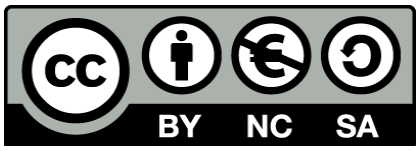
Proyecto Final Máster

a l'Institut Municipal d'Informàtica de Barcelona

Bases Teóricas y Herramientas para el Análisis y Comparación de Sistemas de Votación de Código Abierto

Área	**Màster Universitari en Programari Lliure**
Autor	Jordi Codina Lligoña
Tutor de Prácticas (UOC)	Rubén Mondéjar Andreu
Tutor externo (IMI)	Enrique Felez Zaera
Fecha	19/01/14



Nunca en mi vida había utilizado una herramienta, más con el tiempo, con trabajo, empeño e ingenio descubrí que no había nada que no pudiera construir, en especial, si tenía herramientas.
Robinson Crusoe (Daniel Defoe)

A mi implacable equipo de corrección, que ha leído incontables versiones y a mi infatigable censo de votantes que ha resistido un número infinito de consultas de prueba.

Muchas gracias,

Jordi Codina Lligoña

RESUMEN

Dentro del proceso de definición de las vías de participación en el PLAN DIRECTOR DE PARTICIPACIÓN CIUDADANA 2010-2015 del Ayuntamiento de Barcelona, se plantea la necesidad de fomentar mecanismos de participación directa y a, partir de esta necesidad, se plantea el desarrollo de una aplicación de Consultas Populares.

Dentro de los procesos participativos, las consultas a realizar pueden ser vinculantes y por tanto resulta de la mayor importancia fundamentar su desarrollo garantizando la corrección formal de todo el proceso.

Entre los requerimientos, especificados por en Instituto Municipal de Informática, el sistema de votación a plantear debe estar cubierto por alguna de las licencias consideradas libres o de código abierto. Una de las principales características de este tipo de aplicaciones es que resulta posible la utilización y modificación del código.

Como primer paso se definen los conceptos básicos implicados comenzando con la del propio concepto de e-Voting. A partir de las experiencias previas realizadas en nuestro entorno geográfico, es posible establecer que los estándares más utilizados son los generados por el Consejo de Europa, qué utilizaremos para concretar las diferentes fases de un Proceso de Consulta, los actores implicados así como las diferentes funciones a de cada uno de ellos para cada una de las fases.

Sobre esta base se analizan las diferentes alternativas existentes entre lo que se denominan Esquemas de Votación. Centrándonos en el escogido como más conveniente, el Esquema de Voto Oculto, se describen las características de seguridad que permiten dar validez a un proceso de consulta realizado por este medio. Así mismo se describen los principales conceptos teóricos implicados mediante una aproximación que permita su comprensión de forma intuitiva, especialmente en lo que se refiere a las técnicas criptográficas implicadas.

En tercer lugar se ha realizado una búsqueda de posibles aplicaciones que sirvan de punto de partida y se han hecho pruebas comparativas para las cuales se han adaptado las herramientas correspondientes a QSOS, relacionando los estándares especificados por el Consejo Europeo y las características de seguridad que deben cumplirse.

Índice de ilustraciones

1 Introducción

Como en otros ámbitos, las tecnológicas de la comunicación están modificando radicalmente el modo en que los ciudadanos se relacionan con las administraciones y organismos públicos.

Términos como e-Participación o e-Gobierno se han generalizado e implican una mayor participación además de una relación más directa y dinámica, así como la necesidad de crear las plataformas y sistemas que permitan esta relación.

En este sentido el Ayuntamiento de Barcelona aprobó en el Consejo Plenario de 1 de Octubre de 2010 el PLAN DIRECTOR DE PARTICIPACIÓN CIUDADANA 2010-2015, dónde se define, como uno de los derechos reconocidos, el derecho de Participación.

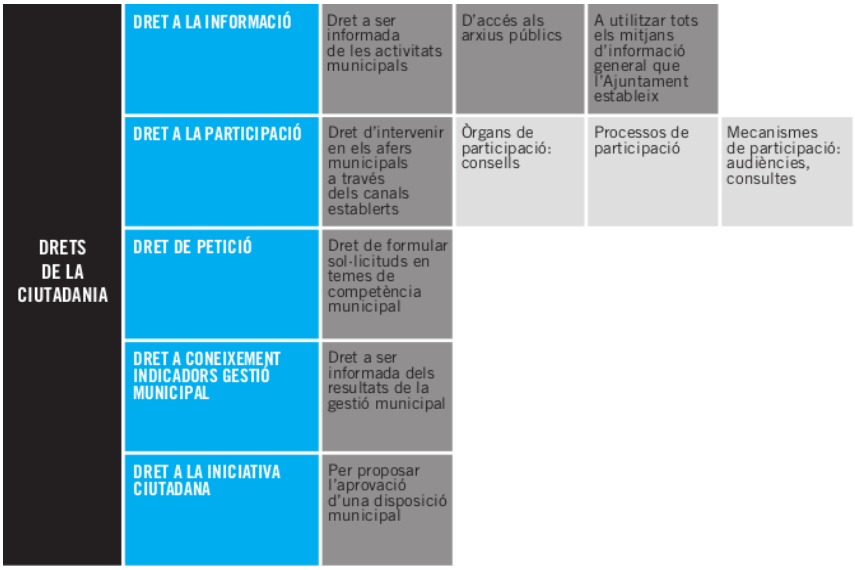


Ilustración 1: Derechos de la Ciudadanía recogidos en el Plan Director de Participación Ciudadana 2010-2015

Dentro de las conclusiones[1] se pueden encontrar los siguientes puntos a tener en cuenta:

- Existe un conocimiento deficiente por parte de la ciudadanía respecto a los órganos, los mecanismos y los canales de participación.
- La ciudadanía reclama ser escuchada, informada y consultada sobre los aspectos de la política municipal que le son más próximos.
- Existen dificultades en ampliar el abanico de la participación, tanto en lo que ser refiere a las asociaciones como a la ciudadanía en general.
- La participación municipal se percibe positivamente tanto en su vertiente asociativa como en su vertiente municipal.

Entre las finalidades expresadas en el plan figura la intención de facilitar mecanismos que fomenten la democracia directa[2].

1 Conclusions de la Diagnosi pag. 29
2 FINALITATS DEL PDMPC pag.31

1.1 Objetivos

En la propuesta de proyecto del IMI (Instituto Municipal de Informática) se fija como objetivo la definición, arquitectura y diseño de un sistema informático distribuido que permita la realización de consultas populares, garantizando la confidencialidad, seguridad, sincronía y posibilidad de acceso móvil.

Se solicita la elaboración teórica de un diseño para un sistema de consultas populares, teniendo en cuenta las diferentes interfaces de relación posibles (fijas, móviles) y análisis de todas las posibles consideraciones de seguridad y garantía de datos.

El sistema debe proveer la posibilidad de funcionamiento asíncrono, por lo que se encuadra en la categoría de un sistema distribuido que permita la utilización de alta disponibilidad y distribución de recursos, es decir, que permita su ejecución de sus diferentes componentes de forma separada.

Deberá ser una propuesta de construcción sobre Software Libre con una definición completa del sistema y una propuesta de implantación.

Para la definición de este sistema trataremos de responder a la siguientes cuestiones:

Qué entendemos por e-Voting.

Definiremos que tipos de consultas se consideran e-voting y localizaremos la aplicación dentro de los diferentes tipos existentes.

Qué entendemos por consulta.

Es necesario especificar las diferentes fases que componen una consulta y qué entidades están implicadas en su realización para poder caracterizar las funciones necesarias.

Qué estándares son aplicables.

A partir casos reales tratar de establecer que estándares son aplicables y deben tenerse en cuenta para garantizar la validez del proceso.

Cómo funcionan las aplicaciones para e-Voting.

Definiremos cual es el modelo a utilizar a partir de los existes para las aplicaciones de e-Voting y qué características deben cumplir este tipo de aplicaciones. Descripción de su funcionamiento para cada una de las fases de una consulta así como de los conceptos teóricos implicados.

Aplicaciones existentes.

Realizaremos un búsqueda de las aplicaciones de código abierto existentes que se ajusten al modelo establecido a fin de analizar su posible utilización como punto de partida en el desarrollo del sistema de consultas.

Herramientas para la comparación y análisis.

Seleccionar o implementar herramientas que permitan la comparación de las diferentes aplicaciones. Debe poderse analizar tanto las características correspondientes al modelo de desarrollo de código abierto como el cumplimiento de las especificaciones y estándares definidos para un sistema de consultas.

1.2 e-Voting

Es un hecho generalizado utilizar todo tipo de medios informáticos y de telecomunicaciones en la realización de elecciones o consultas. Se considera que el concepto e-Voting(1) se refiere a aquellos procesos electorales o de consulta en los que se utilizan medios electrónicos, como mínimo, en la captación del voto. En el caso en que el voto se realice desde dispositivos que no se encuentren situados en un colegio electoral y por tanto no controlados se define como e-Voting remoto(1)

Aunque en una primera aproximación pueda parecer una cuestión con un grado de complejidad asequible, en la práctica presenta algunos aspectos de gran dificultad, ya que implica la creación e implantación de sistemas seguros y, a la vez que permitan su despliegue de una manera sencilla y temporal. Resulta igualmente un problema difícil de tratar el hecho de identificar al votante y, al mismo tiempo, garantizar el secreto del sentido de su voto.

Más allá de cuestiones meramente técnicas, hay quien pone en duda(2) la necesidad de variar un sistema, urna físicas y papeletas,(3) que ha demostrado su eficacia durante mucho tiempo además de una simplicidad que asegura la comprensión y la comprobación de todo el proceso.

Con todo existen también argumentos para defender los sistemas de voto electrónico, como el hecho de facilitar la participación de personas desplazadas, con problemas de movilidad o el hecho de permitir realizar consultas directa sobre una gran cantidad de temas de una manera rápida y asequible.

1.3 Tipos de e-Voting

Se considera e-Voting aquel proceso electoral en el qué, como mínimo, el voto se recoge de forma electrónica(4); esto puede hacerse mediante dispositivos ópticos como escaners, mediante máquinas en las que se registra el voto directamente o DREs (Direct Recording Electronic computers) o bien mediante una aplicación accesible vía internet. En este último caso el voto se puede emitir en ambientes controlados o bien se realiza desde cualquier dispositivo (ordenador, Smartphone...) que pueda comunicarse con la aplicación de voto.

Finalmente la papeleta que se remite a la aplicación no es la captura de una papeleta física si no que se trata de datos tratados para garantizar su confidencialidad. Una representación de los diferentes tipos de voto electrónico puede verse en el siguiente gráfico.

