar esta información oficial, verificando que fue firmada por la comisión de vigilancia.

6. Firma del padrón de electores

Antes de firmar el padrón definitivo de electores, la comisión debe verificar el archivo generado por el Administrador de las votaciones, mediante la opción "De clic aquí para ver el padrón de candidatos definitivo" en ELECTORES


CANDIDATOS
ELECTORES
BITÁCORAS


Eleccion Representante Nov 22
Archivo generado por: Alexander Zapata
2007/11/24 22:47:33.579 Universal
Padrón de candidatos
tic1 tic1
tic2 tic2



Este archivo debe ser guardado en la carpeta donde se encuentre la clave privada de la comisión de vigilancia.

Este archivo debe ser firmado externamente con GnuPG por parte de la Comisión de vigilancia y debe pegarse su firma para que la próxima vez que cualquier usuario acceda a él confirme que está firmado por la comisión. Referirse al Apéndice 3 – Manual de funciones claves de GnuPG.

Una vez se edite el archivo con la firma debe pegarse en la siguiente pantalla, mediante la opción de Firmar archivo de electores.

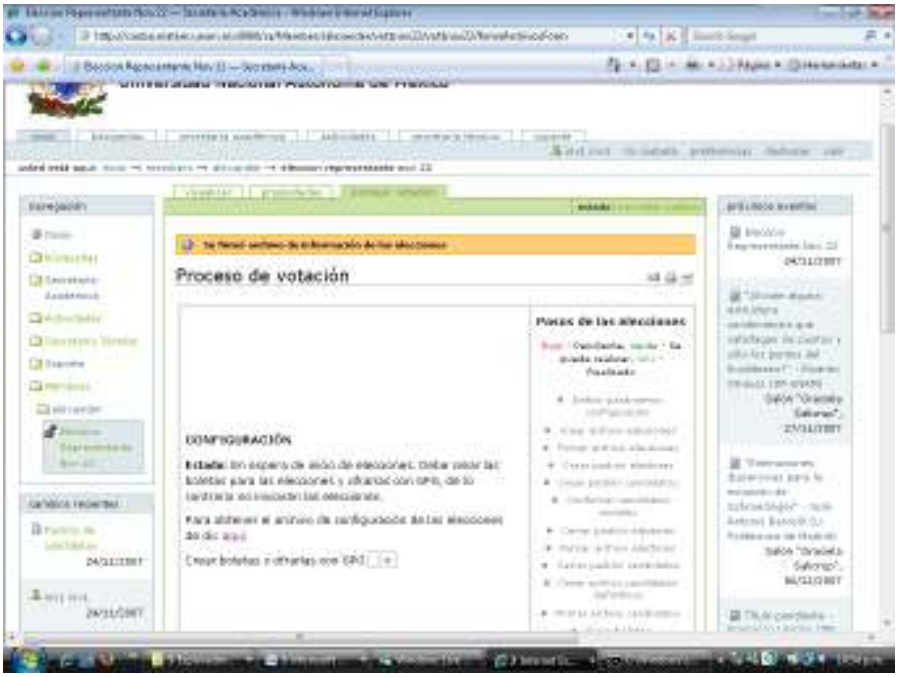
7. Firma del padrón de candidatos

El proceso es similar al de electores pues se utiliza el mismo producto de selección de usuarios.

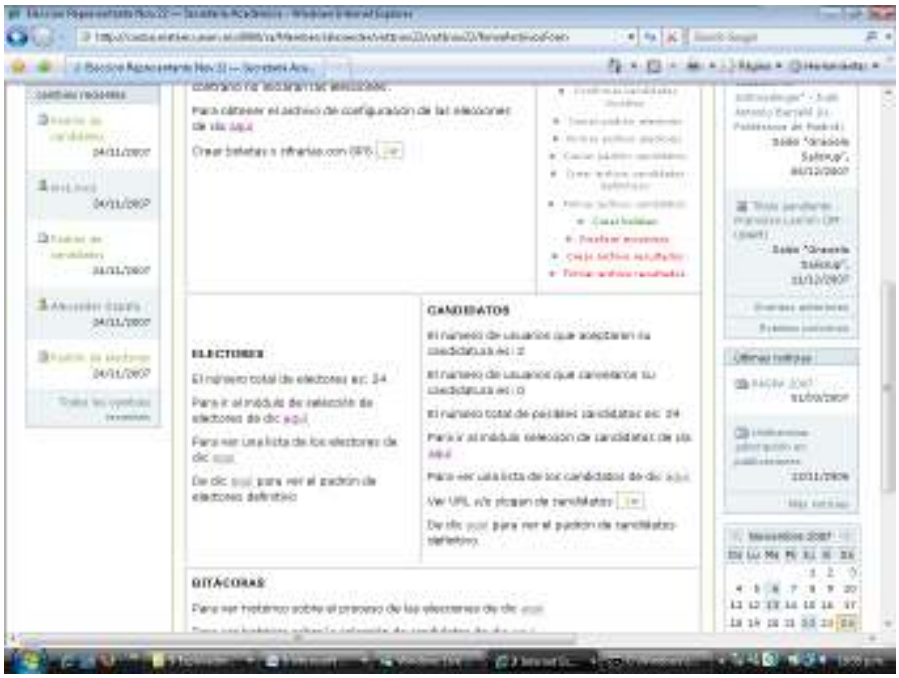
La comisión debe consultar el status de los candidatos en la opción de "Ver lista de candidatos" en la opción del Proceso de votación:


Instituto de Matemáticas
Universidad Nacional Autónoma de México
Firmar archivo *Padrón de candidatos*


Instituto de Matemáticas
Universidad Nacional Autónoma de México
Firmar archivo *Padrón de candidatos*



9. Generación de Códigos de candidatos y de boletas de votación



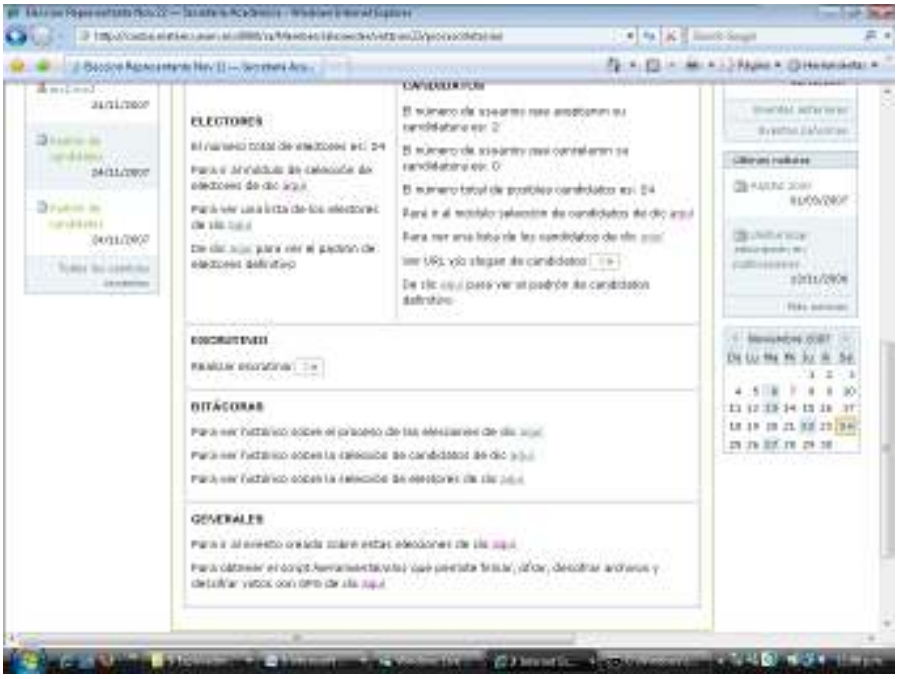
Debe aparecer una pantalla que confirme la generación de las boletas de votación:

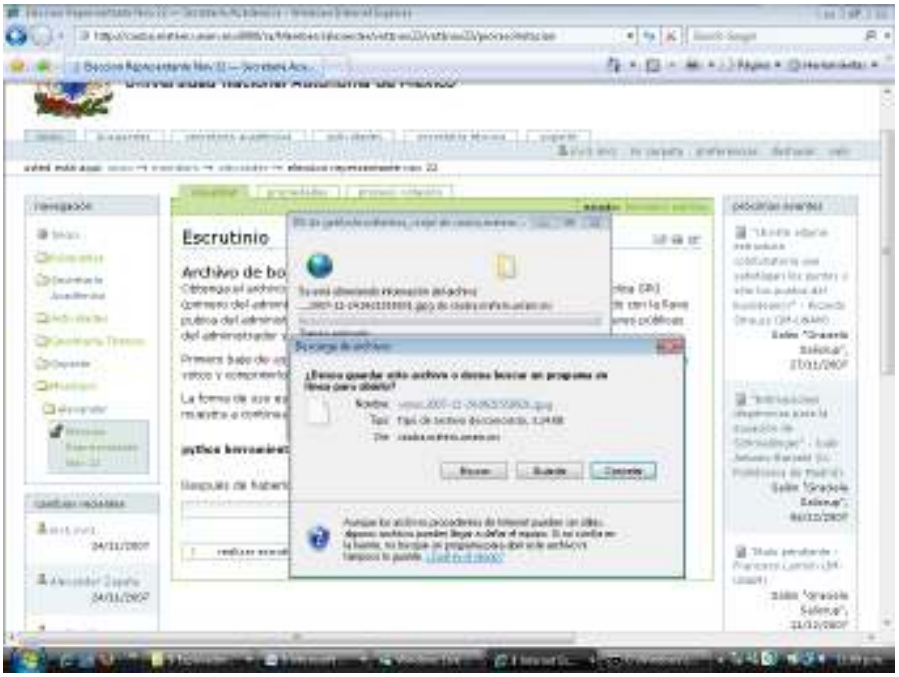
9. Monitoreo del proceso de votación

Ver Apéndice I Manual Administrador – Página 123

10. Escrutinio

Una vez se culminan las votaciones y se llega a la fecha y hora establecida en la configuración de la votación se puede realizar el proceso de escrutinio:

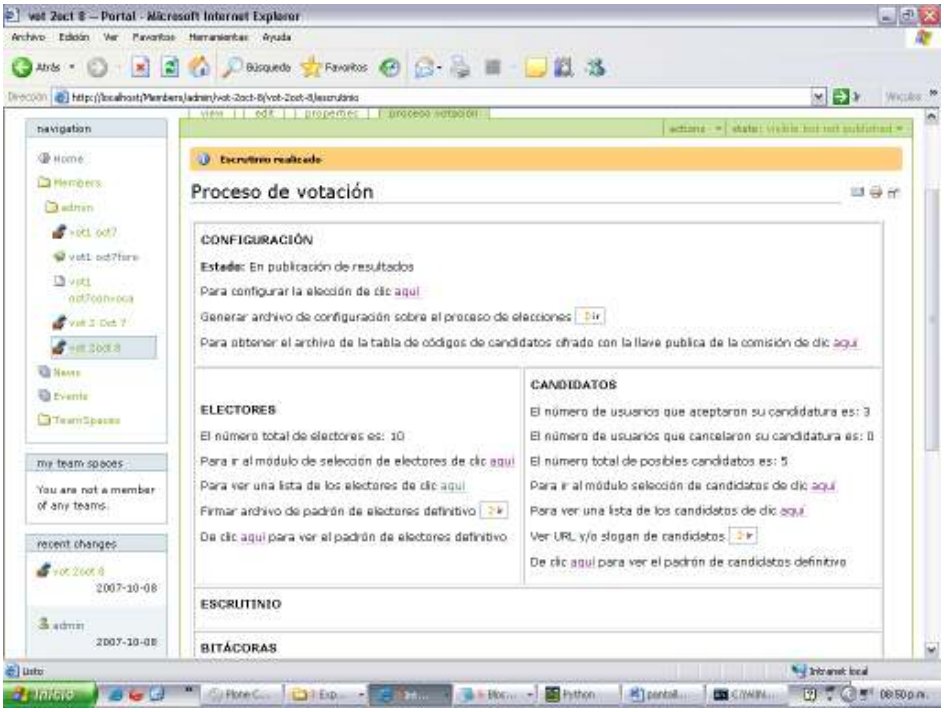




El archivo anterior debe pasar por una primera ronda de descifrado realizada por el Administrador de las votaciones, utilizando la opción 4 de la herramientaVotos que se puede descargar en la pantalla siguiente, y ejecutar con Python herramientaVotos en la carpeta donde se haya generado la clave privada de la comisión.

El archivo de salida debe subirse al escrutinio usando el botón de Examinar

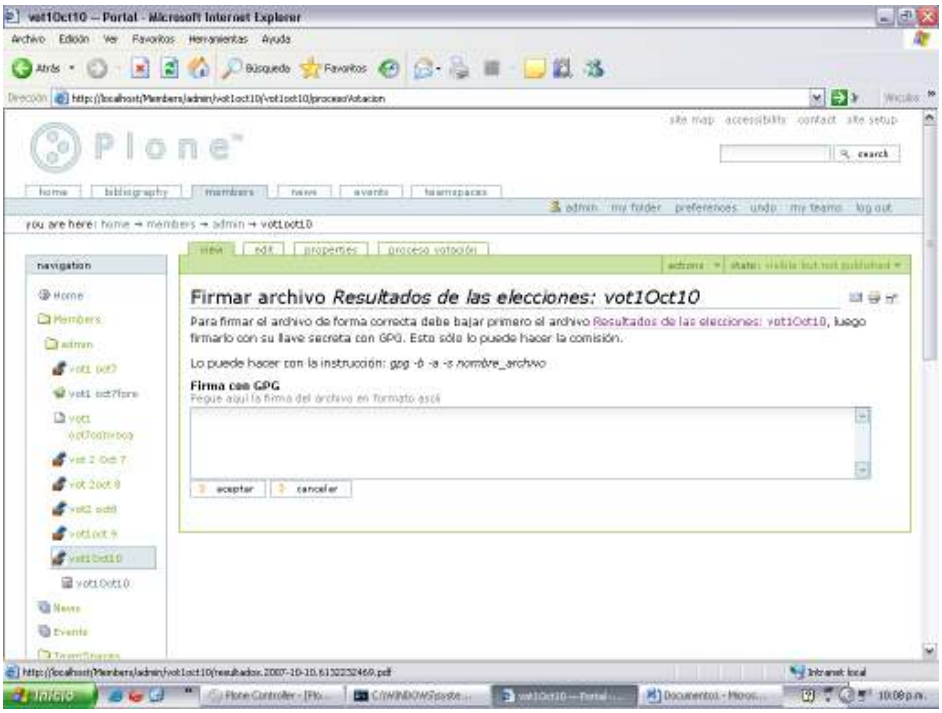
Deberá aparecer un mensaje de "Escrutinio realizado"

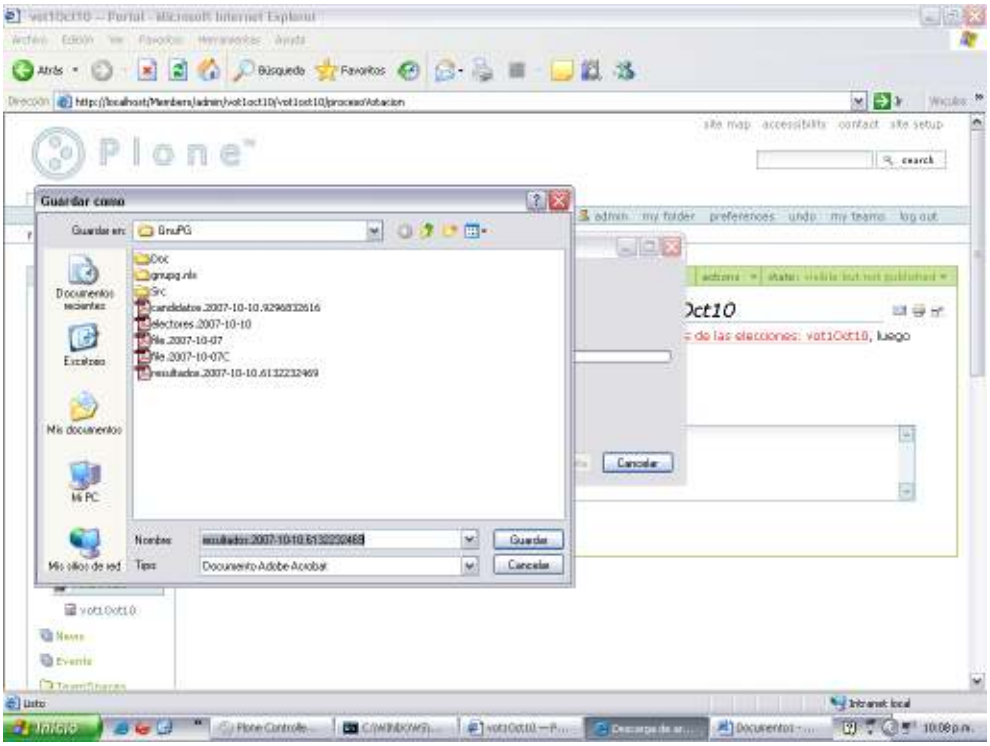


Una vez realizado el escrutinio se pueden consultar los resultados del mismo, en los cuales aparecerán el número total de votos, el porcentaje de los votos por candidato y el detalle de los recibos de votos incluidos en el respectivo conteo, si así lo configuró el dueño de la votación, se podrán también visualizar los números de recibos de votos duplicados en el caso de que se repita el código de un candidato y de los inválidos por no corresponder a un código de candidato válido.