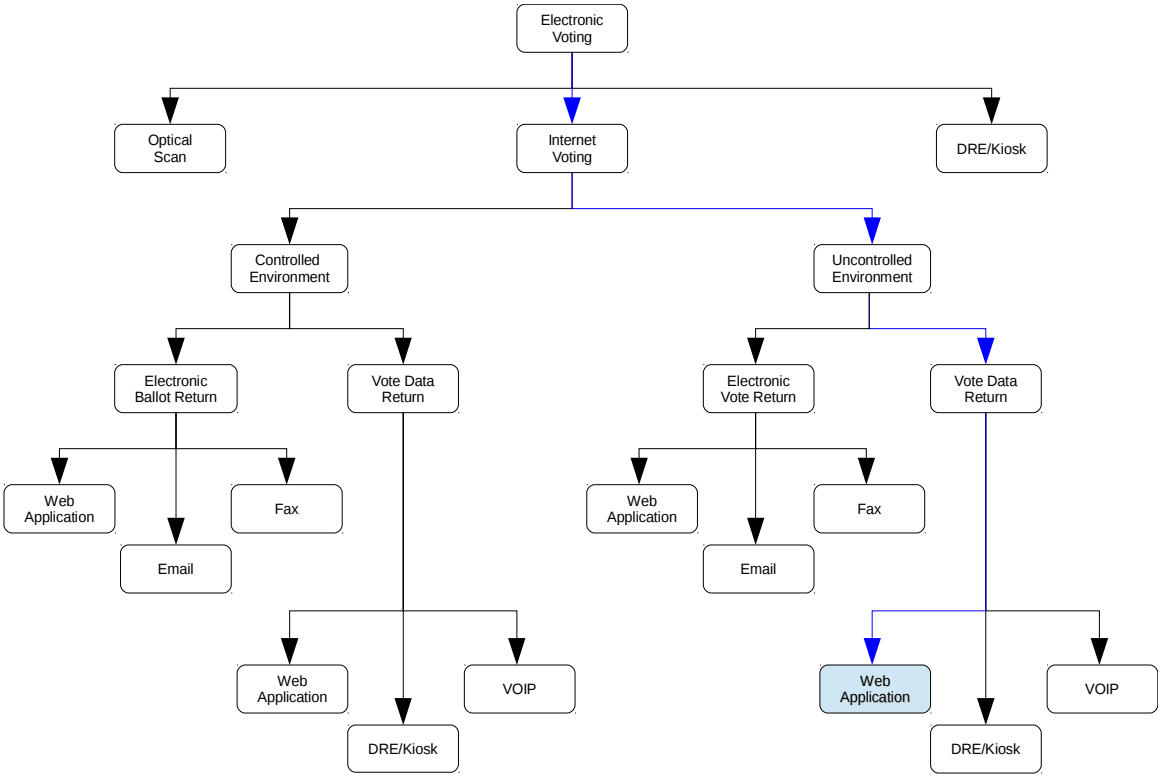


Ilustración 2: Tipos de e-Voting (5).(A Survey of Internett Voting, U.S. Election Asistance Comision)

Dada esta clasificación, el sistema planteado como objetivo deberá implementar un Aplicación web que permita consultas de tipo e-Voting remotos en entornos no controlados. El trabajo de investigación realizado se ha centrado, por tanto en la búsqueda de aplicaciones que permitan la realización de consultas mediante un interfaz web con acceso desde cualquier navegador o dispositivo independientemente de las versiones concretas de cada uno.

2 Proceso de Consulta

En un ámbito general con una “consulta” se busca decidir entre las opciones sobre una pregunta o un grupo de preguntas relacionadas o no. Si el resultado de una consulta se considera vinculante es decir, existe un acuerdo previo de aceptar el resultado por todos los participantes, se denomina Referéndum.

Una “elección” es también un caso particular de consulta en el que se escoge entre unas “opciones”; concretamente entre candidatos a ocupar una determinada posición. Dentro del alcance de este proyecto se especifica que la aplicación debe estar orientada a la realización de consultas.

Como fase previa a la definición de la aplicación es necesario detallar en que consiste un Proceso de Consulta así como sus fases y los diferentes papeles que es posible identificar como necesarios para su realización. Para ello utilizaremos los estándares creados por organismos europeos y organizaciones internacionales de estandarización.

2.1 Estándares existentes

La Comisión de Asistencia la las Elecciones (5)(La U.S. Election Assistance Commission EAC) Publicó en el año 2011 un documento titulado “A Survey of Internet Voting” con una exhaustiva recopilación de las votaciones realizadas por medios electrónicos en todo el mundo. Entre los datos recopilados se encuentra que estándares se han seguido en el diseño de los sistemas utilizado en cada uno de los casos. No existe unanimidad a la hora de aplicar unos estándares concretos y dependiendo del país se aplican diferentes normas.

En las elecciones celebradas en el entorno europeo el más utilizado, aunque no el único, es el documento publicado por el Consejo Europeo(1) en el año 2004 “Legal, operational and technical standards for e-voting: recommendation Rec(2004)11 adopted by the Committee of Ministers of the Council of Europe on 30 September 2004 and explanatory memorandum”

Este documento se ha tomado como referencia (5) en la realización de las siguientes consultas:

	Sponsor:	Election Type:	Date or Voting Period:	Target Population:	Channel:	Technology Provider:
Austria	Federation of Students	Student Union Election	May 18-22, 2009	Austrian Student Union	Uncontrolled>Vote Data Return>Web Application	Austrian Federal Computing Center, Scytl[99]
Estonia	National Election Commission	Local Election	2009	General Electorate	Uncontrolled>Vote Data Return>Web Application	Estonian Government, AS Cybernetica[115]
Finland	Ministry of Justice	Municipal Election	October 26, 2008	Registered voters of Kauniainen, Karkkila and Vihti	Controlled>Vote Data Return>DRE/Kiosk	TietoEnator, Scytl
Netherlands	Ministry of the Interior and Kingdom Relations	Federal Election, European Parliament	2004: June 1-10, 2004	Dutch voters living abroad[168]	Uncontrolled>Vote Data>Web Application	LogicaCMG and Rijnland District Water Control Board
Norway	Ministry of Local Government and Regional Development	Federal Election	August 10 – September 12, 2011	General Electorate	Uncontrolled>Vote Data Return>Web Application	Norwegian Government, Scytl, ErgoGroup

Tabla 1: Experiencias de Voto con Estándares CE

En este documento se detallan y describen 112 estándares recomendados y considerados necesarios para garantizar la confianza en los procesos de votación por medios electrónicos.

El documento se estructura en tres apéndices:

- Estándares legales: Estándares, relacionados con los principios básicos, a tener en cuenta en un proceso de elección.
- Estándares operacionales: Los relacionados con la preparación, realización y seguimiento del procedimiento.
- Estándares técnicos: Los relacionados con los medios utilizados así como la seguridad necesaria y la auditoría del proceso.

Finalmente destacaremos dos recomendaciones sobre la utilización de las normas que aseguren la Accesibilidad y Usabilidad de la aplicación y la recomendación de utilizar EML, un lenguaje desarrollado específicamente para implementar aplicaciones y dispositivos orientados a la realización de elecciones.

2.1.1 Accesibilidad y Usabilidad

Dentro de los estándares del Consejo Europeo se recomienda tener en cuenta las normas publicadas por la WAI (6)(Iniciativa de accesibilidad web) referentes a la creación de paginas web. Estas recomendaciones permiten la creación de contenidos Web de manera que puedan ser accesibles a personas con discapacidades.

2.1.2 EML(Election Markup Languaje)

Otra de las recomendaciones efectuadas(1) por el Consejo Europeo en su definición de estándares, es la conveniencia de utilizar Election Markup Languaje (7)(en adelante EML) desarrollado por OASIS (**Organization for the Advancement of Structured Information Standards)**(8).

Este estándar se ha desarrollado para permitir el intercambio de información entre el hardware, software y proveedores de servicio implicados en cualquier aspecto relacionado con el desarrollo de elecciones o consultas.

Para ello se definen los procesos y estructuras de datos implicadas así como los formatos para el intercambio de información creando un modelo genérico que abarca todo el proceso electoral. Este modelo identifica tres fases. Una inicial de preparación (pre-voting), la correspondiente a la votación propiamente dicha (voting), y finalmente la de recuento y publicación de resultados (post-voting).

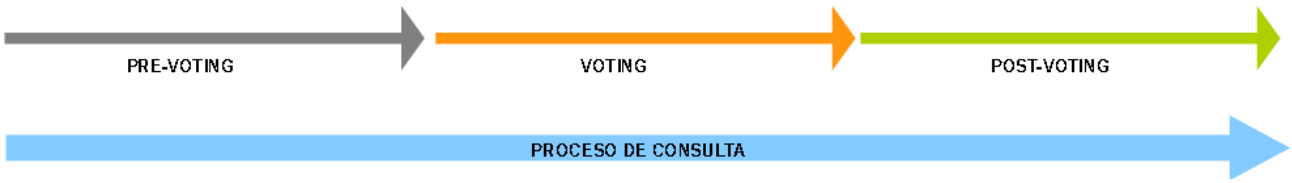


Ilustración 3: Fases de la Consulta.

2.2 Actores (roles) del Proceso.

A partir del modelo definido dentro del estándar EML es posible concretar unos actores diferenciados con tareas asignadas en cada una de las fases.

- Autoridad: Es la responsable de iniciar y definir la Consulta.
- Administrador: Debe configurar y ejecutar la consulta en todas sus fases.
- Auditor/observador: Debe garantizar el proceso y la corrección de los resultados en todas sus fases.
- Votante: Como participación activa en el proceso escoge entre las opciones y deposita su papeleta. Aunque puede también entenderse como participación la inclusión en el censo o la recepción de los resultados.

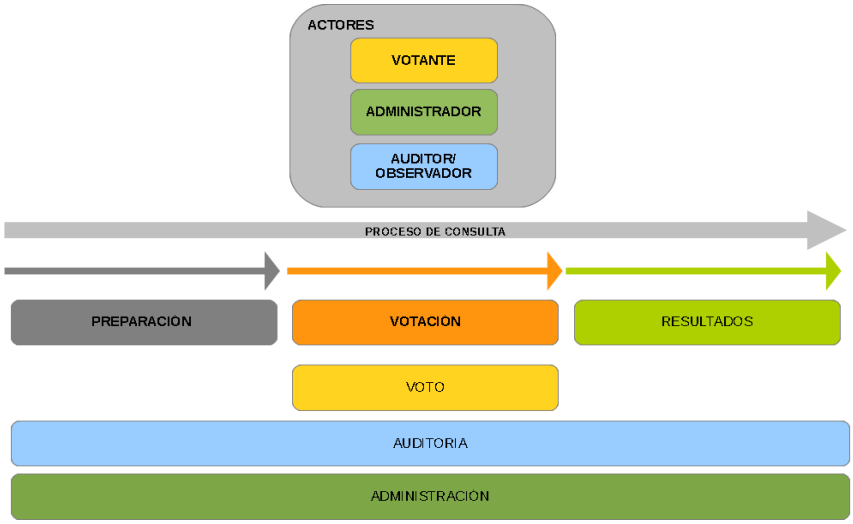


Ilustración 4: Actores de una Consulta.

2.3 Fases de la Consulta

El modelo detallado en la Versión 05 del estándar EML abarca tanto la realización de Consultas como la de Elecciones; para describir las diferentes fases utilizaremos un modelo simplificado orientado únicamente a la realización de una Consulta.

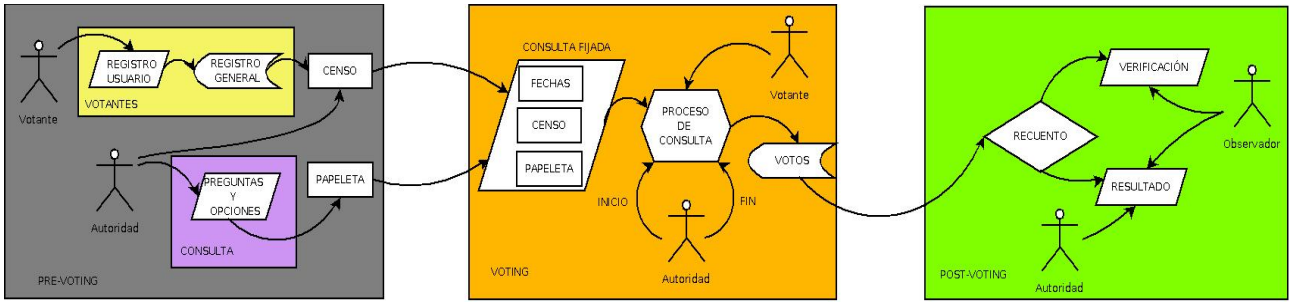


Ilustración 5: Esquema de Fases de una Consulta en el modelo EML.

A partir de este modelo simplificado describiremos esquemáticamente cada una de las fases y las tareas más significativas que corresponden a cada una de los actores.

2.3.1 Fase Pre-Voting

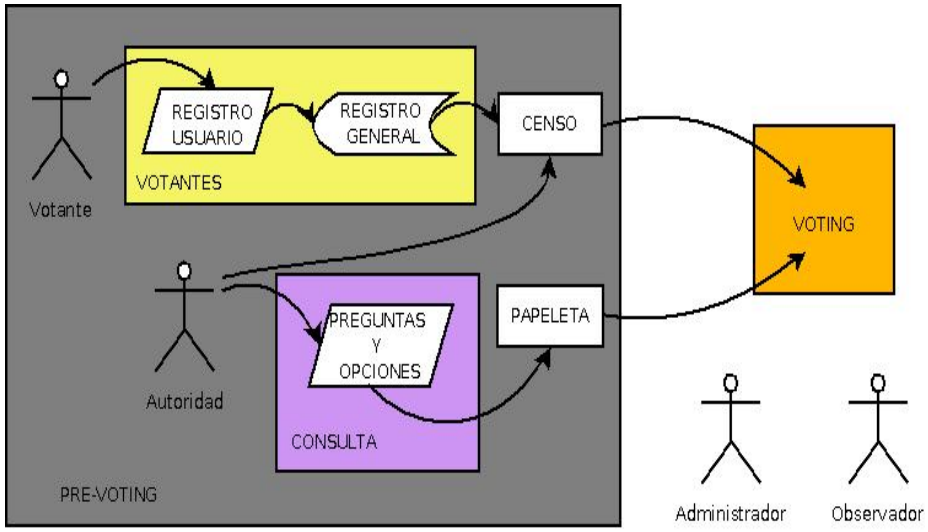


Ilustración 6: Esquema Fase pre-voting EML.

Esta primera fase comprende las tareas previas de preparación de la consulta; así los votantes deben ser registrados dentro del censo de la votación; la autoridad debe crear la consulta con los datos concretos como el nombre o la duración y registrar la o las preguntas con las opciones de respuesta disponibles.
El administrador debe crear o administrar las estructuras de datos necesarias y el observador verificar que la papeleta se corresponde con la consulta así como la corrección del censo.

2.3.2 Fase Voting

Una vez se ha definido la consulta es necesario fijarla e iniciar el periodo de votación. En esta fase se reciben y almacenan los votos emitidos. La Autoridad es la encargada de iniciar y finalizar el proceso:

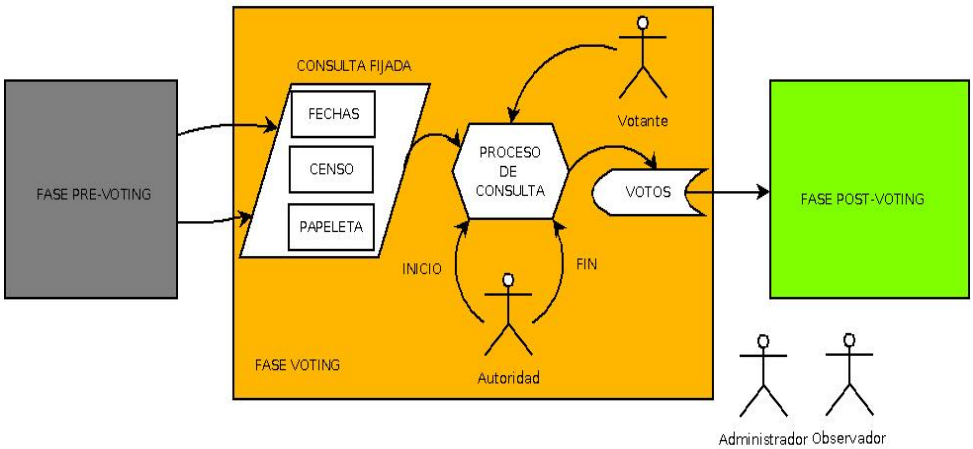


Ilustración 7: Esquema Voting EML.

El administrador es responsable de del funcionamiento del sistema y de asegurar los votos recibidos. Por su parte el Observador debe comprobar que es posible acceder al sistema y que la

integridad de los votos está garantizada.

2.3.3 Fase Post-Voting

En esta fase la Autoridad publica el resultado en función del recuento de los votos y el Observador garantiza por un lado la corrección del recuento así como del resultado derivado de éste.

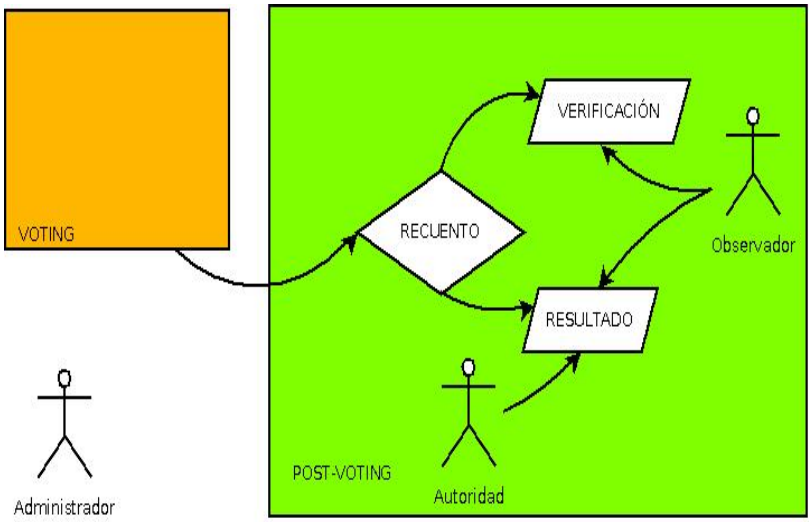


Ilustración 8: Esquema fase Post-Voting EML.

Por su parte el administrador debe garantizar el procesamiento del recuento y el cálculo del resultado.

3 Definición Teórica de la Aplicación

A la hora de hablar de aplicaciones dedicadas a la votación electrónica debe distinguirse claramente entre los siguientes (9)conceptos (10):

Esquema de votación (Voting Scheme): Sistemas y protocolos de encriptación de información empleados para garantizar la elegibilidad y confidencialidad del voto.

Métodos de votación (Votings Methods): Algoritmo utilizado para escoger entre los diferentes candidatos en función de los votos emitidos.

Sistemas de votación (Voting System): Se refiere a la implementación de un conjunto de elementos que permiten realizar una consulta, es decir el sistema completo.

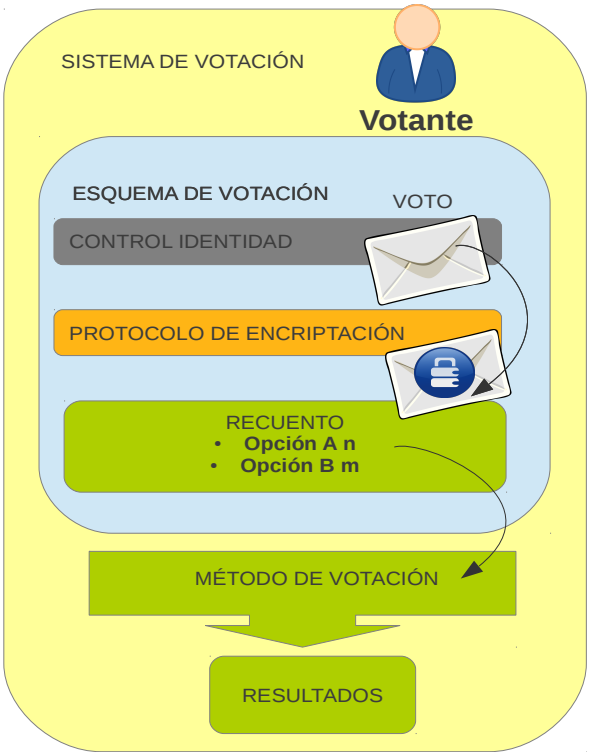


Ilustración 9: Esquema genérico sistema de votación.

La aplicación a definir es por tanto un Sistema de votación ya que se pretende abarcar todo el proceso.

3.1 Esquemas de Votación

Los esquemas de votación son el núcleo de cualquier sistema de votación, son también la parte más compleja. Estos esquemas intentan resolver la cuestión básica de cómo garantizar que el votante está autorizado a votar pero a la vez garantizar que el voto es anónimo, es decir una vez depositada la papeleta no puede relacionarse con un votante en concreto. Además en los tipos de e-Voting remoto es necesaria la utilización de canales de comunicación para el intercambio de información entre los diferentes participantes (votantes, autoridad, observadores) esto implica la necesidad de

proteger esta información de alguna manera para que no pueda ser utilizada o modificada. Para todo esto se utilizan Técnicas Criptográficas[3].

Resulta por tanto imprescindible familiarizarse con algunos conceptos básicos de las técnicas criptográficas utilizadas en este tipo de aplicaciones, para ello no entraremos tanto en la formalización matemática de estos conceptos, disponible sin mayores dificultades, ya que todos los elementos utilizados son de conocimiento publico y de uso libre, si no que intentaremos una aproximación a cada uno de ellos que permita comprender su funcionamiento básico de forma intuitiva.

3.1.1 Diferencia entre Codificación y Encriptación

Un código es un modo formalizado de expresar una información. Si el emisor (quién envía la información) y el receptor (quién recibe la información) conocen la forma en que se han codificado los datos no habrá problema en el proceso de comunicación; estos datos estarán también disponibles para cualquiera capaz de observar el intercambio y que, también, conozca el código.

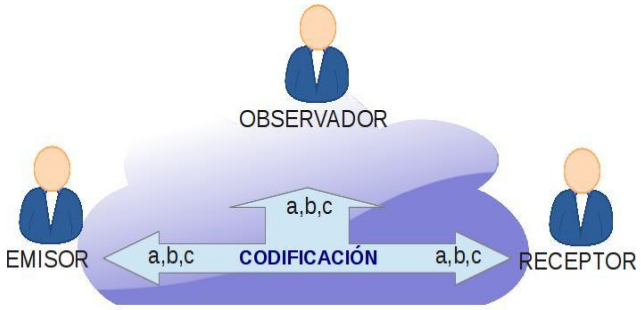


Ilustración 10: Esquema Codificación.

Si es necesario preservar la información secreta para cualquiera que no sean el emisor y el receptor habrá que "encriptar" los datos, es decir "ocultarlos". De esta manera la información encriptada se intercambia codificada de manera que, aunque sea posible para un tercero observar el intercambio, no es posible entender la información.

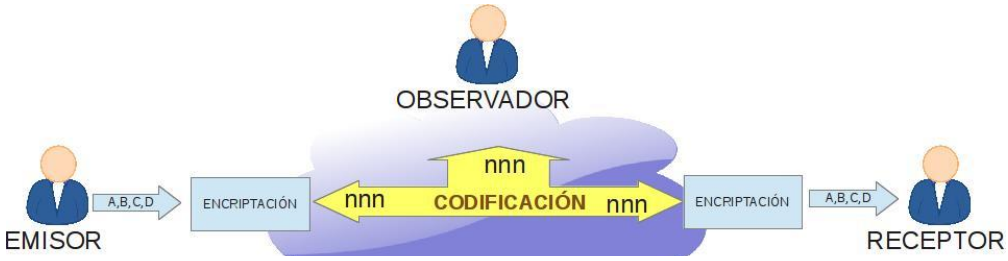


Ilustración 11: Esquema encriptación.