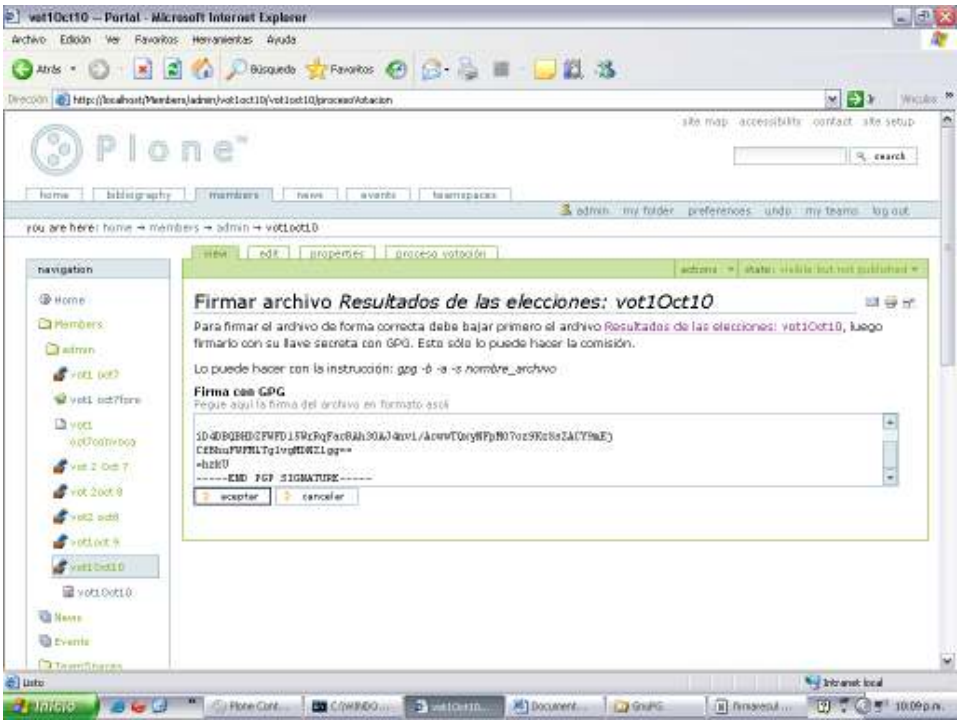
Se debe generar un pdf con los resultados del escrutinio y éste debe ser firmado por la comisión de vigilancia, así como los demás archivos oficiales de la votación, antes de ser publicados.

Una vez se le de click en Firmar Archivo de resultados de elecciones sale la siguiente pantalla para que la comisión guarde un pdf con los resultados:

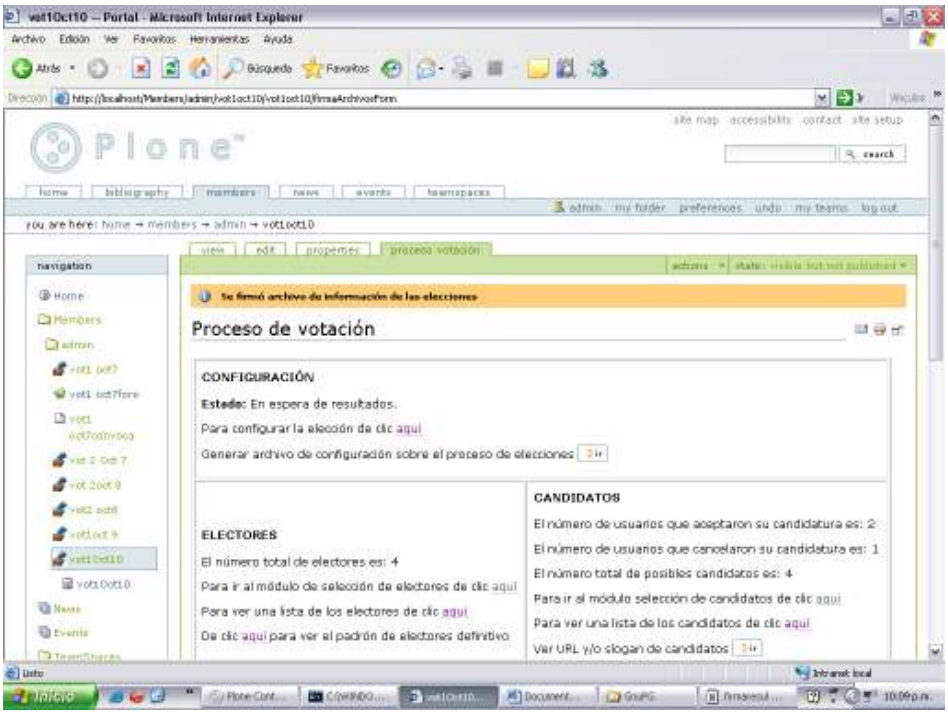




Para que posteriormente pegue la firma respectiva generada con GnuPG. Referirse al Apéndice 3 – Manual de funciones claves de GnuPG.



Finalmente debe aparecer un mensaje de que el archivo fue correctamente firmado y en ese momento los resultados estarán disponibles para todos los usuarios.



Apéndice III – Manual Usuario Final

Funcionalidad Incluida

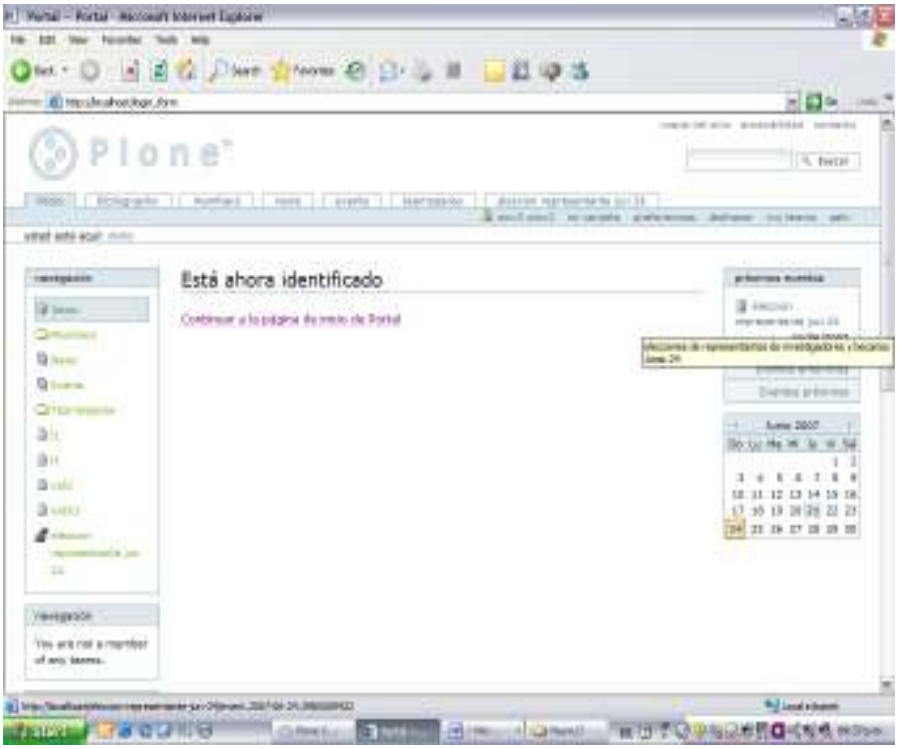
1) *Ingreso al proceso de votaciones*
2) *Verificación de las características y fechas claves del proceso de votación en el que va a participar*
3) *Verificación de su participación como elector*
4) *Verificación de su participación como Candidato*
5) *Aceptación, rechazo y cancelación de candidaturas.*
6) *Despliegue en forma continua del estado actual del proceso de votaciones.*
7) *Consulta de documentos oficiales de la votación*
8) *Registro de su voto*
9) *Generación del recibo del voto para el elector*
10) *Consulta del resultado del proceso de votación y verificación del conteo de su voto.*

Tipos de usuarios en el sistema:

Ver Apéndice I Manual Administrador – Página 108

Ingresar a la votación

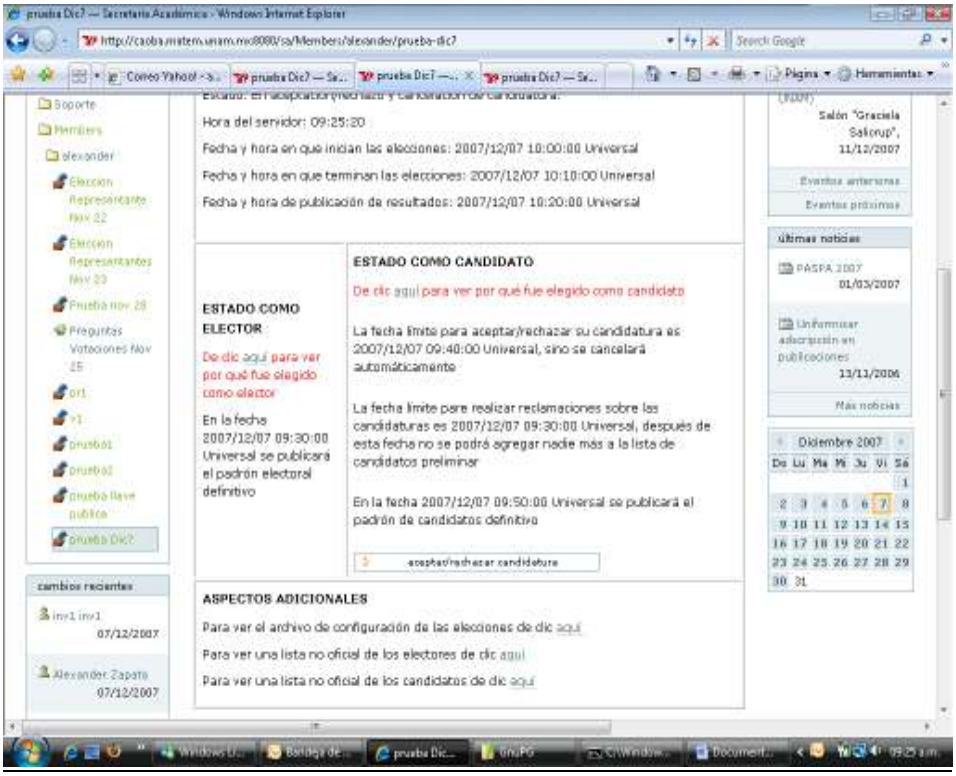
Después de autenticarse en el sistema debe ingresar al evento de la votación que aparece en "Próximos eventos":



Después de dar clic al evento debe ir a la liga de "ver más información sobre este evento" para ingresar a la vista general del proceso de votación.

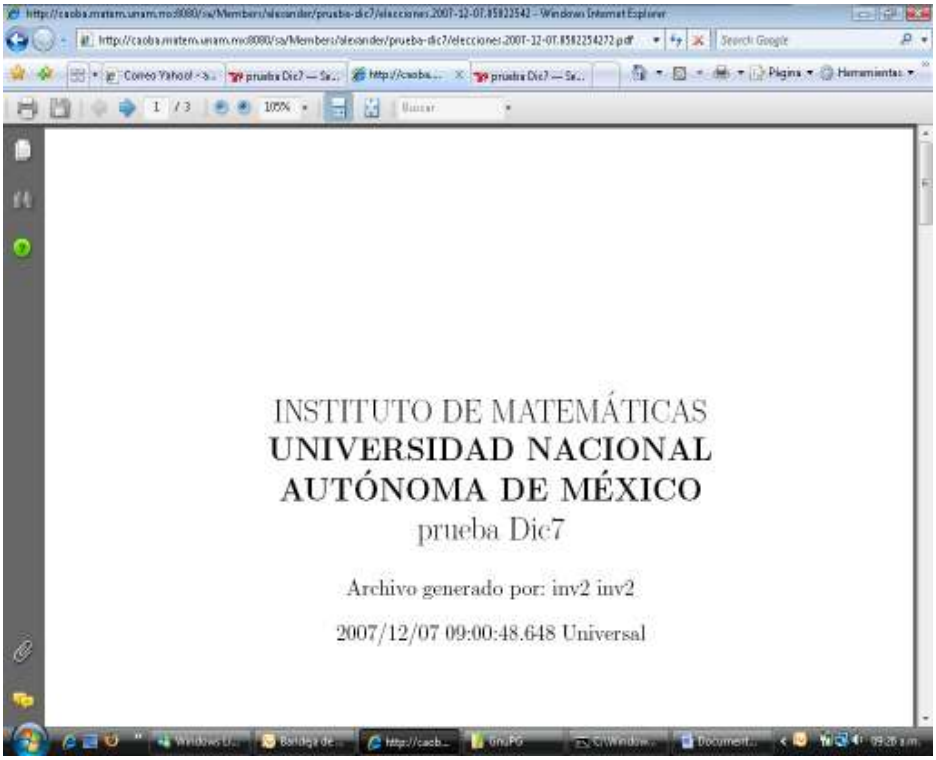
Verificación de las características del proceso de votación en el que va a participar

Lo primero que se recomienda hacer es consultar las características del proceso de votación en el que va a participar, mediante la opción "Para ver el archivo de configuración de elecciones de clic aquí" en la pantalla de ASPECTOS ADICIONALES:



El archivo de configuración debe aparecer firmado por la comisión de vigilancia del proceso:

Este archivo está en formato .pdf y se visualizaría como se presenta en la pantalla siguiente:

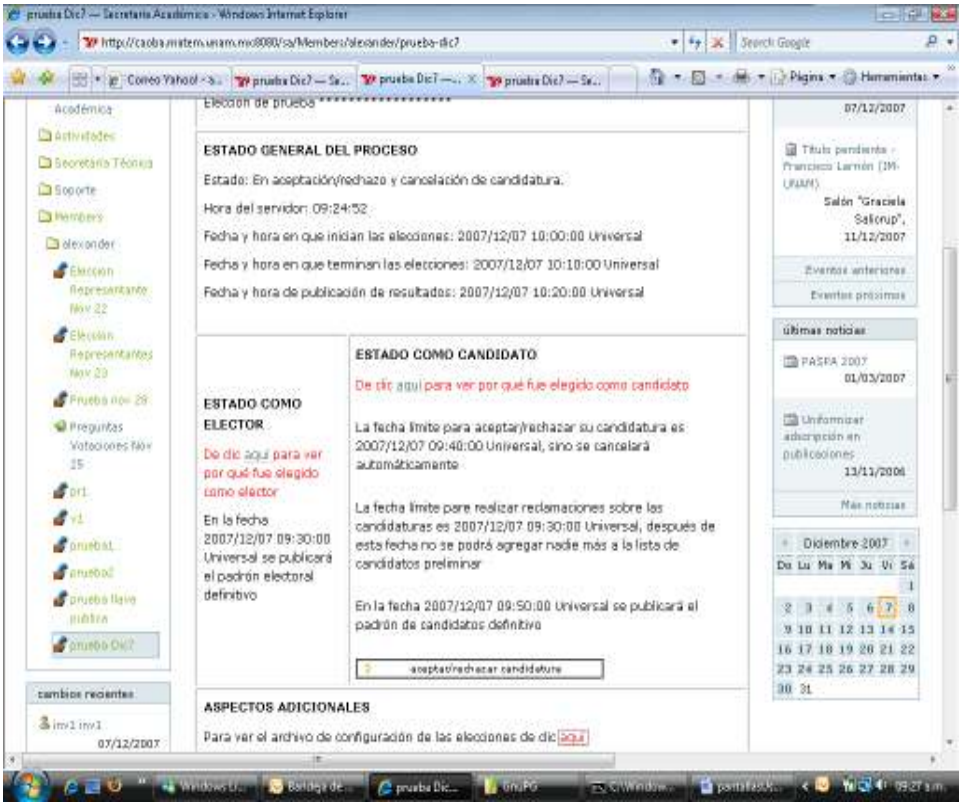


Verificación del estado general del proceso

Ver Apéndice I Manual Administrador – Página 110

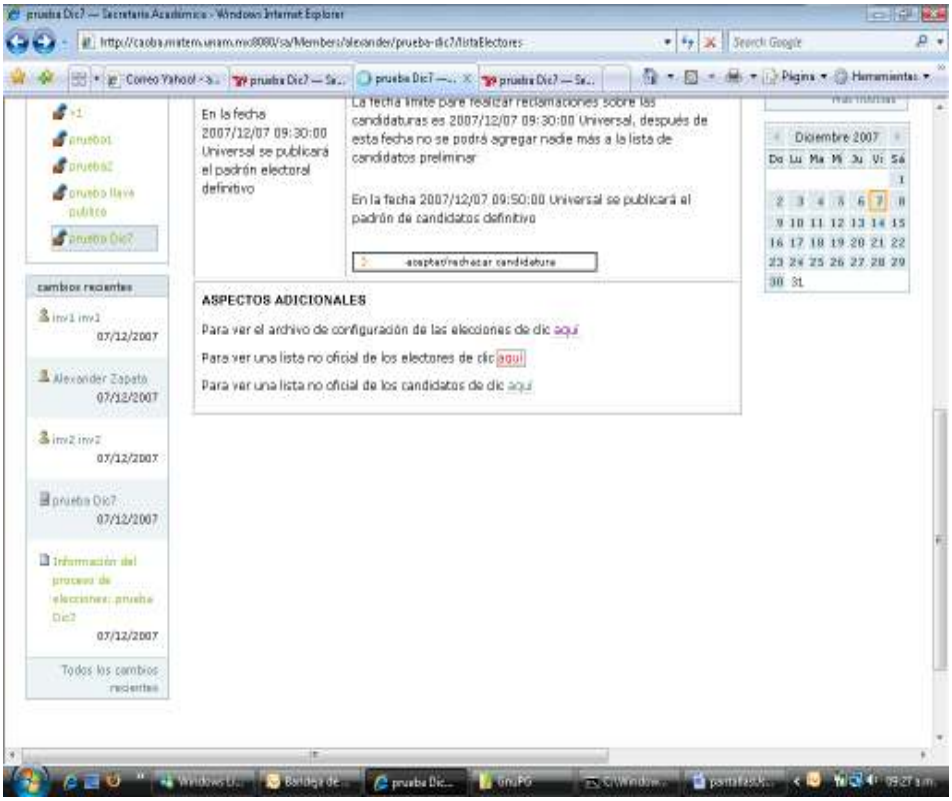
Verificación de participación como Elector

1. Verificar si cumple las condiciones requeridas para votar en el proceso, en la ventana de "Estado como Elector", "De clic aquí para ver por que fue elegido como elector"

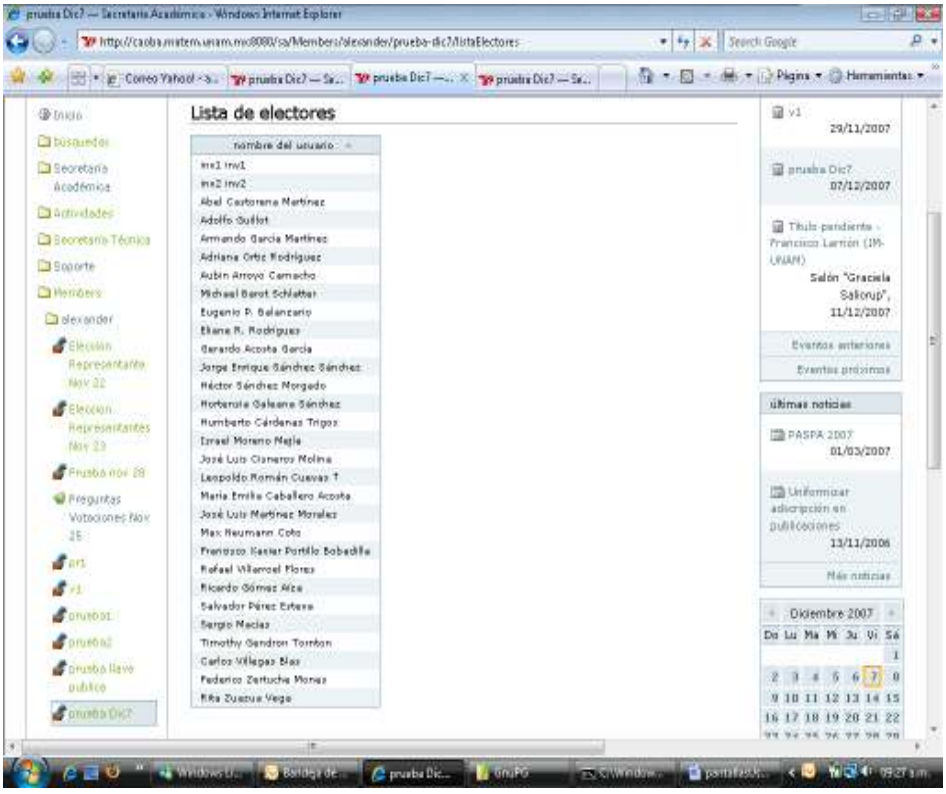


En el caso que no cumpla alguna condición verifíquela y en el caso de algún reclamo o comentario hacerlo al dueño de la votación y solicitarle el ajuste de sus condición como elector válido. Debe reingresar al sistema y verificar que el dueño haya realizado el proceso respectivo usando esta misma opción.

El padrón preliminar de electores puede ser consultado en cualquier momento, para hacer cualquier observación a la comisión de vigilancia. Esta consulta se puede realizar mediante la opción "Para ver lista No oficial de los Electores de clic aquí" en la pantalla de ASPECTOS ADICIONALES.



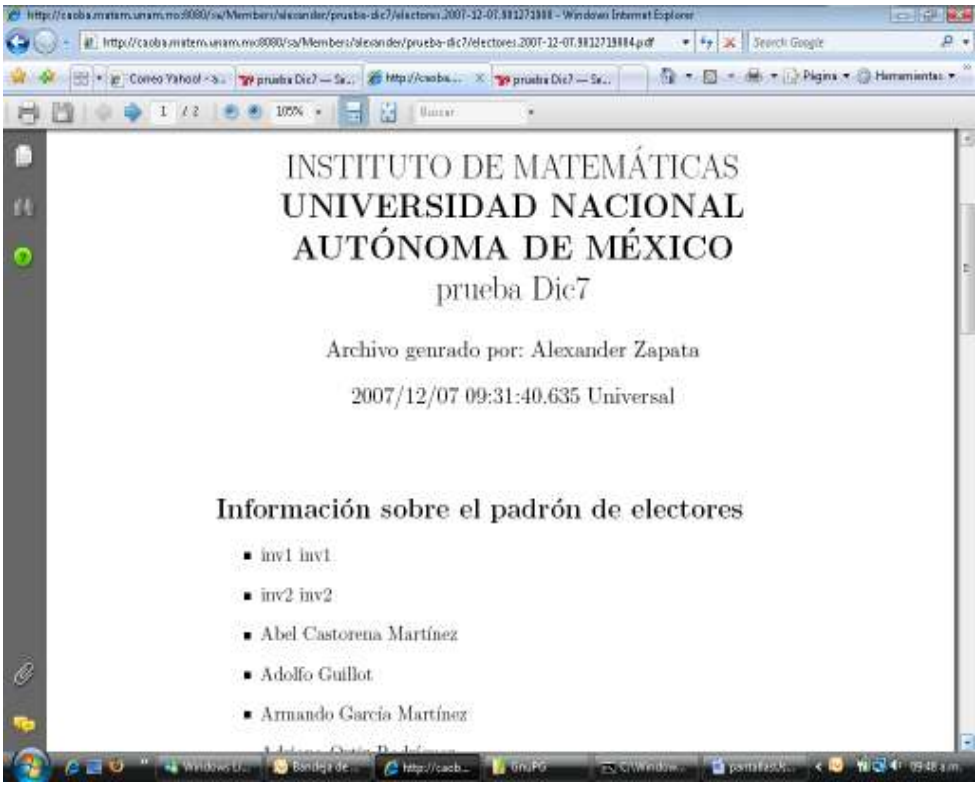
Aparecerá una lista como la siguiente:



Una vez se generé el padrón definitivo de electores podrá ingresar a la opción de "ver padrón definitivo" en la ventana de Estado ASPECTOS ADICIONALES

Se debe verificar que el padrón efectivamente ha sido firmado:

Este archivo puede ser abierto o guardado para realizar cualquier tipo de verificación por parte de los electores, se encuentra en formato pdf.



INSTITUTO DE MATEMÁTICAS
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
prueba Dic7

Archivo genrado por: Alexander Zapata

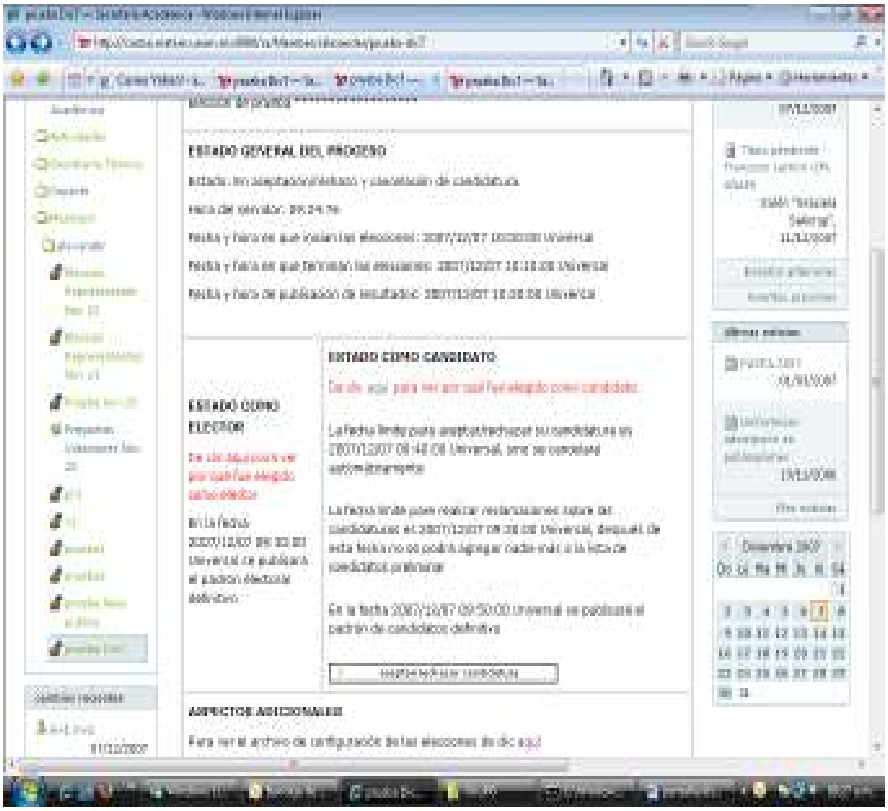
2007/12/07 09:31:40.635 Universal

Información sobre el padrón de electores

- inv1 inv1
- inv2 inv2
- Abel Castorena Martínez
- Adolfo Guillot
- Armando García Martínez

<u>Verificación de participación como Candidato</u>

1. Verificar si cumple las condiciones requeridas para ser candidato en el proceso, en la ventana de "Estado como Candidato", "De clic aquí para ver por que fue elegido como candidato"



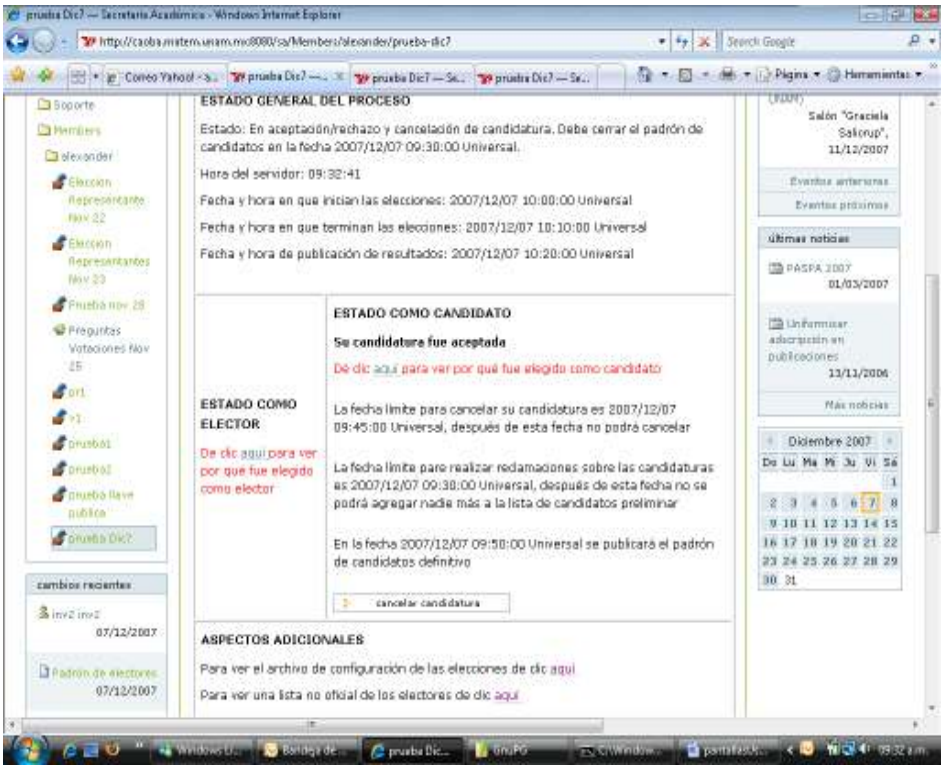
En el caso de no cumplir con las condiciones y estar en desacuerdo comunicarse con el dueño de la votación para solicitarse los ajustes requeridos.