Son ejemplos de código el lenguaje Morse desarrollado para su utilización en el telégrafo o el lenguaje HTML desarrollado para la creación de páginas Web.

3 Criptografía provine de las palabras griegas *Krypto*("esconder") y *Grapho*("escribir") se trata por tanto de la ciencia o el estudio de las escritura secreta.

Canales de Comunicación

Durante el proceso de votación es necesario establecer comunicaciones entre los diferentes implicados; los canales de comunicación empleados deben proporcionar seguridad a estos intercambios y pueden ser(11):

Canales Anónimos

Son los canales de comunicaciones que cancelan o eliminan la identidad del Votante tanto para la Autoridad como para cualquier posible observador. Esto se consigue mediante encriptación y permutación de la información realizadas a través de unos dispositivos físicos llamados Mixnets. La implementación de este tipo de canales es compleja y complicada sobre todo en sistemas de gran envergadura.

Canales Seguros

- Canal No Interceptable (Untapable Channel): Es un canal de comunicaciones no interceptable y por lo tanto seguro; la implementación de este requiere la instalación de lineas punto a punto[4] al margen o segregadas de la red pública de comunicaciones lo que resulta muy complicado e inviable en sistemas de gran envergadura.
- Canal privado (Private Channel): Utilizan la red de pública comunicaciones y por lo tanto es un canal observable, la seguridad se implementa a partir de sistemas de encriptación de clave privada o pública.

3.2 Tipos de Esquemas de Votación

Dependiendo del tipo de privacidad empleada en la comunicación del voto entre el votante y el sistema de votación los esquemas pueden clasificarse en tres tipos(11):

- Votante Oculto: Los votantes depositan el voto de forma anónima. Para la identificación cada voto se adjunta con un testigo(prueba) de la autenticidad del votante (token o bulletin board). Aunque el proceso de recuento es el más simple es preciso el empleo de canales anónimos por lo que este tipo de voto resulta complicado de implementar en sistemas de un cierto tamaño.
- Voto Oculto: El votante depositan un voto secreto (encriptado), junto con un prueba para certificar que es el autor del voto, a través de un Canal Privado utilizando la red de comunicaciones pública convencional. El cálculo del escrutinio puede precisar una cierta capacidad de proceso, ya que es necesario comprobar los votos y desencriptarlos; en votaciones de tipo medio esto no representa un gran obstáculo.
- Votante Oculto y Voto Oculto: Los votantes depositan el voto de forma anónima y encriptada y por lo tanto presenta las dificultades de los dos sistemas, son necesarios canales anónimos y puede precisar gran cantidad de proceso de cálculo.

Dadas las dificultades de implementación que presentan los otros dos tipos de comunicaciones el sistema empleado en el diseño de la aplicación sería de tipo Voto Oculto y por tanto se emplearían Canales Privados es decir canales que utilizan la red pública de comunicaciones y protegidos por medios criptográficos.

4 Puede considerarse que una línea punto a punto es como un cable que une directamente dos teléfonos sin ninguna derivación lo cual impide que nadie capte la conversación.

3.3 Esquema de Votación de Voto Oculto

En este tipo de esquemas, y de forma resumida, el Votante envía una papeleta con el contenido secreto (encriptado) junto con una prueba de que es el autor, para ello se utiliza una clave que previamente le ha sido facilitada por la Autoridad. Este voto junto con la prueba se guardan hasta el momento del recuento(12).

Para obtener el resultado final la Autoridad comprueba la autenticidad de cada uno de los votos utilizando la prueba adjunta; dado que conoce la clave puede ver el contenido del voto y por tanto puede efectuar el cálculo del escrutinio. Debido al tipo de claves utilizadas cualquier observador puede verificar la corrección de este escrutinio a partir de los votos secretos sin necesidad de conocer su contenido.

A continuación describiremos cada una de las fases de un proceso de votación utilizando un esquema de Votación de Voto Oculto.

Para cada una de las fases se identifican los conceptos criptográficos implicados y se resaltan, enmarcados en azul en los gráficos descriptivos a fin de poder ubicar cada uno de ellos en el desarrollo del proceso de votación. Nos aproximaremos a cada uno de estos conceptos de forma que sea posible comprender su funcionamiento sin necesidad de realizar una descripción formal. Todos son de uso público por lo que resulta posible encontrar abundante y detallada información sobre ellos.

3.3.1 Fase Pre-Voting

En esta primera fase se generan las claves necesarias que permitan el intercambio de la información de forma segura; para ello se utiliza **La Criptografía de Clave Asímetrica(13)**. En el esquema propuesto es la Autoridad la que crea las claves pero es posible encontrar esquemas alternativos en los que son generadas por el votante o están fraccionadas entre diversas autoridades. Una vez creadas las claves se remite a cada uno de los votantes que utilizará para encriptar su voto.

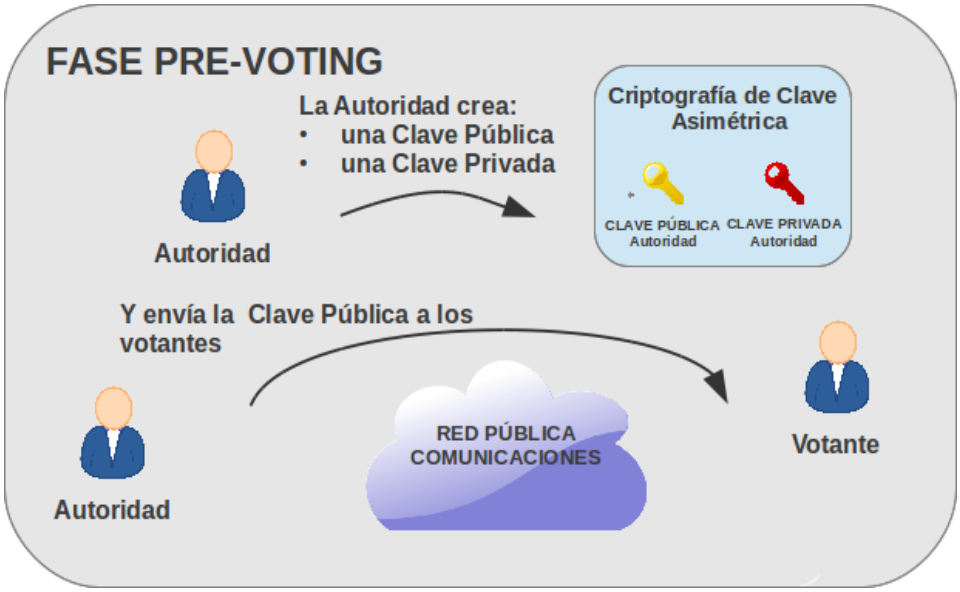


Ilustración 12: Esquema Sistema de Votación Fase Pre-Voting.

Criptografía de Clave Asimétrica

Los sistemas de clave asimétrica permiten el intercambio de información de forma segura en una comunicación.

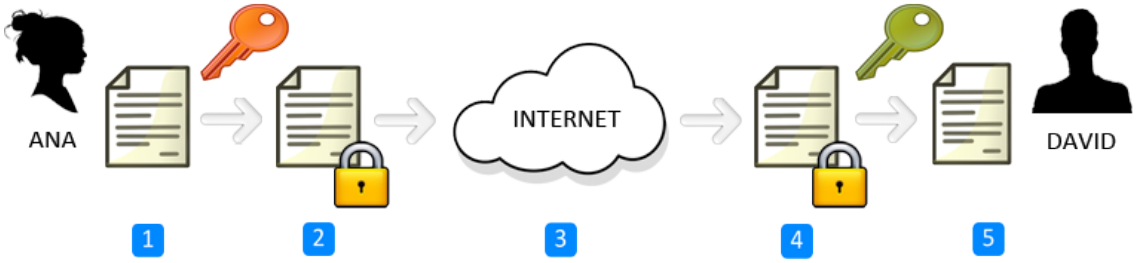


Ilustración 13: Esquema Criptografía de Clave Asimétrica(37)

Ana redacta un mensaje

1. Ana cifra el mensaje con la **clave pública** de David (que este previamente ha enviado o publicado de algún modo)
2. Ana envía el mensaje cifrado a David a través de internet, ya sea por correo electrónico, mensajería instantánea o cualquier otro medio
3. David recibe el mensaje cifrado y lo descifra con su **clave privada**
4. David ya puede leer el mensaje original que le mandó Ana

Para poder conocer el contenido del mensaje es imprescindible disponer de la clave privada. Por lo tanto es posible enviarlo por canales no seguros como el correo electrónico ya que para cualquier observador el contenido el mensaje carecerá de sentido.

Diffie-Hellman, ElGamal (14)o Pallier son algoritmos que permiten implementar este tipo de encriptación.

3.3.2 Fase Voting

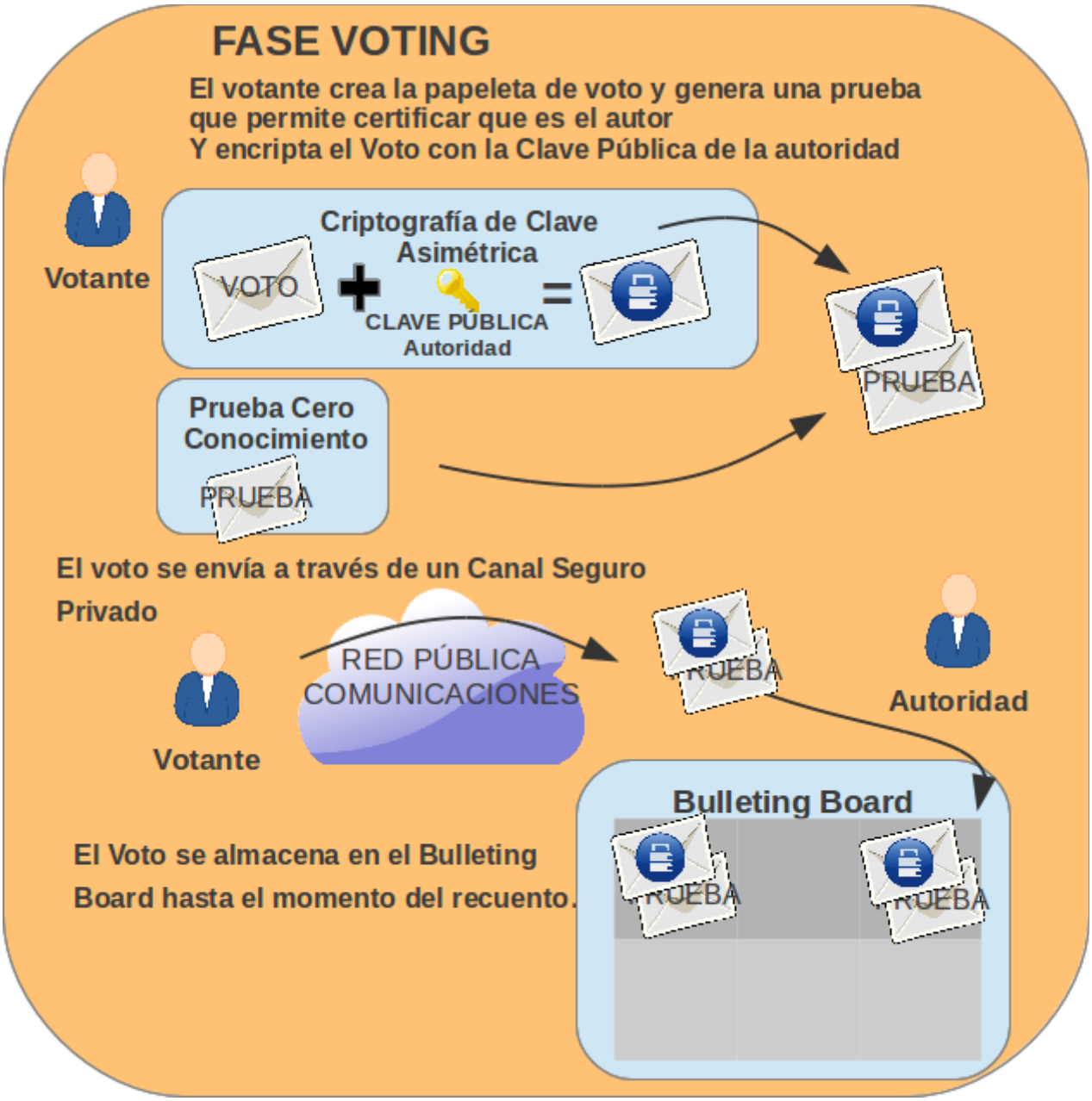


Ilustración 14: Esquema Sistema de Votación Fase Voting.