2. Si es un posible candidato y no se ha cumplido con la fecha límite que aparece en la ventana de "Estado como Candidato" podrá aceptar o rechazar su candidatura

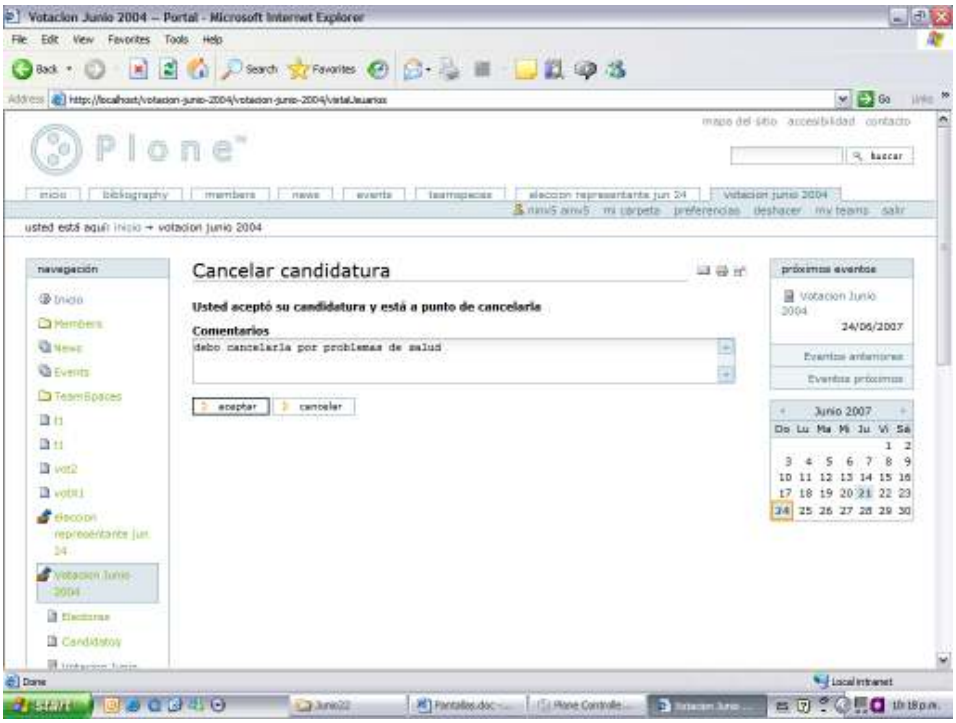
En el caso de aceptar deberá registrar su razón de aceptación, un url donde los votantes podrán consultar información adicional y su slogan:

En el caso de aceptar, deberá aparecer un aviso de candidatura aceptada:

Una vez aceptada una candidatura y en el caso de presentarse un impedimento ésta puede ser cancelada con la opción respectiva en "Estado general del proceso"



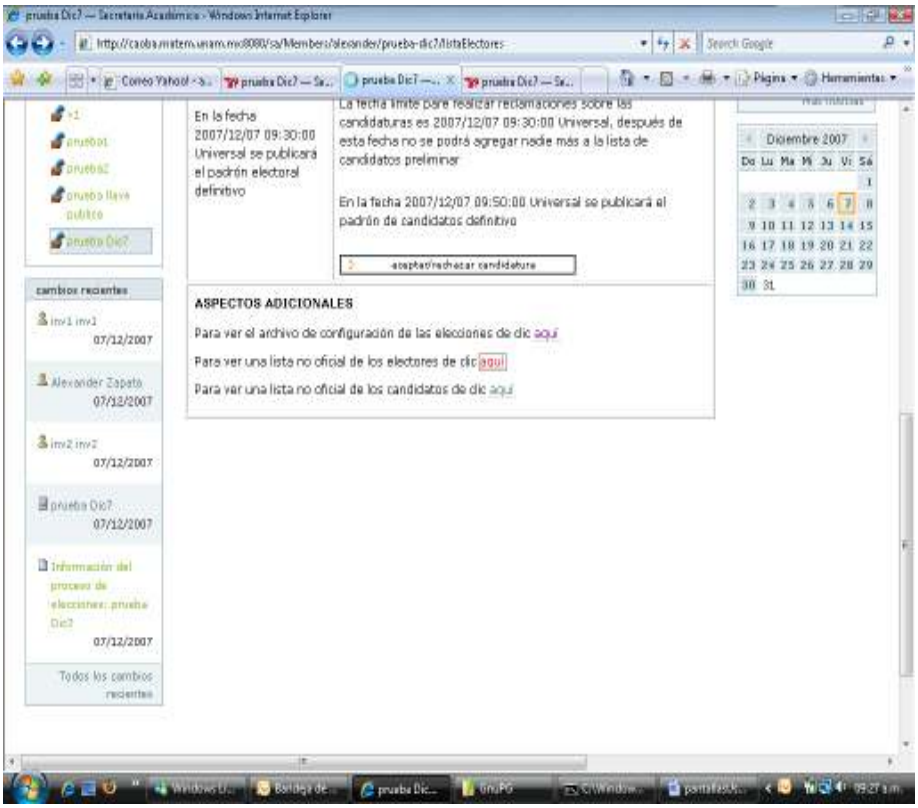
Aparecerá la siguiente pantalla:



Así como en el caso del padrón de electores también se podrá verificar el padrón definitivo de candidatos, el cual debe estar firmado por la comisión de vigilancia.

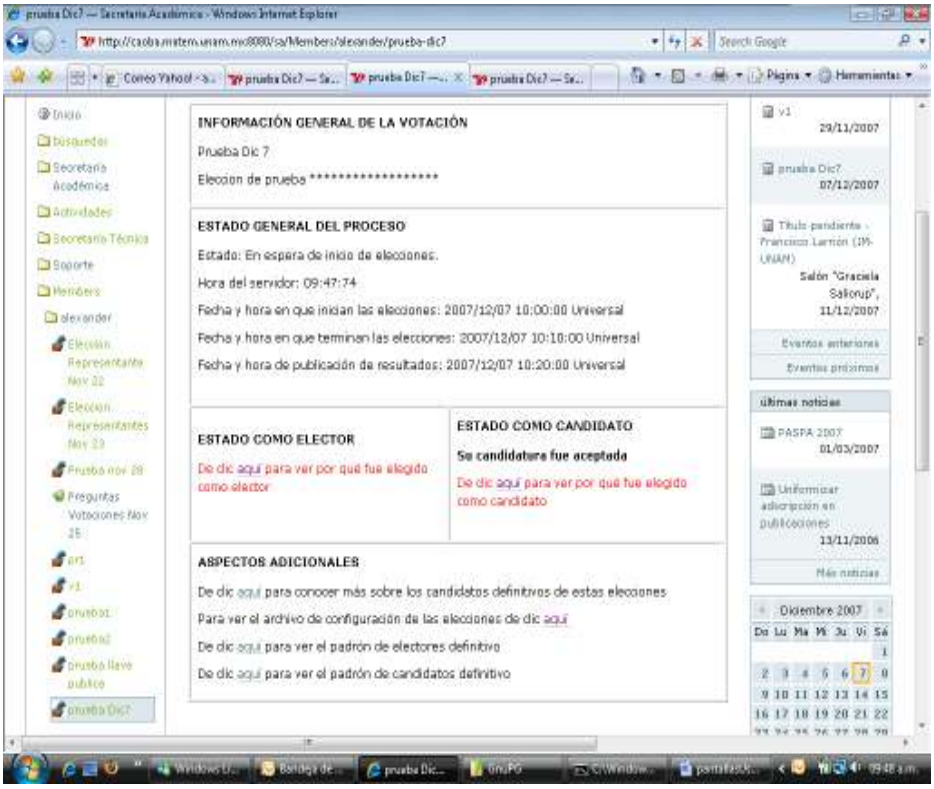
<u>Verificación de Candidatos antes de iniciar la votación</u>

Una vez se haya definido la lista preliminar de candidatos, ésta puede ser consultada por cualquier elector, mediante la opción "Para ver lista No oficial de candidatos de clic aquí" en la pantalla de ASPECTOS ADICIONALES.

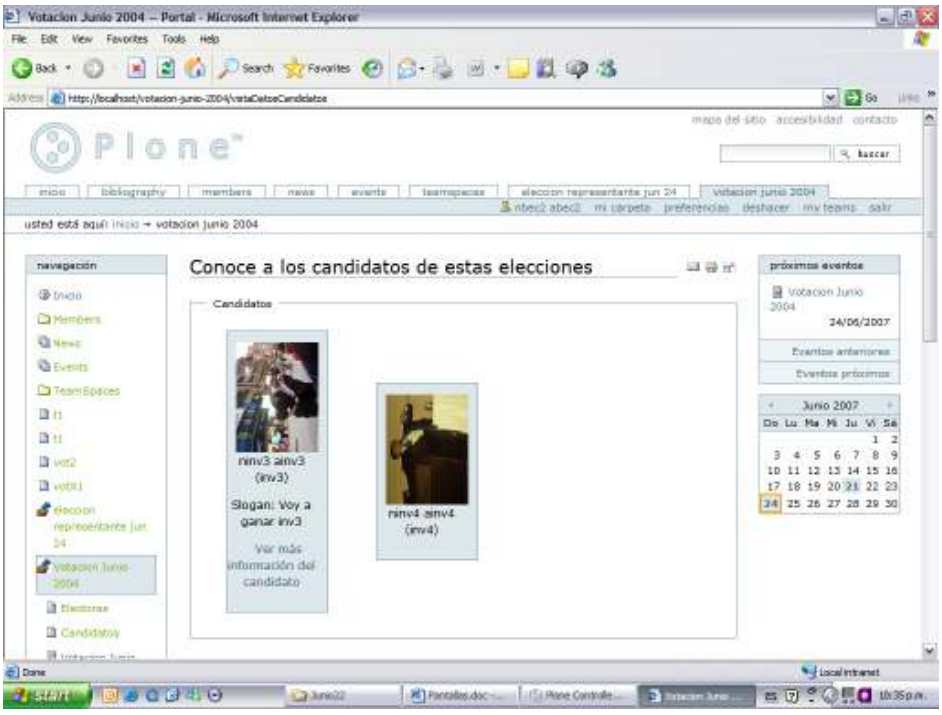


Aparecerá una lista como la siguiente, la cual puede tomarse de base para hacer cualquier tipo de reclamo u observación a la comisión de vigilancia del proceso, en el caso que algún candidato haya aceptado ya su candidatura, aparecerá en la parte superior de la pantalla:

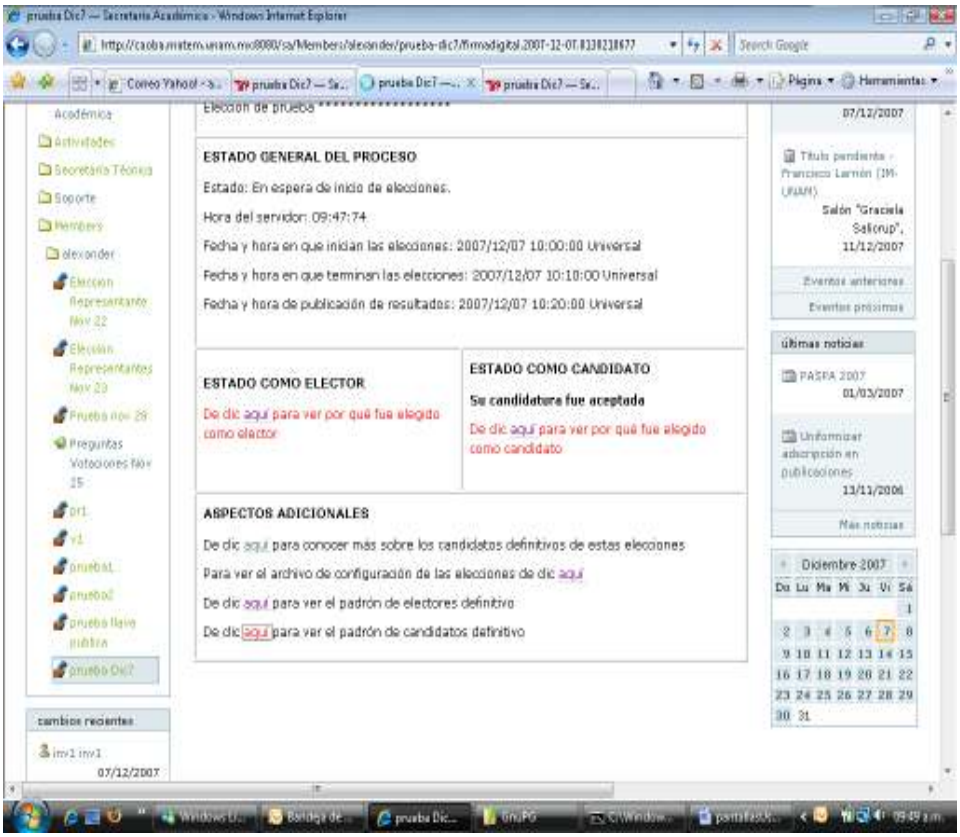
Una vez se haya definido el padrón definitivo de candidatos cualquier usuario podrá conocer los candidatos definitivos en la ventana de ASPECTOS ADICIONALES, y analizar información adicional como su slogan y url, si estos la registraron, en la opción "De clic aquí para conocer más sobre los candidatos definitivos de estas elecciones".



Aparecerá una pantalla como la siguiente, donde se podrá visualizar el slogan de lso candiatos:

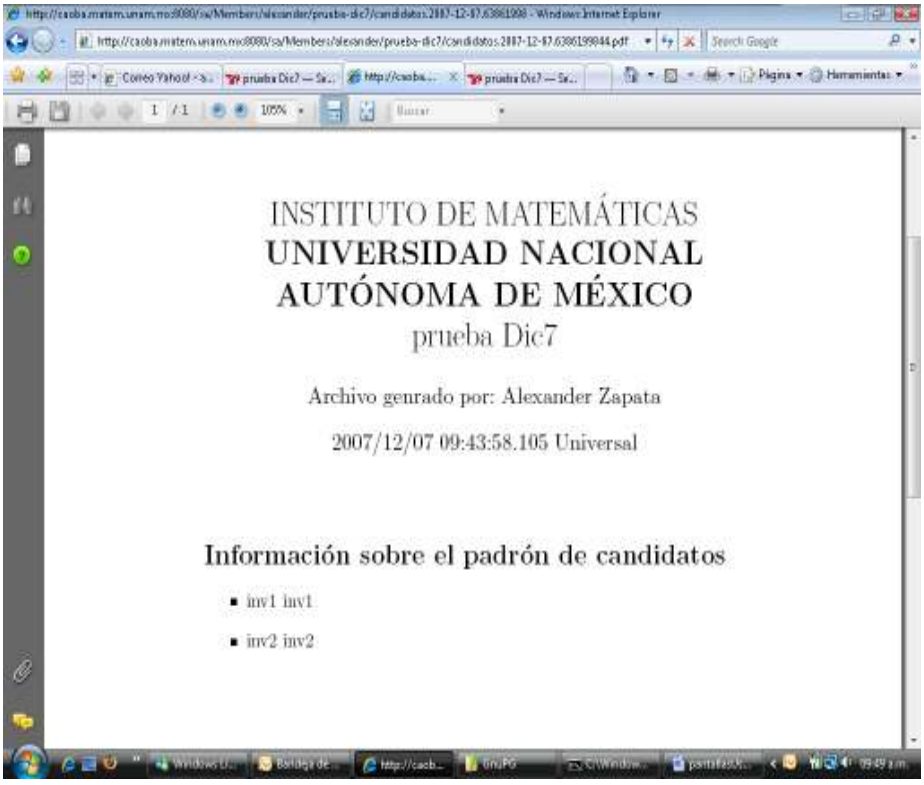


También podrán verificar el padrón definitivo de candidates en ASPECTOS ADICIONALES, mediante la opción "De clic aquí para ver el padrón de candidatos definitivo".



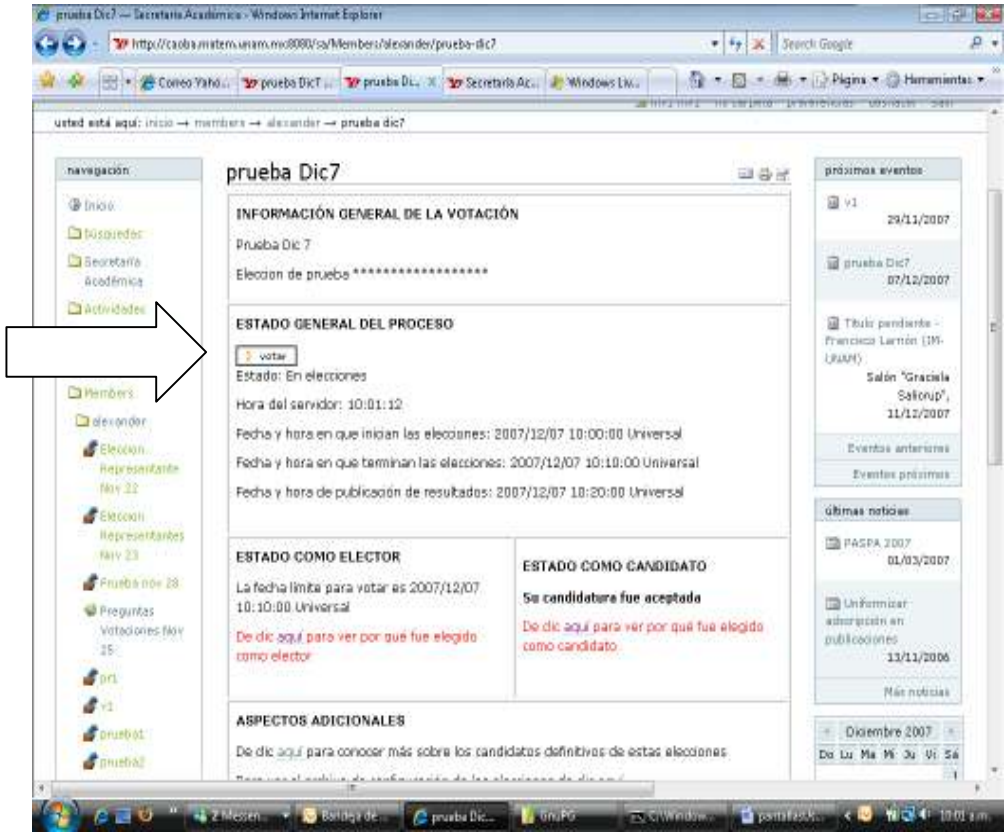
Se puede también verificar el archivo con el padrón definitivo de candidatos firmado por la comisión de vigilancia del proceso:

Si se quiere se puede sólo abrir el archivo o guardar una copia para realizar cualquier tipo de verificación, del archivo en formato pdf:

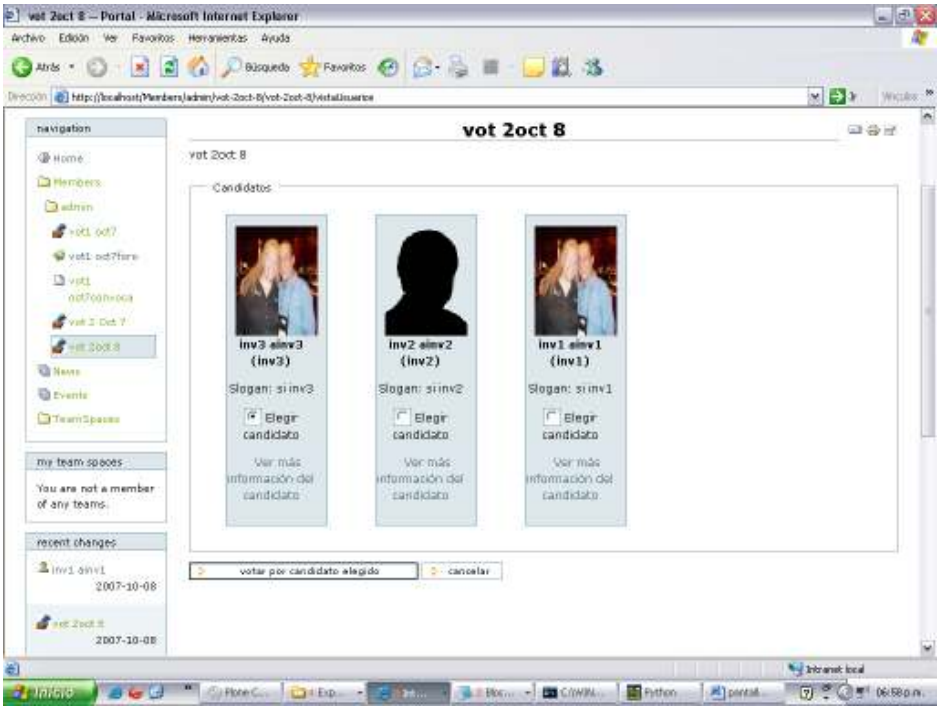


Votación

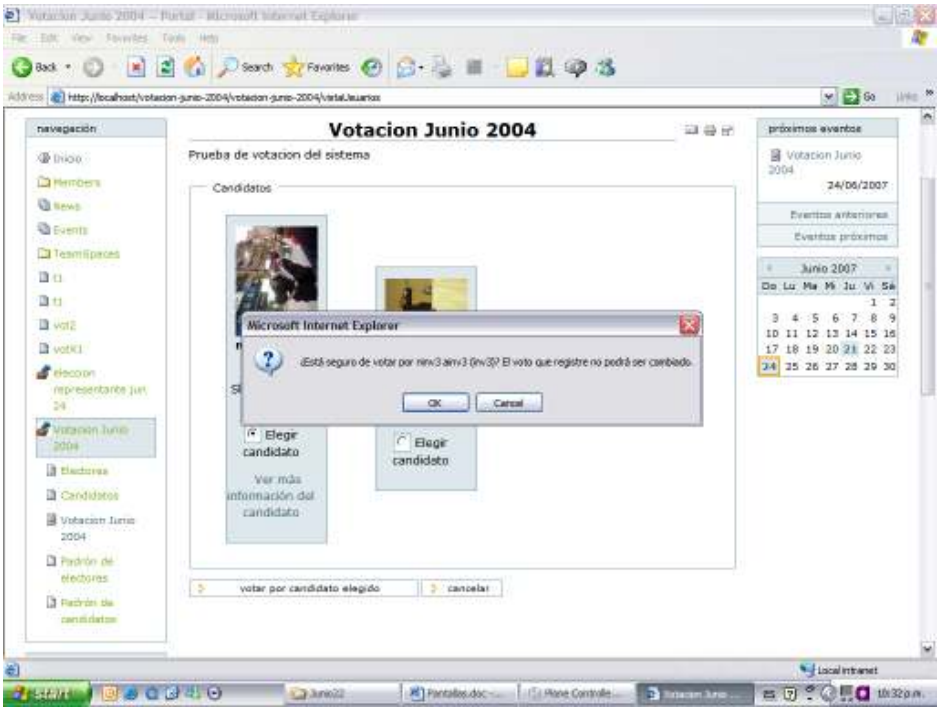
Una vez se llegue a la fecha y hora de inicio de la votación, los usuarios que ingresen al evento de la votación podrán escoger la opción de "Votar" en la ventana de "Estado general del proceso"



Después aparece una pantalla para seleccionar uno los candidatos por los cuales se puede votar:



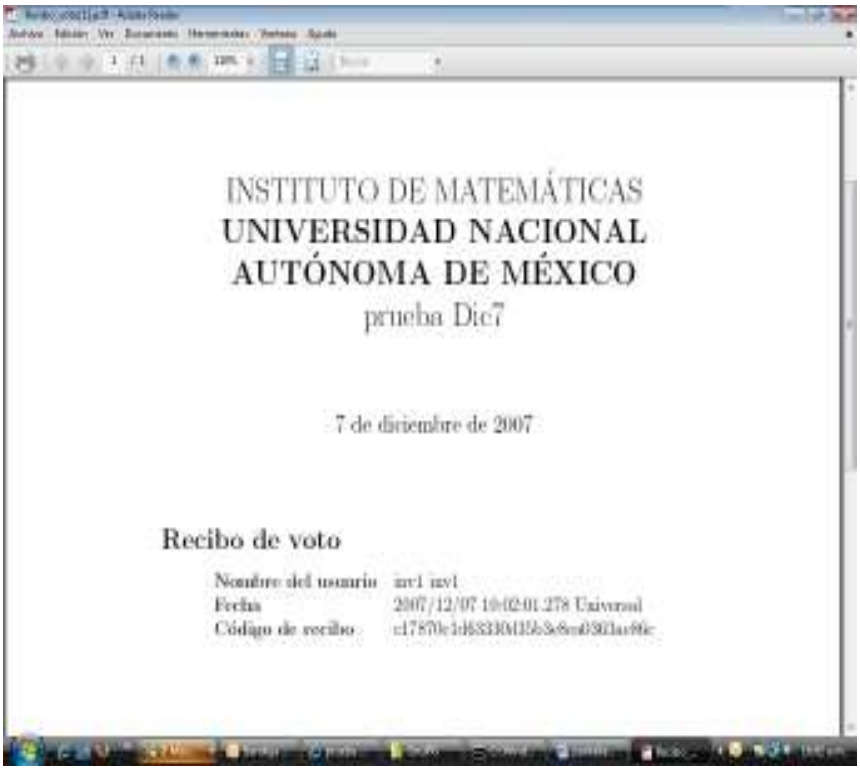
Deberá confirmar si es la opción de voto que realmente desea realizar:



Después deberá aparecer un mensaje de "Voto ha sido registrado correctamente" y aparecerá un Recibo de Voto, este número aparecerá en el escrutinio final y le permitirá al elector verificar que su voto fue apropiadamente contado.

El recibo del voto puede ser abierto para imprimirse en el momento o guardarse para ser impreso después por parte del elector:

El recibo de voto tendrá el siguiente formato, tipo pdf:



En el caso de que intente votar de nuevo aparecerá un mensaje "Usted ya votó"

Resultados de la Votación

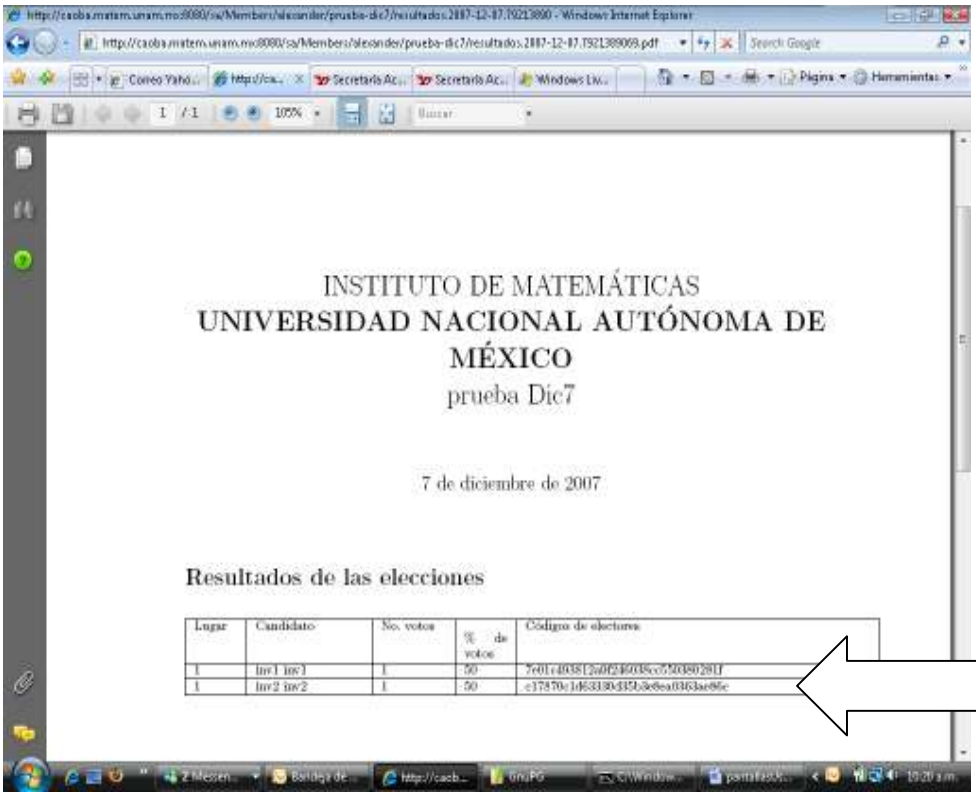
Después de que se termine la votación y se haya realizado el escrutinio se podrán consultar los resultados de la misma, mediante la opción de "Ver resultados de las elecciones" en la pantalla de ESTADO GENERAL:

En los resultados aparecerán el número total de votos, el porcentaje de los votos por candidato y el detalle de los recibos de votos incluidos en el respectivo conteo, si así lo configuró el dueño de la votación, se podrán también visualizar los números de recibos de votos duplicados en el caso de que se repita el código de un candidato y de los inválidos por no corresponder a un código de candidato válido.

Se puede también obtener el archivo de resultados en formato pdf, firmado por la comisión de vigilancia del proceso, mediante la opción "Para obtener resultados de las elecciones como un archivo firmado con GPG por la comisión de vigilancia de clic aquí" en la pantalla de ESTADO GENERAL DEL PROCESO

Aparecerá una pantalla con la firma del archivo de resultados y una liga para accederlo:

El resultado es un pdf como el siguiente, en el cual podrá verificar con su recibo de voto, si efectivamente fue apropiadamente contado:



INSTITUTO DE MATEMÁTICAS

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

prueba Dic7

7 de diciembre de 2007

Resultados de las elecciones

Lugar	Candidato	No. votos	% de votos	Código de elección
1	inv1 inv1	1	50	7e01c4038f2a0f246038cc5203802f1f
1	inv2 inv2	1	50	c17470c1d63330435b3e6ea0363ae86c

Apéndice IV - Manual de funciones claves de GnuPG

1. Utilización del GnuPG por parte de la comisión de vigilancia

1) Generación de un par de claves

Después de instalar GnuPG en su equipo, debe generar un par de claves privada y pública, la cual debe cumplir con las siguientes condiciones de seguridad: de al menos 2048 bits, que sea tipo DSA / El Gamal para que pueda ser usada para cifrar además de verificar firma y que expire después de que se terminen las elecciones. Es importante que especifique sus datos de identificación, email y comentarios para que después pueda determinar que par de llaves utilizará para alguno de los procesos que se requiere dentro de la votación

Debe ejecutar desde la línea de comando / Prompt (si está en Windows deberá dar Inicio-Ejecutar y escribir cmd), pasándose a la carpeta donde tiene instalado el GnuPG (mediante change directory – cd) la siguiente instrucción:

gpg --gen-key

De acuerdo con lo sugerido antes deberá registrar [41]:

- *Clase de llave: escoger 1. DSA and el Gamal, que es la que viene por defecto y que permitirá firmar y cifrar.*

```
evolution:~# gpg --gen-key
gpg (GnuPG) 1.4.1; Copyright (C) 2005 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: directory `/root/.gnupg' created
gpg: creado un nuevo fichero de configuración `/root/.gnupg/gpg.conf'
gpg: AVISO: las opciones en `/root/.gnupg/gpg.conf' no están aún activas en esta ejecución
gpg: anillo `/root/.gnupg/secring.gpg' creado
gpg: anillo `/root/.gnupg/pubring.gpg' creado
Por favor seleccione tipo de clave deseado:
   (1) DSA and Elgamal (default)
   (2) DSA (sólo firmar)
   (5) RSA (sólo firmar)
Su elección: 1
```

- *Seleccionar el tamaño de la clave de al menos 2048 bits*

```
DSA keypair will have 1024 bits.
ELG-E keys may be between 1024 and 4096 bits long.
What keysize do you want? (2048) 2048
El tamaño requerido es de 2048 bits
```

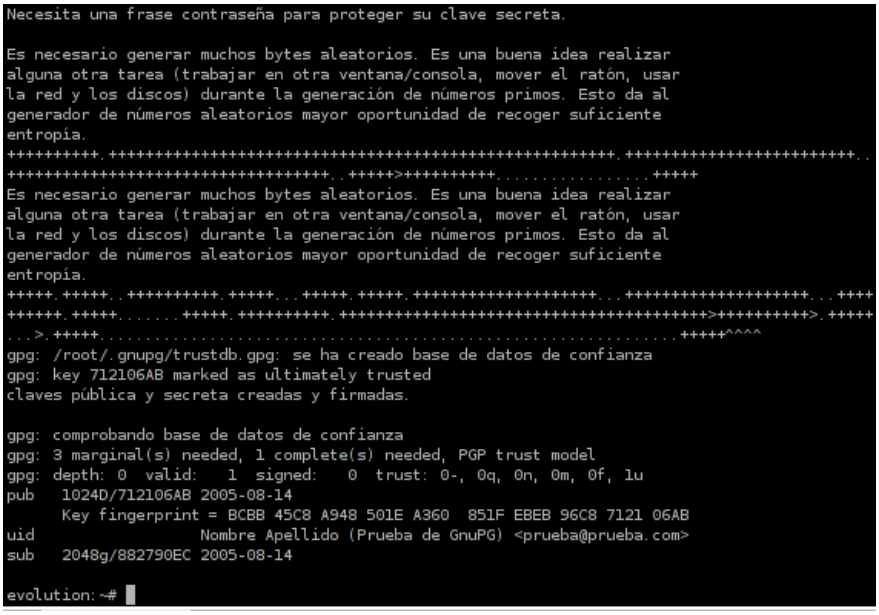
- *Registrar el número de años, meses, semanas o días en la que expirará la clave, tenga en cuenta que esté disponible sólo cierto tiempo después de la finalización de las elecciones para atender algún reclamo.*

```
Por favor, especifique el período de validez de la clave.
         0 = la clave nunca caduca
       <n>  = la clave caduca en n días
       <n>w = la clave caduca en n semanas
       <n>m = la clave caduca en n meses
       <n>y = la clave caduca en n años
¿Validez de la clave (0)? 0
Key does not expire at all
Is this correct? (y/N) y
```

- *Especifique que está correcto "y" o realice las correcciones que requiera*
- *Registre su nombre: ej. Miembro Comision Oct19, el cual debe ser diferente para cada par de claves que genere.*
- *Digite su dirección de correo electrónico.*
- *Digite un comentario que lo identifique.*
- *Seleccione O-Okay si todo está bien.*

```
You need a user ID to identify your key; the software constructs the user ID
from the Real Name, Comment and Email Address in this form:
    "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Nombre y apellidos: Nombre Apellido
Dirección de correo electrónico: prueba@prueba.com
Comentario: Prueba de GnuPG
Ha seleccionado este ID de usuario:
    "Nombre Apellido (Prueba de GnuPG) <prueba@prueba.com>"

¿Cambia (N)ombre, (C)omentario, (D)irección o (V)ale/(S)alir? V
```

- *Registre su contraseña con la cual protegerá su clave privada, ésta debe combinar números, letras y caracteres especiales, tener una longitud mínima de 8. Es buena idea reproducir mp3, mover el ratón o presionar teclas para que se generen números aleatorios y se creen antes las claves.*



2) *Exportación Clave Pública*

Una vez generado su par de claves, debe exportar su clave pública a un archivo plano, para que la pueda registrar dentro del proceso de configuración de las elecciones.