El votante utilizando la clave pública de la Autoridad crea un voto encriptado y un **Archivo de Prueba** que demuestra que es el autor del voto y envía, los dos, a la Autoridad. Ambos archivos son almacenados de forma segura en una estructura de datos llamada **Tabla de anuncios** (Bulleting Board) hasta el momento del recuento.

Métodos de Prueba(15) (Interactive and non interactive proofs).

En determinadas circunstancias es posible que a la Autoridad de una votación se le solicite que demuestre la validez de una determinada operación criptográfica o el votante deba demostrar que su

voto es válido. Pero, en ninguno de los dos casos, no debe revelarse ningún dato más allá de la corrección de la operación; para esto es utilizan los Métodos de Prueba o Zero Knowledge Proofs (Pruebas de Cero Conocimiento) es decir métodos en los que se comprueba que se conoce un secreto pero no se revela.

En los métodos interactivos una entidad P (Prover) debe demostrar que conoce un determinado *"secreto"* a una entidad V (Verifier). Normalmente se requieren tres interacciones.

- P envía un mensaje con el compromiso (Commitment) de que conoce el secreto
- V envía un mensaje con un desafío (Challenge) para que P demuestre su compromiso
- P envía un mensaje con la prueba (Proof) que V aceptará si es correcta. Este proceso puede repetirse varias veces para asegurar la verificación.

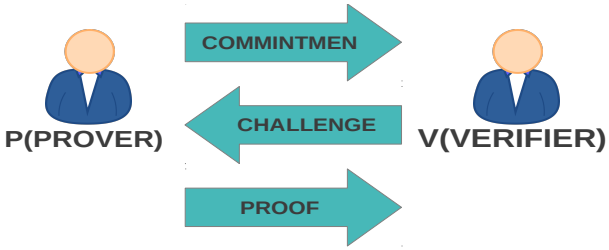


Ilustración 15: Esquema Métodos de Prueba.

En las pruebas no interactivas el proceso se realiza en una sola interacción. P envía el compromiso, el desafío y la prueba a V, que únicamente debe decidir si acepta la transacción.

Este último tipo de pruebas no interactivas permiten, dentro de un sistema de voto, crear una papeleta y adjuntar una prueba de que está correctamente encriptada sin revelar el contenido, es decir el sentido del voto.

Tabla de Anuncios (Bulleting Board)

Puede entenderse como un "Casillero" en el que cada usuario autorizado tiene un espacio reservado en el que únicamente su propietario puede depositar notas pero cualquiera puede leerlas. La tabla de anuncios o Bulleting Board (11)se describe como un canal de comunicaciones público con memoria, toda la información que se crea y se transmite queda registrada de forma accesible.

Existen zonas reservadas en las que se precisa el uso de contraseñas y en las que únicamente el Votante autorizado puede escribir y guardar sus votos, pero es accesible como lectura de manera que pueda comprobarse la corrección del voto. Igualmente, la Autoridad puede publicar información, como el resultado y los datos necesarios para verificarlo.

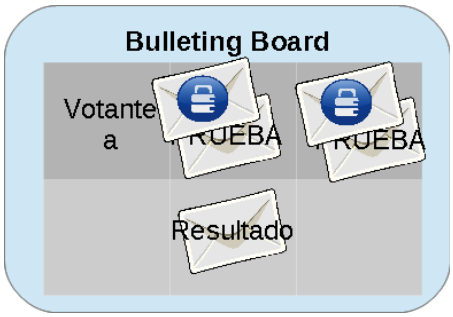


Ilustración 16: Esquema Tablón de anuncios.

3.3.3 Fase Post-Voting

Una vez finalizada la votación en esta fase la Autoridad comprueba, utilizando la Prueba de Cero Conocimiento, que los votos corresponden a un votante autorizado; utilizando su clave privada desencripta cada uno de los votos válidos[5] y se obtiene el resultado final de la votación. Gracias a las propiedades homomórficas(16) (17) de algunos algoritmos de Clave Asimétrica respecto a operaciones como la suma y la multiplicación, es posible comprobar que toda la operación es correcta sin necesidad de conocer el contenido de cada voto. Es este último punto, al que en principio puede resultar más complicado enfrentarse pero es, también, el concepto en el que se basa, la capacidad de verificar el resultado sin revelar el sentido de voto.

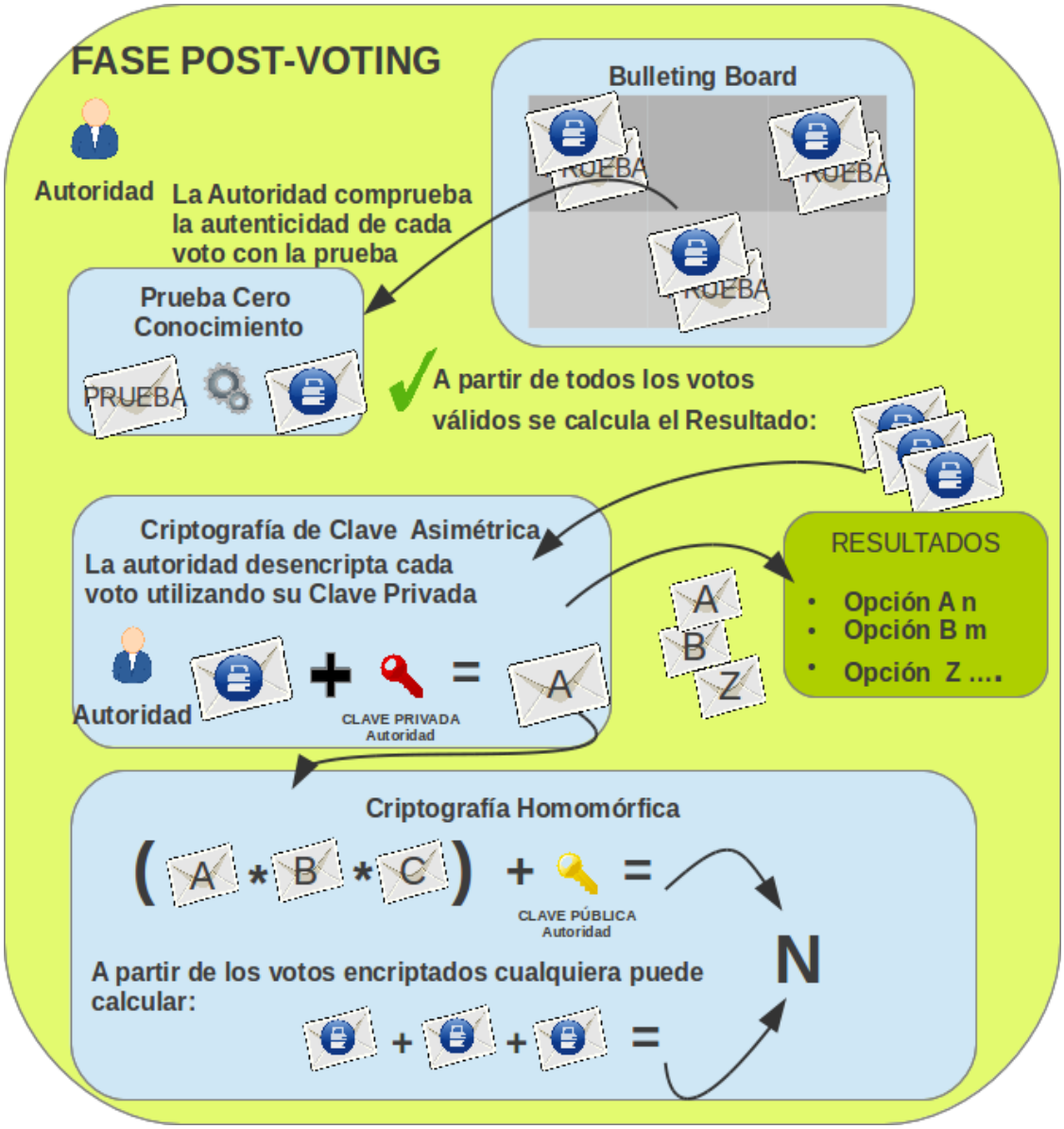


Ilustración 17: Esquema Sistema de Votación Fase Post-Voting.

5 Una Autoridad desleal podría relacionar los votos con los votantes; es por ello que en los esquemas de este tipo existe más de una autoridad y cada voto se divide entre ellas es necesaria la participación de todas para realizar el escrutinio.

Criptografía Homomórfica

Como paso previo es importante tener en cuenta que cualquier archivo informático es en definitiva un número, de manera que una papeleta con un **SI** (Codificado en Codigo ASCII) se podría representar con el número binario 11100111101001 formado por los dos caracteres **S** (1110011) e **I** (1101001) y se corresponde con el número 14825 en decimal.

Carácter	Valor Binario	Decimal
s	1110011	115
i	1101001	105
si	11100111101001	14825

Tabla 2: Valor Binario SI.

En el caso de un archivo con un **NO**:

Carácter	Valor Binario	Decimal
n	1101110	110
o	1101111	111
no	11011101101111	14191

Tabla 3: Valor Binario NO.

Como ejemplo de homomorfismo podemos utilizar la función Logaritmo respecto de las operaciones Suma (R,+) y la Multiplicación (R+,*) lo que significa que el logaritmo resultante de la multiplicación de dos números es igual a la suma de los logaritmos de cada uno de los números.

$$\ln(a*b)=\ln(a)+\ln(b)$$

Lo podemos comprobar con los valores correspondientes a los archivos SI y NO:

Si = 14825
No = 14191

Logaritmos de ambos números

Ln 14825 = 9,6
Ln 14191 = 9,56

Efectivamente se cumple que:

Ln (ab) = 19,16
Ln(a)+Ln(b) = 19,16

Esto es igualmente cierto para algunos algoritmos de Clave Asimétrica como ElGamal o Paillier así, si multiplicamos el valor que representa cada voto sin encriptar, el resultado es igual a la suma de los votos encriptada.

Más formalmente expresado diremos que para tres mensajes **(a) ,(b) y (c)**[6] Encriptados **(E)** con una clave **(k): E_k(a), E_k(b), E_k(c)** se cumple :

$$E_k(a*b*c) = E_k(a) + E_k(b) + E_k(c)$$

6 En la práctica la papeleta tiene un formato (a,r) (b,r) donde r es un número aleatoriamente escogido ya que de otra forma todas las papeletas Si o No serían iguales una vez codificadas.

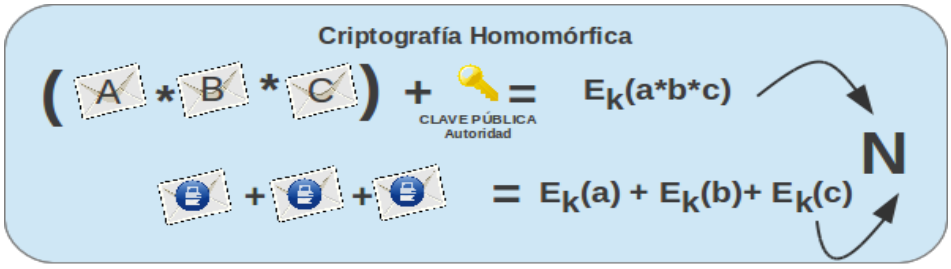


Ilustración 18: Esquema Criptografía Homomórfica.

En un Esquema de Votación y basándose en esta propiedad, la Autoridad desencripta[7] cada voto y realiza el escrutinio, además calcula la suma de todos los votos y la encripta publicando este dato junto con una prueba de que se ha codificado correctamente. Cualquiera puede calcular la multiplicación de los votos encriptados y verificar que coincide con el resultado publicado **lo que implica que no se modificado ni eliminado ninguno de los votos** emitidos(11).

7 Existen métodos en los que no es necesario desencriptar cada voto individualmente y todo el proceso se realiza sin que sea necesario este paso.

3.4 Características de Seguridad

Se entiende como características de seguridad las que garantizan los principios básicos que aseguran la validez del procedimiento de votación.

Existe una abundante literatura académica sobre las características de seguridad que debe cumplir un esquema de votación. Parece lógico qué, como mínimo, deban garantizarse las características esperadas de un método de votación tradicional. No hay un acuerdo total y, en algún caso, existen definiciones o denominaciones redundantes.

Ilustración 19: Esquema Características de Seguridad.

Pero se deberían implementar de la forma más completa posible las siguientes características(18) (19) (20):

Registro y verificación (Eligibility): Se debe garantizar que la identidad de la persona que emite el voto pertenece al censo de la votación; también puede referirse al hecho de emitir un único voto. En el caso de los sistemas electrónicos esto puede no ser así, es posible emitir más de un voto y únicamente ser valido el depositado en último lugar. Algunos autores consideran esto último como una característica independiente denominada **Democracy.**

Privacidad (Privacy): No debe revelarse información sobre un votante o sobre el contenido de su voto.

Verificabilidad (Verifiabiliy): Debe ser posible verificar todo el procedimiento de votación tanto individualmente, verificar cada uno de los votos, como de forma global, verificar o auditar el recuento y los resultados.

Imparcialidad (Fairness): El procedimiento de voto no debe inclinar la decisión de los votantes a favor o en contra de una determinada opción, como por ejemplo mediante la redacción o presentación de las opciones o candidatos en la papeleta. Además debe garantizar el acceso al procedimiento de votaciones a todos los participantes por igual como ejemplo la traducción a todos los idiomas necesarios o la implementación de medidas para personas con problemas visuales.

Solidez (Robustness): El sistema debe ser capaz de resistir ataques activos o pasivos (combinación de administradores y/o votantes malintencionados) o posibles errores que puedan entorpecer el procedimiento.

Las dos características siguientes son similares ya que están relacionadas con la necesidad de mantener el secreto sobre el sentido del voto así como impedir que sea impuesto por coacción o por algún tipo de compensación:

Resistencia a la Coacción (Coercion-Resistance): En caso de existir coacción sobre un votante, no puede probarse en qué el sentido ha emitido el voto, impidiéndose de esta forma comprobar si la coacción ha tenido resultados.

Uno de los principales problemas, en los procedimientos de voto remotos, es garantizar que no se producirá ninguna coacción **(Coercion)** sobre los votantes, es decir se influirá de una forma u otra en el sentido del voto emitido. En este sistema el voto no se ejerce en un ambiente controlado como un colegio electoral por lo que el votante es fácilmente vulnerable a esta amenaza.

Es por esto que en muchos casos se recomienda este tipo de Sistemas de Votación para lo que se denomina consultas "Low- coercion"(4) es decir elecciones donde el secreto del voto es importante, pero el hecho de votar remotamente es garantía suficiente de privacidad aunque el voto no se realicen en entornos controlados, como por ejemplo un colegio electoral.

Un ejemplo de ese tipo de consultas sería la elección del Presidente de la Universidad de Lovaina (2009)(21) o las Elecciones para la Renovación del Claustro de la Universidad Politécnica de Catalunya en el año 2012. En general se trata de procesos que implican un numero limitado de votantes. [8]

Los procedimientos de voto que se ejecutarían con el sistema a diseñar entraría dentro de esta categoría.

Así mismo dentro de las recomendaciones efectuadas por el Consejo Europeo (4)se mencionan estas consultas como las más indicadas para realizar pruebas piloto o test como paso previo a la introducción del voto electrónico en procesos electorales de mayor envergadura.

Resistencia a la Venta del Voto (Receipt-Freeness): No es posible obtener ninguna prueba escrita o impresa sobre el sentido del voto emitido, siendo imposible certificar que se ha cumplido con lo pactado en caso de acuerdo.

Dependiendo del autor se describen otras características como:

Corrección (Correctness): Se asegura que los resultados reflejan exactamente el recuento de los votos emitidos. Esta propiedad se denomina también como **Completeness.**

Escalabilidad (Scalability): Dada la complejidad de los protocolos empleados el sistema debe ser capaz de hacer frente a los requerimientos de capacidad de proceso, almacenamiento y comunicaciones.

Seguridad (Security): Con unas necesidades razonables de seguridad se garantiza que no será posible influir sobre el resultado de las votaciones interfiriendo el sistema (por ejemplo, impidiendo el registro de votos mediante interferencias en las comunicaciones).

8 Alrededor de 4000 en la elección celebrada en la Universidad de Lovaina

3.5 Métodos de Votación

Una vez realizada la votación y con el recuento de votos efectuado se obtiene el resultado de la votación, el método escogido para "interpretar" los resultados puede ser determinante (22) (23).

En el caso de que la pregunta planteada tenga un esquema de respuesta de tipo Si/No resulta relativamente fácil ya que la opción **mayoritaria** es la ganadora aunque pueden plantearse cuestiones como el grado de participación considerado válido o la distribución del voto en caso de existir diferentes zonas de votación.

Pero este enfoque puede no resultar válido en el caso de plantear cuestiones con varias respuestas. En este caso el problema de como escoger de forma "justa" o "correcta" a partir de los resultados obtenidos puede resultar algo más complicado.

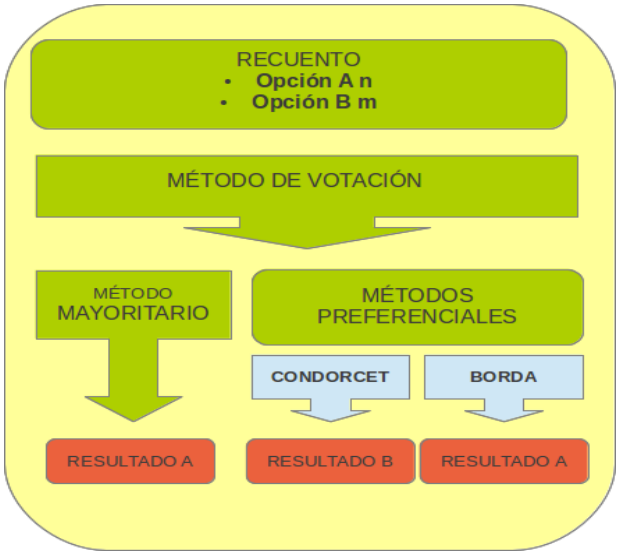


Ilustración 20: Esquema métodos de Votación.

A diferencia del método mayoritario dónde únicamente se tiene en cuenta el número final de votos; los métodos llamados **Preferenciales** permiten escoger entre diferentes opciones teniendo en cuenta el orden (preferencia) en que se han escogido en cada uno de los votos emitidos. Entre este tipo de métodos se encuentran el método Condorcet o el método Borda. Como puede verse, en el ejemplo desarrollado en el anexo 9.1 Ejemplo de Métodos de Votación(22) a partir del mismo escrutinio el resultado final puede ser diferente en función del método escogido.

Por tanto la elección del método de votación puede influir claramente el resultado final; no existe una solución "correcta" y todos los métodos presentan ventajas e inconvenientes.

4 Sistemas de Votación Free/Open Software

En momento de desarrollar un programa informático este se escribe en un lenguaje próximo al lenguaje humano sin embargo es necesario “traducirlo” a lenguaje “máquina” (compilado) para que sea posible ejecutarlo. El código original en lenguaje humano se conoce como Código Fuente. Y es imprescindible disponer de él para conocer el comportamiento o modificar un programa.

En lo que se refiere a los sistemas de votación esta es una característica muy importante, ya que, la posibilidad de inspeccionar el código y ejecutarlo independientemente, incide de forma directa en la capacidad de supervisión y auditora. Permite además tener independencia del proveedor, facilitando la interoperatividad entre dispositivos.

El Consejo Europeo recomienda el uso de este tipo de programas, tanto en el Manual de Voto Electrónico como en sus Estándares(1).

Un proceso de consulta ya sea para escoger entre unos candidatos o entre las opciones de una determinada cuestión, implican una cierta organización y la implementación de una infraestructura por mínima que sea; en la mayoría de los casos esta organización es delegada en empresas especializadas que se encargan de todo el proceso.

Las aplicaciones empleadas son suministradas como una parte más del servicio y, aunque se puedan implementar procesos de verificación y observación del código, no son de libre distribución ni pueden considerarse como de código abierto. Este es el caso de las aplicaciones utilizadas en los países en que el empleo de sistemas de votación electrónica es habitual, como Estonia(24) , Australia, o algunas experiencias como la realizada en Noruega (25). En este último caso el código es accesible con derecho a modificarlo pero con limitaciones en el uso.

En general para garantizar la seguridad de los sistemas de votación empleados en procesos electorales se mantiene en secreto gran parte del código, aplicando una política de Seguridad a través de la Oscuridad (Security trought the Obscurity[9]) y por lo tanto existen partes del sistema sobre los que no se conoce el funcionamiento, esto crea una cierta paradoja al utilizar unas urnas no del todo transparentes.

Otra aproximación a la necesidad de garantizar la seguridad es a través de la transparencia (Security trought the Transparency)[10]. El principal ejemplo de este tipo de aproximación es la criptografía moderna e implica el conocimiento de toda la información.

Este tipo de seguridad se basa en el principio de Kerckhoffs qué, de forma resumida, únicamente mantiene en secreto la clave empleada en una encriptación concreta, ya que los sistemas y algoritmos utilizados son públicos. Por lo tanto, el secreto de los datos no depende del secreto de los procedimientos seguidos o infraestructuras empleadas para encriptarlos.

De igual manera es posible aplicar este principio a los sistemas de votación ya que para poder garantizar las libertades contempladas en una licencia libre el código fuente debe ser estar disponible y ser accesible garantizando una serie de derechos a los usuarios:

9 http://es.wikipedia.org/wiki/Seguridad_por_oscuridad
10 http://en.wikipedia.org/wiki/Security_through_transparency

- Libertad 0 de usar el programa con cualquier propósito.
- Libertad 1 de estudiar cómo funciona el programa y de modificarlo adaptándolo a nuestras necesidades.
- Libertad 2 de distribuir copias del programa sin ninguna limitación.
- Libertad 3 de mejorar el programa y hacer públicas esas mejoras.

A fin de garantizar estas libertades este tipo de programas se encuentra protegido por unas licencias especificas.

La FSF (Free Sofware Fundation) es la organización que promueve este tipo software.