Para realizar esta actividad debe ejecutar el siguiente comando:

gpg --export -a "nombre" > archivo_salida.asc

Donde:

- *archivo_salida.asc: será el nombre del archivo que tendrá que editar para sacar su llave pública y registrarla en la opción de configuración de la votación.*
- *nombre: corresponde al nombre real que identifica en forma única al par de claves que generó. En el caso que tenga varios pares de claves generadas debe usar un nombre real diferente.*

3) *Firma de documentos oficiales (esta opción está integrada al menú de opciones que aparece al utilizar el script herramientaVotos)*

Para firmar un documento oficial de la votación, éste debe guardarse en el equipo donde se encuentra la clave privada de la comisión de vigilancia y después ejecutar el siguiente comando:

gpg –output archivo_salida.fir -u "nombre" –b –a –s archivo_a_firmar

Donde:

- *archivo_salida.fir: será el nombre del archivo donde quedará la firma digital en texto plano, este texto debe copiarse con un editor*

y pegarse en el campo de "Firma con GPG" en el producto de votaciones del sitio Plone. Para poder generar esta firma la comisión debe digitar su contraseña.

- *nombre: corresponde al nombre real que identifica en forma única al par de claves que generó. En el caso que tenga varios pares de claves generadas debe usar un nombre real diferente.*
- *Archivo_a_firmar: es el nombre con el que quedó guardado el archivo que se sacó del sitio plone de las votaciones para ser firmado.*

4) ***Descifrado de Archivos (esta opción está integrada al menú de opciones que aparece al utilizar el script herramientaVotos)***

Para descifrar un archivo que fue cifrado con la clave pública de la comisión de vigilancia debe utilizarse su clave privada, por lo que el archivo debe guardarse en el mismo equipo donde se encuentre almacenada ésta, y se debe ejecutar el siguiente comando:

gpg --output archivo_descifrado -u "nombre" --decrypt archivo_cifrado -u "nombre"

Donde:

- *archivo_descifrado: nombre del archivo que queda la información descifrada después de que la comisión ingrese su contraseña.*
- *nombre: corresponde al nombre real que identifica en forma única al par de claves que generó. En el caso que tenga varios pares de claves generadas debe usar un nombre real diferente.*
- *Archivo_a_firmar: es el nombre con el que quedó guardado el archivo que se sacó del sitio plone de las votaciones para ser firmado.*

5) ***Segunda Ronda de Descifrado de Votos (esta opción está integrada al menú de opciones que aparece al utilizar el script herramientaVotos)***

Para poder realizar la segunda ronda de descifrado de los votos lo primero que debe asegurarse es que la comisión reciba del administrador de la votación el archivo de salida resultante de la primera ronda de descifrado, que en forma estándar se debe denominar UrnaRonda1. Luego lo debe guardar en el equipo donde se encuentra su llave privada.

Después se debe descargar el script de Python descifradorVotos *(su ruta de instalación debe agregarse a la variable PATH del sistema) en la carpeta donde guardó el archivo de salida de la primera ronda de descifrado. Se debe verificar con que extensión queda guardado el script, si como .py o como .txt, pues en algunas instalaciones de windows le deja esta última extensión.*

Finalmente en la carpeta donde se tiene el archivo de la urna después de la primera ronda de descifrado, el script descifrador de votos y la clave privada de la comisión de vigilancia se debe ejecutar el siguiente comando:

Python descifradorVotos.extension contraseña Salida_primera_ronda votos_descifrados

Donde:

- *Extensión: debe ser py o txt en algunas instalaciones de Windows, sólo basta con verificar el nombre del archivo después de ser descargado.*
- *Contraseña: corresponde a la contraseña de la comisión de vigilancia.*
- *Salida_primera_ronda: nombre del archivo con la urna de los votos después de la primera ronda de descifrado por parte del administrador, su nombre estándar será UrnaRonda1.*
- *Votos_descifrados: Este es un archivo .zip con un archivo por cada voto descifrado. Este archivo debe subirse como segundo parámetro del escrutinio de las votaciones.*

2. Utilización del GnuPG por parte del Administrador de las votaciones

1) *Generación de un par de claves tal como se especifica en el punto 1 de la ayuda de usuario para el uso de GnuPG para la comisión de vigilancia*
2) *Exportación de la clave pública tal como se especifica en el punto 2 de la ayuda de usuario para el uso de GnuPG para la comisión de vigilancia*

3) Primera Ronda de Descifrado de Votos (esta opción está integrada al menú de opciones que aparece al utilizar el script herramientaVotos)

Para poder realizar la primera ronda de descifrado de los votos lo primero que debe realizarse es sacar del sitio Plone la urna con los votos cifrados y guardarla en el equipo donde se encuentra la llave privada del administrador de la votación.

Después se debe descargar el script de Python descifradorVotos *(su ruta de instalación debe agregarse a la variable PATH del sistema) en la carpeta donde se descargó la urna con los votos cifrados. Se debe verificar con que extensión queda guardado, si como .py o como .txt, pues en algunas instalaciones de windows le deja esta última extensión.*

Finalmente en la carpeta donde se tiene el archivo de la urna con los votos cifrados, el script descifrador de votos y la clave privada del administrador se debe ejecutar el siguiente comando:

Python descifradorVotos.extension contraseña archivo_votos_cifrados archivo_ronda1

Donde:

- *Extensión: debe ser py o txt en algunas instalaciones de Windows, sólo basta con verificar el nombre del archivo después de ser descargado.*
- *archivo_votos_cifrados: nombre del archivo con la urna de los votos cifrados que como se explicó antes se descarga desde la opción de escrutinio de la votación.*
- *Contraseña: corresponde a la contraseña del administrador de la votación.*
- *Archivo_ronda1: se recomienda que este archivo se le llame siempre UrnaRonda1 por estandarización. Este archivo aún está cifrado con la clave pública de la comisión por lo que el administrador se la podría enviar por email u otro medio más seguro, para que la comisión realice la segunda ronda de descifrado.*

Apéndice V – Implementación de un producto en Plone.

En este apéndice se describen primero los aspectos técnicos relacionados con la plataforma de Plone y las herramientas de soporte usadas para la implementación del sistema de votación electrónica.

Es importante mencionar que lo descrito a continuación parte del supuesto de que el lector conoce las características generales del sistema de administración de contenido Plone [6], la base de datos Zope sobre la que opera y el lenguaje de programación Python.

Herramientas de implementación utilizadas

1. Diseño de Arquetipos

Los arquetipos son un marco de trabajo diseñado para facilitar la construcción de aplicaciones para Plone, su principal finalidad es proveer una opción común para construir objetos de contenido, basándose en la definición de esquemas.

Su característica principal es que permite crear tipos de contenido basados en esquemas, donde un esquema es una secuencia de campos.

Para crear un arquetipo se debe definir un esquema, asignárselo a una clase heredada de BaseContent, BaseFolder o BaseBTreeFolder.

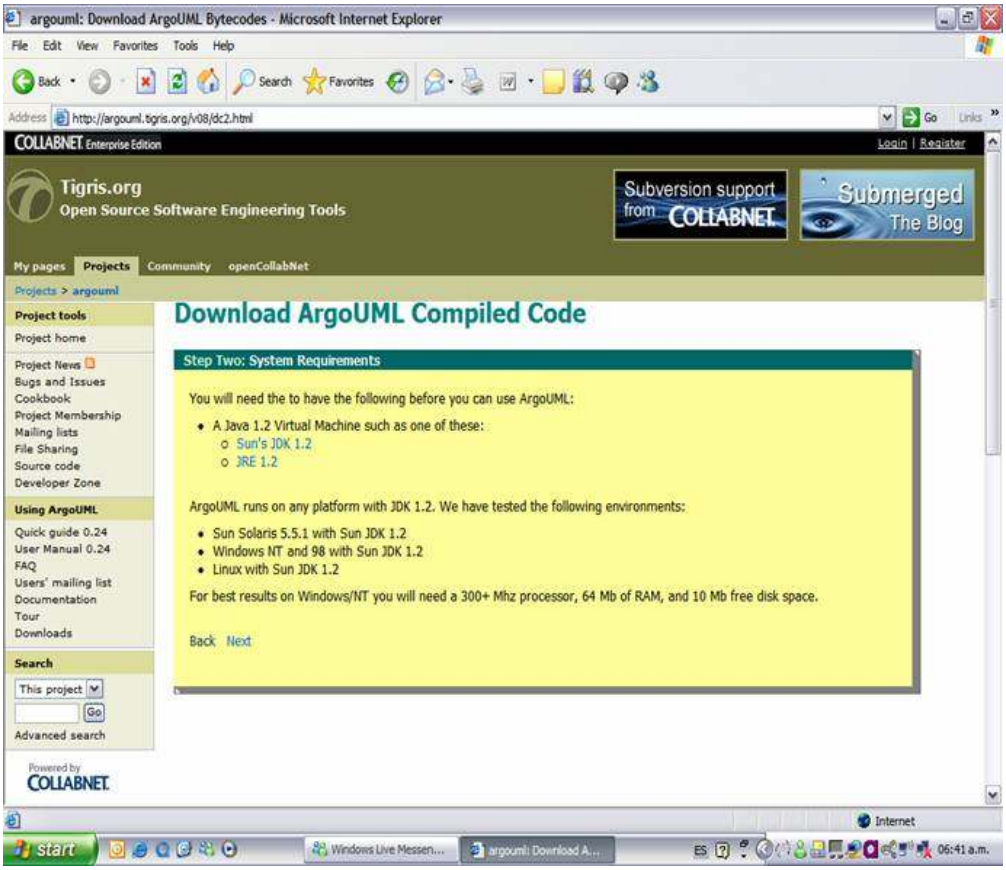
El proceso seguido para el diseño de los arquetipos básicos del sistema propuesto involucra la utilización de herramientas adicionales:

- *ArgoUML*
- *ArchGenXML*

ArgoUML

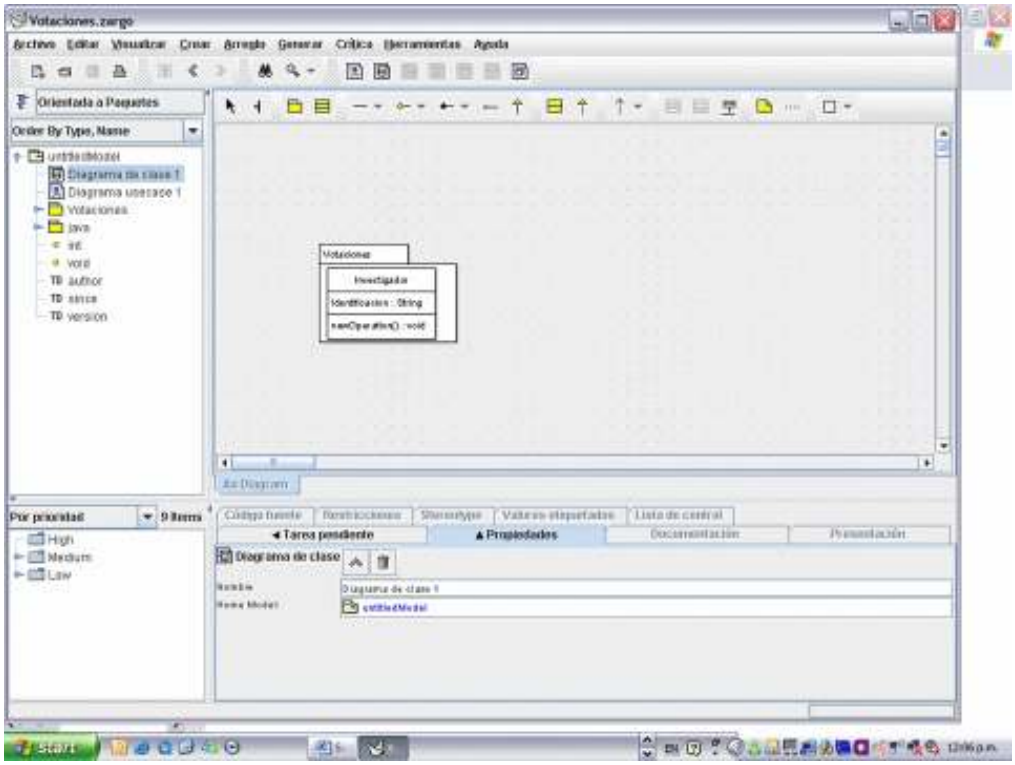
Es una herramienta utilizada en el modelaje de sistemas, mediante la cual se realizan diseños en UML llevados a cabo en el análisis y pre-diseño de Sistemas de Software.

Esta herramienta puede descargarse del sitio http://argouml.tigris.org y requiere tener instalada una máquina virtual de Java 1.2



Después de que se haya instalado esta herramienta, se puede ejecutar en el caso de Windows dando doble clic sobre el ícono de argouml-mdr.jar y en el caso de Linux con sh /xx/argouml/argouml.sh

Por cada arquetipo que se va a construir se define un diagrama de clase con sus atributos y métodos, tal como se muestra en la figura siguiente, generándose un archivo .zargo:



ArchGenXML

Es un generador de código para aplicaciones CMF/Plone (Productos) basado framework de Arquetipos.

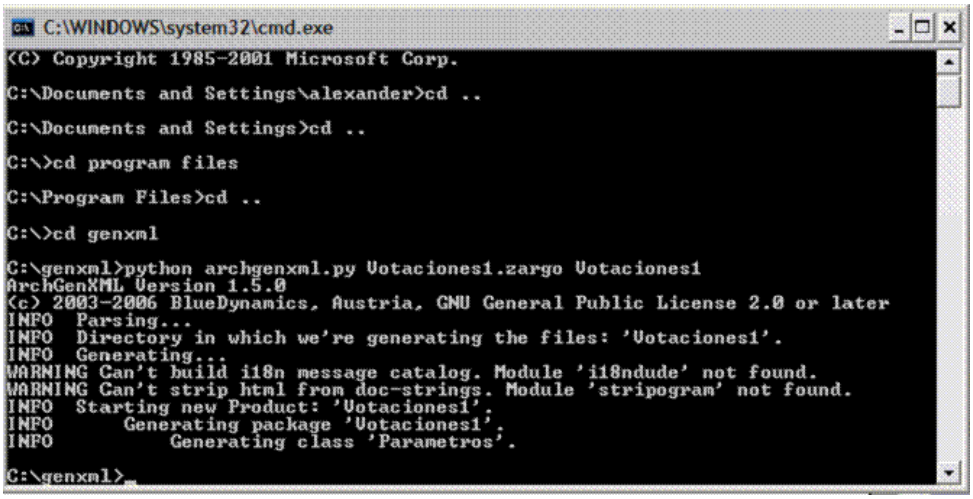
*Interpreta modelos UML en XMI-Format (.xmi, **.zargo**, .zuml), creados con aplicaciones como ArgoUML, Poseidon o ObjectDomain.*

Puede ser descargado de la página www.plone.org/products y requiere tener instalado al menos Plone 2 y el interpretador de Python 2.3

En el caso de Linux se ejecuta de la siguiente forma:

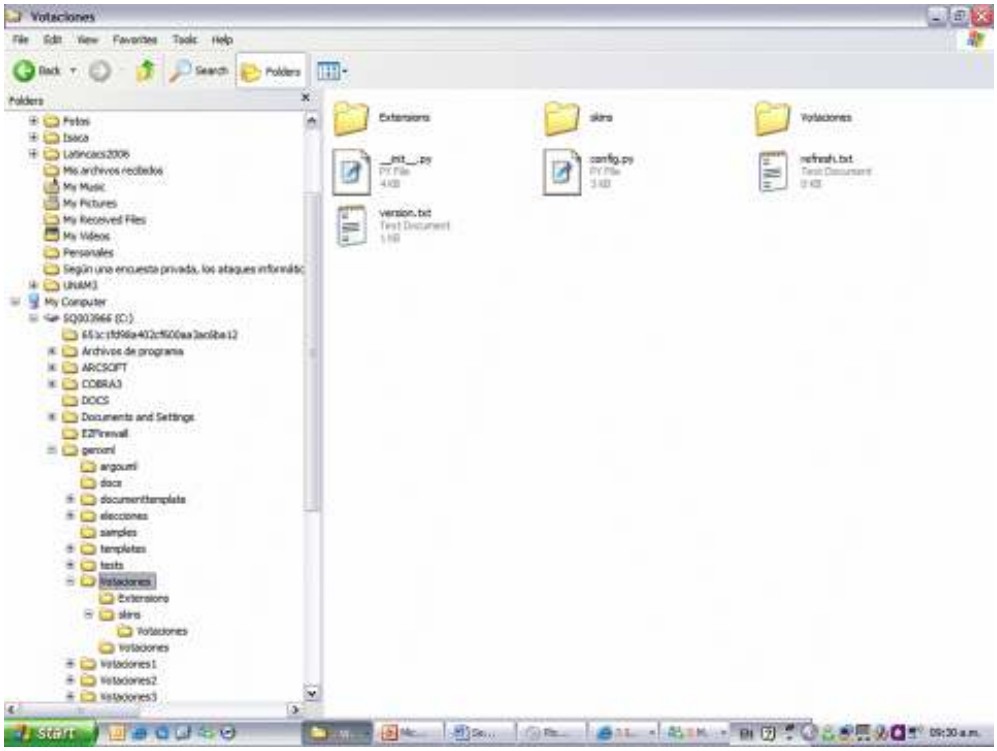
archgenxml.py –o salida entrada.zargo

En el caso de Windows se ejecuta de la siguiente forma:



2. Estructura de un producto para Plone

Después de utilizar la herramienta de ArchGenXML se genera la estructura de directorios requerida para subir un nuevo producto a Plone, copiando el directorio a la carpeta de Data/Products:



A continuación se debe agregar en el archivo de instalación del producto Plone, el nombre de la clase que se creó con ArgoUML:

Install.py en Extensions

```
# enable portal_factory for given types
    factory_tool = getToolByName(self,'portal_factory')
    factory_types=[
        "Investigador","Parametros",
        ] + factory_tool.getFactoryTypes().keys()
factory_tool.manage_setPortalFactoryTypes(listOfTypeIds=f
actory_types)
```

Posteriormente, se debe agregar también en el archivo de configuración inicial del producto Plone, el nombre de la clase que se creó con ArgoUML:

__init__.py en la subcarpeta del producto:

```
# Classes
import Investigador
import Parametros
```

A continuación se debe modificar la referencia al producto en cada archivo Python de cada clase generada:

Investigador.py en la subcarpeta del producto:

```
from AccessControl import ClassSecurityInfo
from Products.Archetypes.atapi import *
from Products.Votaciones.config import *
##code-section module-header #fill in your manual code
here
##/code-section module-header
schema = Schema((
    StringField(
        name='Identificacion',
        required=1,
        widget=StringWidget(
            label='Identificacion',
```

```
label_msgid='Votaciones_label_Identificacion',
            i18n_domain='Votaciones',
            size=10,
        ),
        version="1.0",
        since="November 2006",
        author="Alexander Zapata"
    ),
```

Finalmente se debe reiniciar Zope y agregar el producto en Plone, tal como se muestra en la pantalla de la página siguiente. En este caso el producto creado fue Votaciones.

En el caso de Linux basta con copiar el producto a la carpeta Products, reiniciar Zope y si no hay dependencias ya se puede instalar el producto y utilizar

./zopectl start

./zopectl restart

http://localhost:8080/sa

3. Widgets

Son pequeñas aplicaciones o programas, usualmente presentados en archivos o ficheros pequeños que son ejecutados por un motor de Widgets o Widget Engine [7].

Entre sus objetivos están los de dar fácil acceso a funciones frecuentemente usadas y proveer de información visual.

El modelo de mini aplicaciones de widgets, es muy atractivo por su relativamente fácil desarrollo: muchos de los widgets, pueden ser creados con unas cuantas imágenes y con pocas líneas de código, en lenguajes que van desde XML pasando por JavaScript a Perl y C# entre otros.

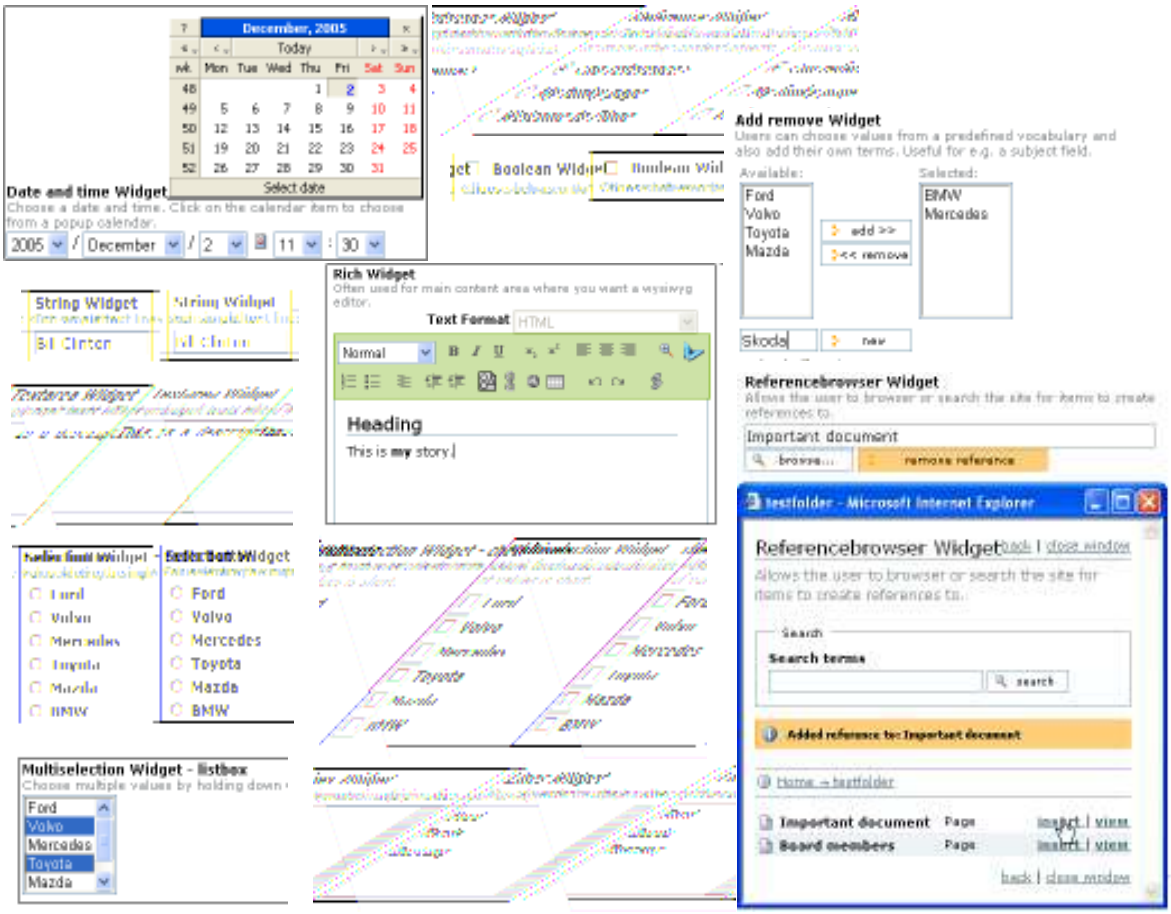
Los tipos de widgets utilizados son los siguientes:

- *BooleanField (falso o verdadero)*
- *DateTimeField (fechas con hora)*
- *FileField (archivos)*
- *FixedPointedField (números de punto fijo)*
- *FloatField (números de punto flotante)*
- *ImageField (imágenes)*
- *IntegerField. (números enteros)*
- *LinesField (Listas)*
- *ReferenceField (referencias entre objetos)*
- *StringField (cadenas de caracteres, optimizado para menos de 100)*
- *TextField (cadenas de caracteres, optimizado para más de 100)*
- *ComputedField (de sólo lectura, su valor es calculado por medio de una expresión de python).*

A cada field se le puede asociar un widget, que define la forma en que será mostrado dicho campo

Se puede crear un nuevo widget o field si ninguno satisface sus necesidades.

Los más comunes son los siguientes:

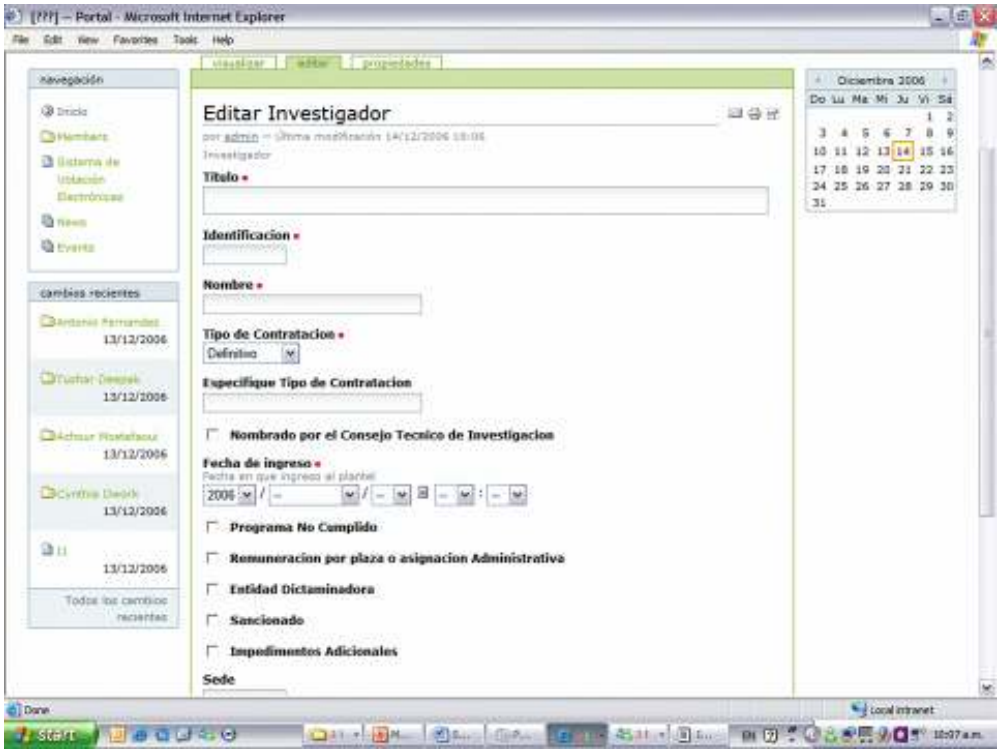


Algunos ejemplos de la implementación de estos widgets en código Python son los siguientes:

```
StringField(
        name='Nombre',
        required=1,
        widget=StringWidget(
            label='Nombre',
            label_msgid='Votaciones_label_Nombre',
            i18n_domain='Votaciones',
        ),
        version="1.0",
        since="November 2006",
```

```
        author="Alexander Zapata"),


StringField('TipoInvestigador',

    required=1,

    vocabulary = [[1,"Definitivo"],[2,"Por Contrato"],
[3,"Interino"], [4,"Otro"]],

    widget = SelectionWidget(label = "Tipo de
Contratacion", format = "select")

    ),

 BooleanField('Nombrado por el Consejo Tecnico de
Investigacion',

                searchable=1,

                default='',

                widget=BooleanWidget(format='select',),

                ),

DateTimeField('fecha_ingreso',

          required=1,

           widget=CalendarWidget(

               label='Fecha de ingreso',

               label_msgid='label_fecha_ingreso',

               description='Fecha en que ingreso al
plantel',

               description_msgid='help_fecha_ingreso',

               format="%d/%m/%y",

           ),

       ),
```

Un arquetipo básico generado con el proceso anterior, se muestra en la pantalla siguiente, es importante mencionar que este arquetipo se creó en la fase inicial del proyecto por lo que no forma parte del producto de votaciones actual:



Los widgets deben ser utilizados dependiendo del tipo del campo, y la asociación permitida es la siguiente:

- *BooleanField. LabelWidget y BooleanWidget*
- *ComputedField. LabelWidget y ComputedWidget*
- *DateTimeField. LabelWidget y CalendarWidget*
- *FileField. LabelWidget y FileWidget*
- *FixedPointField. LabelWidget y DecimalWidget*
- *ImageField. ImageWidget y LabelWidget*
- *IntegerField. LabelWidget e IntegerWidget*
- *LinesField. LinesWidget, LabelWidget, MultiSelectionWidget, PickListWidget, InOutWidget y KeywordWidget*
- *ReferenceField. ReferenceWidget, LabelWidget, InOutWidget*
- *StringField. StringWidget, TextAreaWidget, SelectionWidget, IdWidget, PasswordWidget, VisualWidget, EpozWidget*
- *TextField. TextAreaWidget, RichWidget*

Es importante mencionar que se debe definir un esquema para el objeto, tal como se muestra en el ejemplo siguiente:

```
from Products.Archetypes.public import *

schema= BaseSchema.copy() + Schema((

    StringField('edad',

        required=0,

        widget=StringWidget(label="Edad",

                    description="Escriba su edad",

                    ),

        ),

    StringField('sexo',

        required=1,

        widget=SelectionWidget(label="Genero'',

                    description="Seleccione un genero",

                    ),

        vocabulary=DisplayList( (('m', 'Masculino'), ('f', 'Femenino')) ),

        ),

    .........

    ))
```

Finalmente, se deben incluir descripciones que expliquen las características de los campos del arquetipo, tal como se muestra a continuación:

```
from Products.Archetypes.public import *

DateTimeField('FechaInicioVotaciones',

    widget=CalendarWidget(

      description="Fecha y hora en la que iniciaran las elecciones. A partir de este momento los usuarios pueden empezar a votar. Si no se selecciona la hora se tomara¡ por default las 00:00 hrs.",

      label='Fecha de inicio de las elecciones',
```

),

required=1,

),

DateTimeField('FechaFinVotaciones',

widget=CalendarWidget(

description="Fecha y hora en la que finalizaran las elecciones. A partir de este momento los usuarios ya no podrán votar. Si no se selecciona la hora se tomara por default las 00:00 hrs.",