4.1 Licencias Free[11] Software

Este tipo de licencias se creó para poder distribuir programas informáticos garantizando las libertades ya mencionadas. Es importante saber que de ninguna manera puede considerarse que estos programas se encuentren en el "Dominio Público", es decir, se puedan distribuir sin ninguna restricción. Para ello se utiliza la legislación de protección de los derechos intelectuales o Copyright normalmente esta legislación pretende restringir el uso de los programas mediante, por ejemplo, el uso de licencias (software privado). Para describir las licencias libres se utiliza un juego de palabras entre Right (derecha) y Left(izquierda) y se consideran de tipo Copyleft.

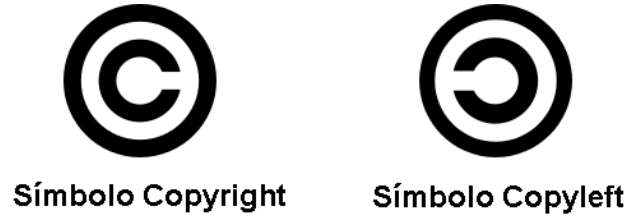


Ilustración 21: Símbolos Copyright y Copyleft.

En general las licencias copyleft se pueden dividir en dos subgrupos:

Copyleft dévil(26): De forma muy esquemática, estas licencias pueden utilizarse prácticamente sin ningún tipo de restricción e incluso es posible integrar estos programas dentro de un software privado: Un ejemplo de este tipo de licencias es la **Berkeley Software Distribution** (BSD) *(27).*

Copyleft Fuerte (24): Son licencias que obligan a mantener el programa modificado dentro de los mismos términos de la licencia original, por lo que no resulta posible integrarlos en programas privados: Un ejemplo de este tipo de licencias es la familia de licencias **GENERAL PUBLIC LICENCE** (GPL)*(28)*

4.2 Código Abierto (Open Software)

El concepto código abierto nació como una alternativa al de "free" (libre/gratis) Software que podía tener diferentes significados según el contexto. En la practica ha derivado en un tipo de desarrollo centrado en las ventajas del modelo de aplicaciones que permiten el libre acceso al código fuente.

11 En este contexto debe entenderse Free como "libre" no como " Gratis".

Como en el Software Libre, han de cumplirse con una serie de requisitos para poder ser considerado Código abierto:

- Libre redistribución: El software debe poder ser regalado o vendido libremente.
- Código fuente: El código fuente debe estar incluido u obtenerse libremente.
- Trabajos derivados: La redistribución de modificaciones debe estar permitida.
- Integridad del código fuente del autor: Las licencias pueden requerir que las modificaciones sean redistribuidas sólo como parches.
- Sin discriminación de personas o grupos: Nadie puede dejarse fuera.
- Sin discriminación de áreas de iniciativa: Los usuarios comerciales no pueden ser excluidos.
- Distribución de la licencia: Deben aplicarse los mismos derechos a todo el que reciba el programa.
- La licencia no debe ser específica de un producto: El programa no puede licenciarse solo como parte de una distribución mayor.
- La licencia no debe restringir otro software: La licencia no puede obligar a que algún otro software que sea distribuido con el software abierto deba también ser de código abierto.
- La licencia debe ser tecnológicamente neutral: No debe requerirse la aceptación de la licencia por medio de un acceso por clic de ratón o de otra forma específica del medio de soporte del software.

La organización Open Source Initiative*(29)* es la encargada de promoción de este tipo de software. Aunque los dos movimientos Free y Open Software presentan diferencias de orientación en su filosofía, en la práctica la mayoría de licencias aceptadas por una organización son reconocidas por la otra.

Finalmente, el término FLOSS (siglas de free/libre and open source software, en inglés) engloba ambos tipos de programas.

4.3 Sistemas de votación FLOSS Existentes

Una de las características del desarrollo de proyectos basados en Open/Free Source es que, en muchos casos, no será necesario iniciar una aplicación desde cero; la gran cantidad de proyectos en diferentes fases que permiten, habitualmente, reutilizar o partir de un código existente.

En el momento de plantearse la creación de una nueva aplicación es por tanto indispensable realizar una búsqueda de posibles aplicaciones similares, ya sea en desarrollo o funcionales, para no duplicar esfuerzos innecesariamente.

Tanto en castellano como en catalán la diferencia entre Encuestas o Consultas es bastante clara:

- **Encuesta:** Es un procedimiento de recopilación de datos, generalmente con propósitos estadísticos, dirigido a una muestra escogida dentro de un conjunto de la población. Los participantes y su número son escogidos aleatoriamente para garantizar la fiabilidad del resultado.
- **Consulta Popular:** Es un procedimiento de participación ciudadana por el cual se interroga

a un conjunto de la población sobre una cuestión. Existe, por tanto, un censo que identifica a los posibles votantes y el nivel de participación es relevante en el procedimiento.

Sin embargo este concepto no está tan claramente diferenciado en Inglés y la palabra ***POLL*** engloba ambos tipos/conceptos, esto resulta relevante a la hora de realizar búsquedas ya que este trabajo está orientado al segundo tipo de significado, es decir a la realización de consultas populares.

Se plantea una búsqueda organizada en dos ejes:

- Tratar de encontrar experiencias de voto realizadas mediante aplicaciones de las características especificadas.
- Tratar de encontrar directamente este tipo de aplicaciones.

4.3.1 Experiencias de Voto

Ha sido posible documentar dos experiencias realizadas con este tipo de programas.

District of Columbia Board of Elections and Ethics

Este organismo es una agencia independiente encargada de la administración de las elecciones en el Distrito Columbia (Washington D.C.).

Con la intención de introducir un sistema de votación electrónico (“Digital Vote by Mail”) realizó una serie de pruebas al sistema desarrollado por Open Source Digital Voting Foundation (30).

Sponsor:	Election Type:	Date or Voting Period:	Target Population:	Channel:	Technology Provider:
District of Columbia Board of Elections and Ethics	General Election	Scheduled for November 2, 2010 General Election	UOCAVA voters	Controlled>Electronic Ballot Return>Web Application	Open Source Digital Vote Foundation

Este software llamado TrustTheVote Project (http://wiki.trustthevote.org/index.php/Main_Page) Está patrocinado por empresas como RedHat HP o Sun. Sin embargo las últimas actualizaciones datan del año 2010 y no ha sido posible descargar una versión de pruebas.

Universidad de Lovaina

El único caso de consulta vinculante realizada con un sistema implementado sobre Software de Código Abierto es la elección del Presidente de la Universidad de Lovaina (Marzo del 2009)(21) . El sistema utilizado es Helios e-Votin System y se detalla como parte de los sistemas recopilados mediante búsqueda directa.

En nuestro entorno más cercano y entre las experiencias sobre las que se dispone de información, únicamente se ha podido documentar un solo caso; la consulta realizada en Sant Bartomeu del Grau en el año 2001(31), en la que se empleó una aplicación de código abierto Java FreeVote(32).

4.3.2 Búsqueda Directa

Los proyectos de código abierto se desarrollan de forma colaborativa y, aunque en su inicio puedan iniciarse de forma individual, se estructuran en algún tipo de comunidad, para ello se utilizan los entornos colaborativos conocidos como Forjas (Forges)(33). Este tipo de aplicaciones permiten almacenar y compartir el código, mantener un control de cambios o versiones así como la

comunicación y coordinación de los desarrolladores.

Se han realizado búsquedas en las forjas (30) consideradas como las más importantes[12]:

- Google Code (http://code.google.com/intl/es/)
- Freecode (http://freecode.com/)
- Github (https://github.com/)
- Lunchpad (https://code.launchpad.net/)
- Free Sofware Directory (http://directory.fsf.org/wiki/Main_Page)

Los resultados se han filtrado con los siguientes parámetros:

- **Actualizados durante el año 2012/2013:** Debe tratarse de proyectos activos en los que el proceso de desarrollo y mejora sea constante.
- **Sistemas de votación (consultas):** Debe tratarse de aplicaciones completas que permitan todo el proceso y deben permitir concretamente la realización de consultas.
- **Funcionales y no experimentales:** Debe tratarse de proyectos orientados a la utilización práctica y no a la experimentación de esquemas o algoritmos de votación.

La descripción de los resultados de estas búsquedas y las fichas de los programas analizado puede verse en el ANEXO 9.2 Comparación de Aplicaciones Open Source.
Finalmente se ha seleccionado cuatro aplicaciones:

- Civis Condorcet (http://civs.cs.cornell.edu/)
- Helios (https://vote.heliosvoting.org/)
- Pollen (http://maven-site.chorem.org/pollen/)
- Ágora (https://www.agoravoting.com/)

12 Las búsquedas se desarrollan durante los meses de Marzo/Abril 2013

5 Comparación y análisis de las Aplicaciones

No siempre resulta fácil realizar comparaciones entre programas que pueden tener prestaciones similares. Como ejemplo, qué importancia se le adjudica al tipo de organización escogida o a la licencia utilizada.
El proceso de producción de aplicaciones de Código Abierto tiene diferencias significativas con otros tipos de desarrollos, entre ellas y como ejemplo, el trabajo colaborativo mediante comunidades. Dentro de un proceso de comparación, hay que valorar qué importancia tiene y cómo medimos la influencia de estas comunidades dentro del desarrollo de los dos programas comparados.

De igual manera no resulta menos importante como analizamos el comportamiento de la aplicación desde el punto de vista funcional. En los apartados anteriores hemos analizado algunos de los conceptos de seguridad implicados así como las garantías necesarias que permitan asegurar la corrección formal del proceso de votación que es indispensable para que el resultado sea aceptado.

Todo ello hace necesario necesaria una valoración diferenciada en dos niveles:

- Elementos Free/Open Source: Comparar las características propias del modelo de desarrollo de este tipo de programas.
- Elementos Sistema de Votación: Comparar y analizar el comportamiento de las aplicaciones con relación a los estándares y a las características de seguridad.

Para el análisis de las aplicaciones seleccionadas se utilizarán las herramientas correspondientes al método QSOS (Qualification and Selection of Opensource Software).(34)

5.1 QSOS (Qualification and Selection of Opensource Software).

Este método permite analizar y seleccionar todo tipo de aplicaciones de Código Abierto. Entre las diferentes herramientas proporciona unas plantillas adaptables que permiten recopilar la información de cada una de las aplicaciones y realizar comparaciones en función de los resultados obtenidos.

En nuestro caso adaptaremos la plantilla para que, además de los datos concretos del programa (Metdata) y los correspondientes a su valoración como Open/free Software(Madurez), podamos añadir los tres grupos de datos siguientes:

- Especificaciones IMI: Donde se recogen los datos que figuran en las especificaciones del Instituto Municipal de Informática promotora del proyecto.
- E-Voting: Donde se recogen los datos sobre el comportamiento de la aplicación como Sistema de Votación.
- Métodos de Votación: Donde se reflejan los Métodos de Votación que implementa la aplicación.

Para ello se han traducido las plantillas XML, adaptándolas a las necesidades, creando la siguiente estructura:

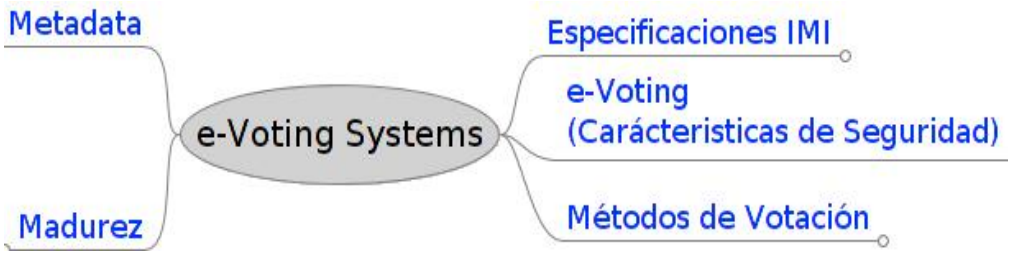


Ilustración 22: Esquema Plantilla QSOS.