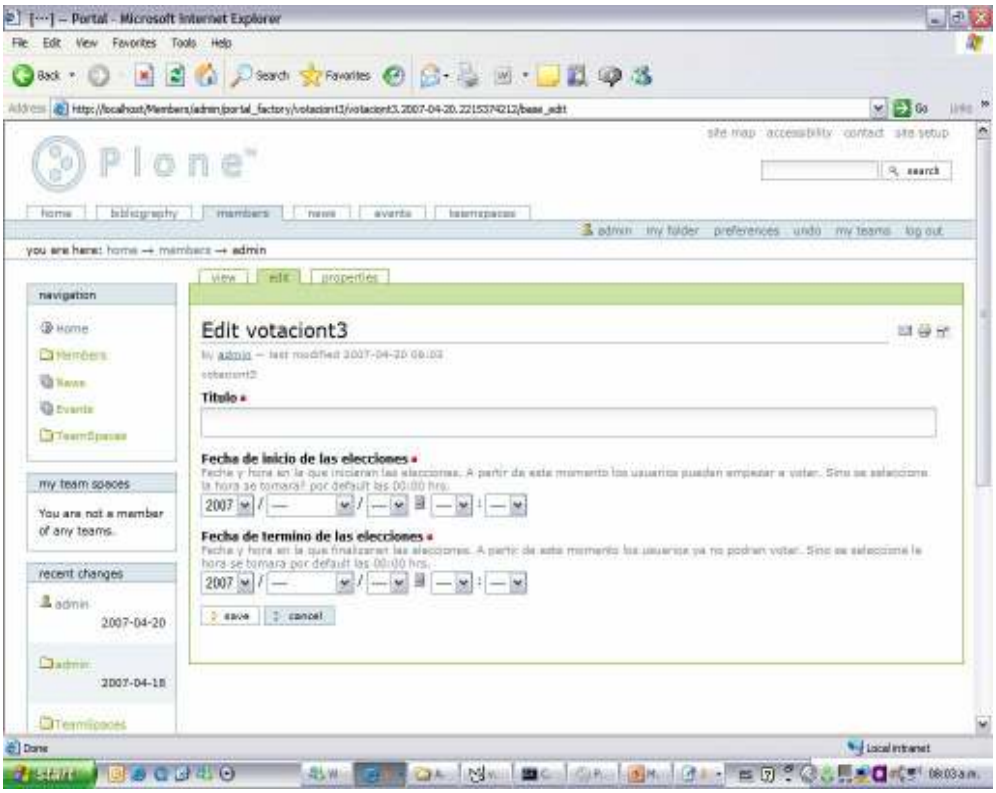
label='Fecha de termino de las elecciones',

),

required=1,

),

Un arquetipo básico generado con el proceso anterior, es el siguiente:



4. Vistas

Plone trae por default varias vistas que son muy útiles pero no siempre son suficientes.

Existen dos lenguajes para crear vistas en Plone DTML y ZPT, de los cuáles ZPT es el más utilizado.

ZPT Zope Page Template utiliza mucho las macros, que son pedazos de código reutilizable para diferentes vistas.

Las macros utilizan algo llamado slots (ej. Calendario, main, búsqueda, noticias, etc) que permiten redefinir sólo una parte de la macro.

Las etiquetas básicas de ZPT son las siguientes:

- *<ETIQUETA tal:content="expresion" > Algo </ETIQUETA>*

  Reemplaza Algo por lo que haya en una expresion

- *<ETIQUETA tal:replace="expresion" > Algo </ETIQUETA>*

  Reemplaza hasta la etiqueta por lo que haya en una expresion

- *<ETIQUETA tal:define="var1 expr1; var2 expr2" >*

  Algo

  </ETIQUETA>

  Asigna expresiones a variables que se pueden usar en el ámbito de la etiqueta

- *<ETIQUETA tal:attributes="at1 expr1; at2 expr2; atn exprn" >*

  Algo

  </ETIQUETA>

  Redefine el valor de un atributo, ej href, image, value

- *<ETIQUETA tal:condition="expresion" > Algo </ETIQUETA>*

  Si se cumple la condición se ejecuta lo que esta dentro de la etiq

En el caso de que se utilicen macros se requieren las siguientes etiquetas tipo "metal":

metal:ETIQUETA1 ***define-macro****="NOMBRE_MACRO">*

* Algunas líneas de html *

*<metal:ETIQUETA2 **define-slot**=”NOMBRE_SLOT”>*

* algo que hace el slot por default *

</metal:ETIQUETA2>

* más código html *

</metal:ETIQUETA1>

*<metal:ETIQUETA3 **use-macro**= ”here/nombre_archivo/macros/NOMBRE_MACRO”>*

*<metal:ETIQUETA4 fill-slot=”NOMBRE_SLOT”> **redefinir-slot***

* Este slot se pudo no haber utilizado *

</metal:ETIQUETA2>

</metal:ETIQUETA3>

Regularmente en Plone se utiliza la macro “main_template/macros/master”. Esta macro tiene varios slots, pero el principal se llama “main” y su utilización se realiza de la siguiente forma:

*<metal:master **use-macro**="here/main_template/macros/master">*

*<metal:main fill-slot="**main**">*

Aquí ponemos texto en la vista principal

</metal:main>

</metal:master>

Este archivo se guarda dentro de la carpeta skins que ya está previamente registrada en el Producto.

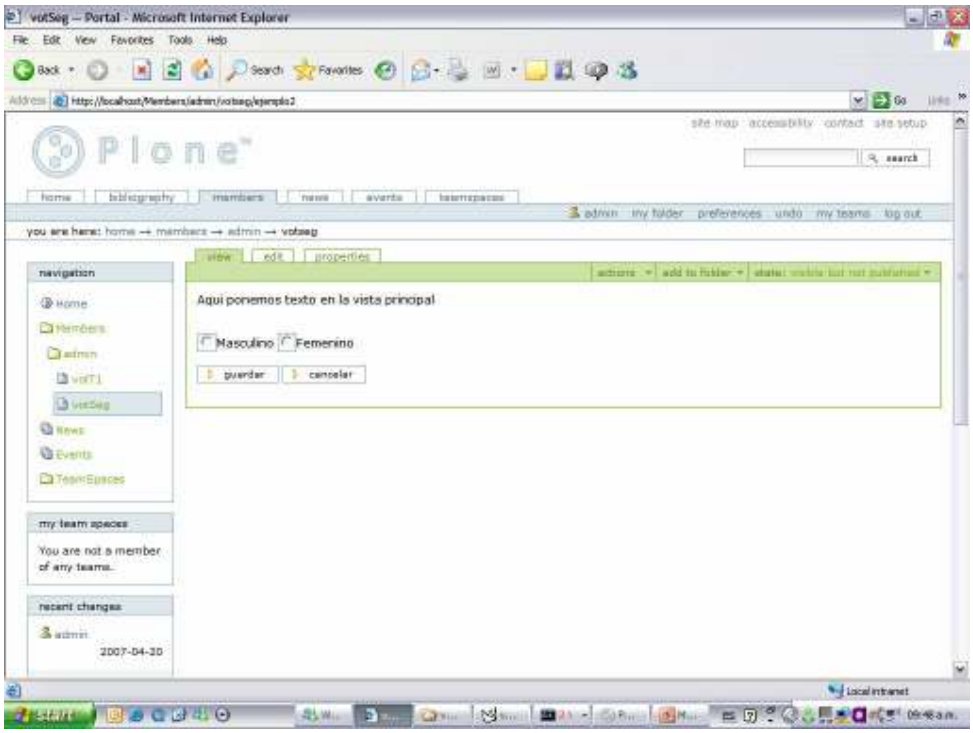
La funcionalidad de las vistas se complementa con el control de metadatas y validadores, los cuales tienen diferentes extensiones dentro de los archivos del producto.

Para comprender mejor la forma cómo interactúan las vistas los metadatas y los validadores se presenta el siguiente ejemplo que no fue utilizado en el producto de votaciones, pero satisface la siguiente funcionalidad: Vista controlada por un metadata que permita escoger el género y que permita aceptar/cancelar, en el caso de que no se escoja el género y se dé aceptar debe desplegarse una vista de error.

La vista es la siguiente:

```
<metal:master use-macro="here/main_template/macros/master">
  <metal:main fill-slot="main">
     Aquí ponemos texto en la vista principal
<form name="procesoSeleccion" enctype="multipart/form-data"
           action="#" method="post"
          tal:attributes="action string:${here/getId}/ $
{template/getId}">
    <input type="hidden" name="form.submitted" value="1">
    <br>
    <br>
     <input name="genero" value="masculino" type="radio"> Masculino
     <input name="genero" value="femenino" type="radio"> Femenino
    <br>
    <br>
      <input class="context"
            type="submit"
            name="form.button.guardar"
            value="Guardar">
    <input class="context"
            type="submit"
            name="form.button.cancelar"
            value="Cancelar">
 </form>
</metal:main>
</metal:master>
```

La vista generada en plone vista.pt es la siguiente:



La metadata con vista.cpt.metadata que controla la funcionalidad de los botones guardar y cancelar de la vista anterior es la siguiente:

[default]

title=Edicion de electores

[validators]

validators..guardar=validagenero

[actions]

action.success..guardar=traverse_to:string:base_edit

action.success..cancelar=traverse_to:string:base_view

action.failure=traverse_to:string:error

El validador con extensión .py que complementa la funcionalidad de la vista anterior, mostrando un error en el caso que se de aceptar si escoger el género, es el siguiente:

Controller Python Script "validagenero"

##bind container=container

```
##bind context=context
##bind namespace=
##bind script=script
##bind state=state
##bind subpath=traverse_subpath
##parameters=
##title=Validate folder renaming
##
genero=context.REQUEST.get('genero',None)
if not(genero):
   state.set(status="failure", portal_status_message="error")
return state
```

Bibliografía

[1] Xenakis Alexandros. E-electoral Administration: Organizational Lessons Learned from the Deployment of E-voting in the UK*. International Teledemocracy Center, Napier University. 2005*

[2] Barbara Simmons. Electronic Voting Systems: the Good, the Bad, and the Stupid*. ACM QUEUE. 2004*

[3] David L. Dill, Schneier Bruce, Simons Barbara. Voting and Technology: Who Gets to Count Your Vote? *Communications of the ACM. 2003*

[4] Schneier Bruce. Voting and Security*. IEEE Security and Privacy. 2006.*

[5] Dill David, Mercuri Rebecca, Neumann Peter, Wallach Dan. Frequently Asked Questions about DRE Voting Systems. *VerifiedVoting.org.*

[6] Pelletier Michel and Shariff Munwar. Plone Live*. SourceBeat. 2006*

[7] Andy McKay. The definitive Guide to Plone. *Apress. 2006*

[8] Lotze Thomas, Theune Christian. Content Management with Plone*. GoseptPress. 2006*

[9] Smith Jeremy. Permissions in Plone*. Plone Users Group of Davis. 2007.*

[10] www.plone.org

[11] Bracho Carpizo Javier. Convocatoria ordinaria para elegir al representante del personal académico del instituto de matemáticas ante el consejo técnico de la investigación científica para el periodo 2006-2009*. IMATE. 2006.*

[12] Daltabuit Enrique, Hernández Leobardo, Mallén Guillermo, Vásquez José de Jesús. La Seguridad de la Información*. Universidad Nacional Autónoma de México. Limusa 2007.* Cap 4 y Cap 5.

[13] http://www.cca.org.mx/dds/cursos/estadistica/html/m11/desviacion_estandar.htm

[14] Astigarraga Eneko. Delphi Method*. Universidad de Deusto Facultad de CC.EE. y Empresariales.*

[15] http://www.qtic.ssr.upm.es/encuestas/delphi.htm

[16] Different views on e-voting security, Site officiel de l´Etat de Genève, p 1-11*. www.geneve.ch/evoting/english/secutiry.asp*

[17] Lambrinoudakis C, Tsoumas V, Karyda M, Ikonomopoulos S. Secure E-Voting The Current Landscape. Dept. of Information and Communication Systems*. University of the Aegean Karlovassi, Dept. of Informatics, Athens University of Economics and Business, p 1-19.*

*[19] Gómez R.*Votación electrónica, ¿una opción?. *ITESM. Septiembre 2006, p 1-3.*

[20] García-Zamora Claudia, Rodríguez-Henríquez Francisco, Ortiz-Arroyo Daniel. SELES: An e-Voting System for Medium Scale Online Elections*. Proceedings of the Sixth Mexican International Conference on Computer Science. 2005, p 1-2.*

[21] Sampigethaya Krishna, Poovendran Radha. A framework and taxonomy for comparison of electronic voting schemes*. Department of Electrical Engineering, University of Washington. November 2005, p 137-44.*

[22] Kiniry J, Cochran D, Tierney P. A Verification-Centric Realization of e-Voting*. School of Computer Science and Informatics, University College Dublin. May 2007, p 1.*

[23] Kiniry J, Cochran D, Fairmichael F. The KOA Remote Voting System: A Summary of Work To Date*. School of Computer Science and Informatics, University College Dublin; Patrice Chalin, Department of Computer Science and Informatics, Concordia University; Martjin Oostdijk and Engelbert Hubbers, Nijmegen Institute of Information and Computing Sciences, Radboud University Nijmegen. 2007, p 12-4.*

[24] Sistema de Información del Instituto de Matemáticas. *https://info.matem.unam.mx*

[25] Boyd C. A new multiple key cipher and an improved voting scheme. In: Advances in cryptology – EUROCRYPT '89. Springer-Verlag; 1990. p. 617–25.

[26] Sako K, Killian J. Receipt-free mix-type voting scheme – a practical solution to the implementation of a voting booth. In: Advances in cryptology – EUROCRYPT '95. LNCS, vol. 921. Springer-Verlag; 1995. p. 393–403.

[27] Chaum D. Secret-ballot receipts: true voter-verifiable elections IEEE Security & Privacy Magazine Feb 2004.

[28] Iverson KR. A cryptographic scheme for computerized general elections. In: Advances in cryptology – CRYPTO '91. LNCS, vol. 576. Springer-Verlag; 1992. p. 405–19.

[29] Schoenmakers B. A simple publicly verifiable secret sharing scheme and its applications to electronic voting. In: Advances in cryptology – CRYPTO '99. LNCS, vol. 1666. Springer-Verlag; 1999. p. 148–64.

[30] Lee B, Kim K. Receipt-free electronic voting scheme with a tamper-resistant randomizer. In: ICISC '02. LNCS, vol. 2587. Springer-Verlag; 2002. p. 389–406.

[31] Kiayias Aggelos, Yung Moti. Self-tallying elections and perfect ballot secrecy. In: Proceedings of public key cryptography, fifth international workshop on practice and theory in public key cryptosystems, PKC 2002. LNCS, vol. 2274. Springer-Verlag; 2002. p. 141–58.

[32] Fujioka A, Okamoto T, Ohta K. A practical secret voting scheme for large scale elections. In: Advances in cryptology – AUSCRYPT '92. LNCS, vol. 718. Springer-Verlag; 1993. p. 248–59.

[33] Okamoto T. Receipt-free electronic voting schemes for large scale elections. In: Proceedings of the workshop on security protocols '97. LNCS, vol. 1361. Springer-Verlag; 1997. p. 25–35.

[34] Lee B, Boyd C, Dawson E, Kim K, Yang J, Yoo S. Providing receiptfreeness in mixnet-based voting protocols. In: Proceedings of the ICISC '03, 2003. p. 261–74.

[35] Kiayias Aggelos, Yung Moti. The vector-ballot e-voting approach. In: Financial cryptography. LNCS, vol. 3110. Springer-Verlag; 2004. p. 72–89.

[36] Juels A, Jakobsson M. Coercion-resistant electronic elections. Cryptology ePrint Archive, Report 2002/165, <http://eprint. iacr.org/>; 2002.

[37] Shamir A, "How to share a secret", Communications of the ACM, 22(1), p 612–613, 1979

[38] Administración Electoral e implementación de sistemas de votación electrónica, http://www.slideshare.net/fbarrienmx/voto-electronico-y-calidad-de-la-democracia, Slide 4.

[39] Voutssás Juan, "SVE – Sistema de Votaciones Electrónicas de la UNAM", Dirección General de Cómputo Académico de la UNAM, 2007.

[40] Department of homeland security, Making Software Development Processes and Software Produced by Them—More Secure, Version 1.2, 2006. p. 2.

[41] http://www.lostscene.com/manuales/gnupg.php, GnuPG Básico.