5.2 Elementos Sistema de Votación (e-Voting)

Para poder realizar esta valoración disponemos de dos grupos de elementos:

- Las Características de Seguridad
- Los Estándares recomendados por el Consejo Europeo.

Podemos relacionar ambos entendiendo que a partir de los estándares recomendados se cumplen o implementan las características de seguridad. Para ello seleccionamos los 52 estándares que podemos relacionar con una aplicación de e-Voting y los agrupados a partir de la característica de seguridad, teniendo en cuenta que un mismo estándar puede considerarse relacionado con más de una característica.

Como ejemplo, la característica de Elegibilidad puede relacionarse entre otros con el estándar nº39: *"Los votantes deben poder verificar su información en el registro y solicitar modificaciones"*. Se valorará la si aplicación dispone de esta funcionalidad, lo hace parcialmente o no dispone de ella.

Ilustración 23: Esquema estándar nº 39

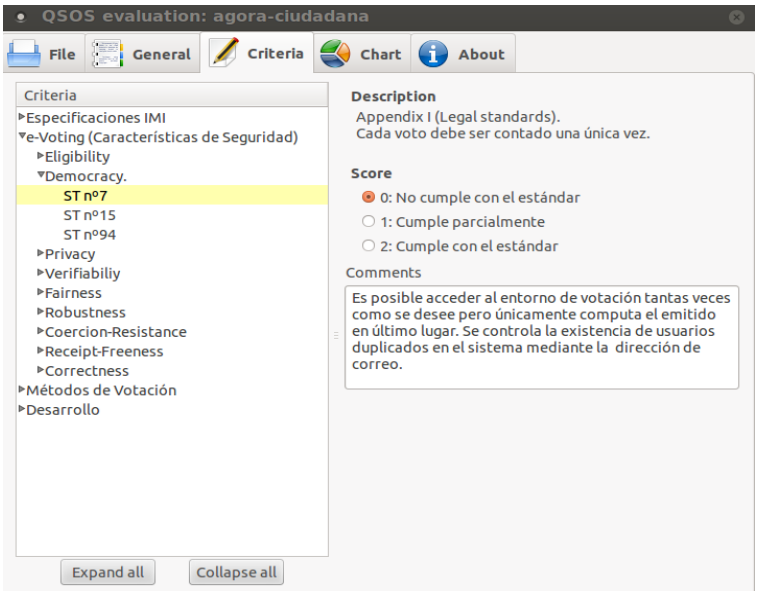
Es importante resaltar que estos estándares especifican comportamientos generales de la aplicación pero no definen la implementación concreta de la aplicación ni los módulos o estructuras necesarios.

Por lo que el cuestionario permite una primera aproximación y por tanto una visión del comportamiento de la aplicación desde un punto de vista general.

Agruparemos los estándares relacionados con cada una de las características de seguridad :

Ilustración 24: Esquema características de seguridad.

A partir de este esquema se genera una plantilla que permite realizar las diferentes valoraciones utilizando un extensión (QSOS) del navegador Firefox:



lustración 25: Formulario de valoración

Los diferentes puntos se valoran de 0 (no se tiene en cuenta) a 2 (se cumple con el punto) a partir de estas valoraciones es posible establecer la comparación entre aplicaciones.

Pruebas realizadas.

Para Comprobar el comportamiento de las aplicaciones se realiza una consulta con cada una de ellas con un grupo de 30 votantes de edades comprendidas entre los 18 y 65 años.

Se utiliza en todos los casos la misma pregunta utilizando la última versión disponible para cada una de ellas. Se realiza la misma pregunta en todas las aplicaciones y el periodo de votación es de una semana con tres días de descanso entre cada una de ellas.

5.3 Resultados de la Comparación

Dos de las aplicaciones analizadas destacan claramente Helios y Ágora Ciudadana tanto en lo que respecta a la madurez de su estado de desarrollo como en el cumplimiento de los estándares.

Como puede verse en los siguientes cuadrantes ambas quedan situadas prácticamente en la misma posición mientras que Pollen[13] con una unas funcionalidades algo menores muestra una menor madurez en su desarrollo.

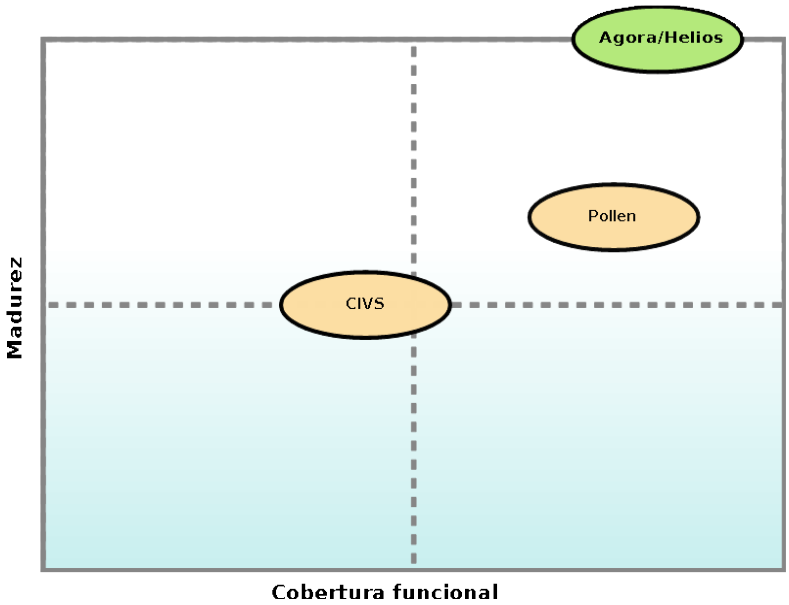


Ilustración 26: Comparación entre las cuatro aplicaciones

Dados estos resultados centraremos la comparación y el análisis en las dos aplicaciones Ágora y Helios que se encuentran mejor situadas.

13 Es importante resaltar que esta aplicación está diseñada como una herramienta para tomar decisiones dentro de ámbitos laborales y por tanto no es estrictamente una aplicación dedicada a e-voting por lo que, en realidad, puede considerarse que presenta un comportamiento muy correcto.

5.4 Comparación General

Ambas aplicaciones presentan resultados similares en lo que ser refiere a los métodos de votación y las especificaciones facilitadas por el IMI(Instituto Municipal de Informática).

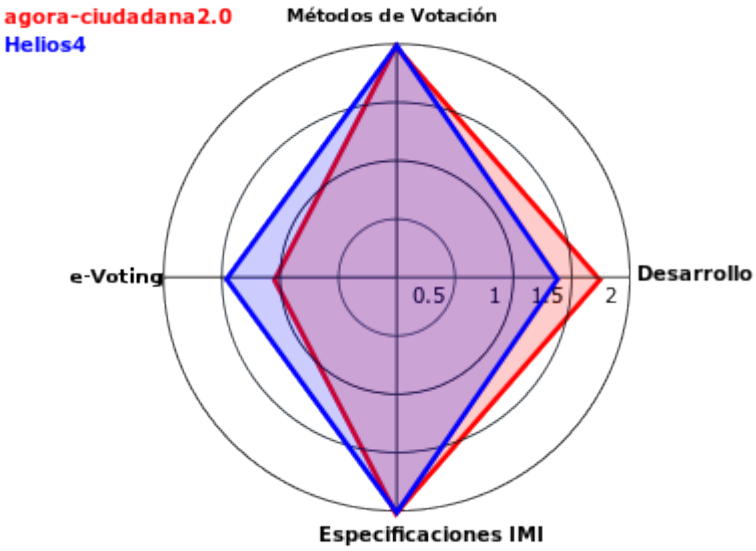


Ilustración 27: Comparación General.

5.5 Nivel de desarrollo

En este aspecto Ágora presenta un mejor nivel de desarrollo ya que dispone de una estructura más clara y consolidada.

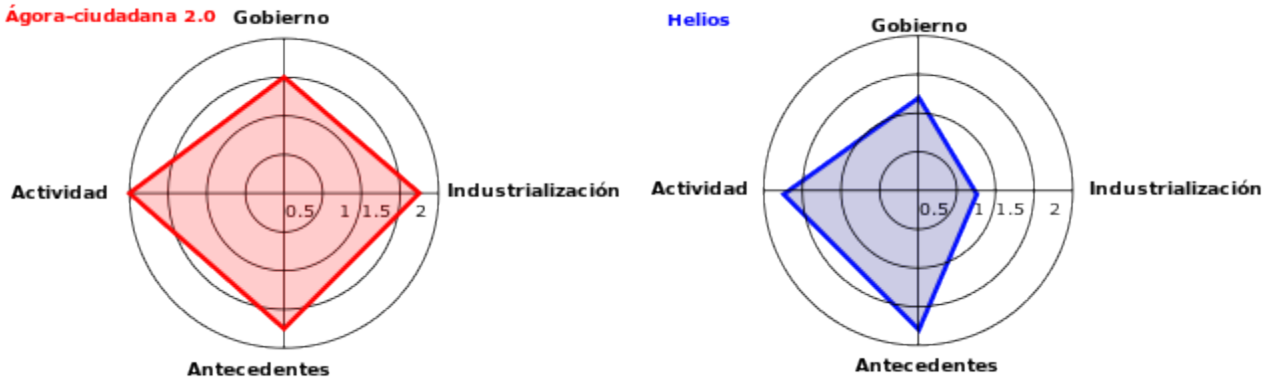


Ilustración 28 Niveles de desarrollo Ágora - Helios

En lo que se refiere a actividad, gobierno y antecedentes las características son muy similares pero en el nivel de industrialización la diferencia es más significativa.

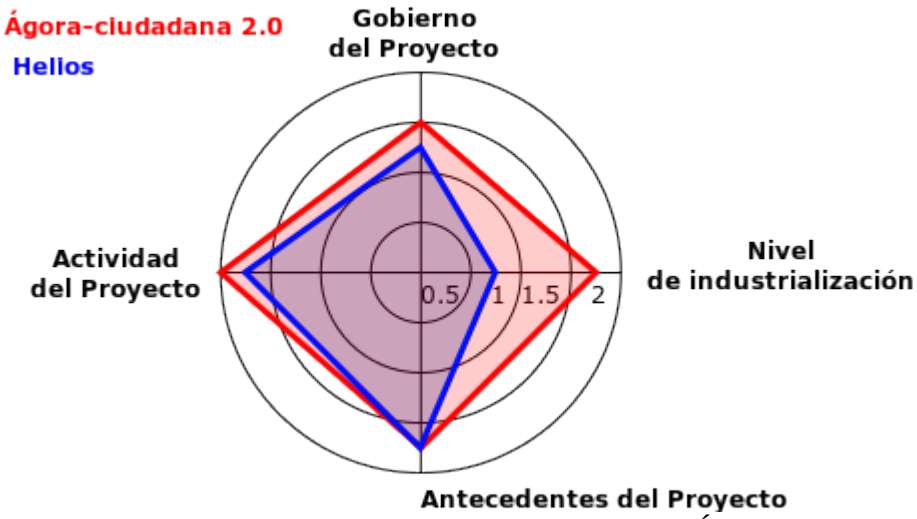


Ilustración 29: Comparación niveles de desarrollo Ágora - Helios

Como puede verse la actividad existente en Ágora es mayor y los procesos de calidad y corrección de errores implementados son más claros y parecen más organizados; finalmente está aplicación dispone de servicios asociados facilitados por la empresa que desarrolla el proyecto.

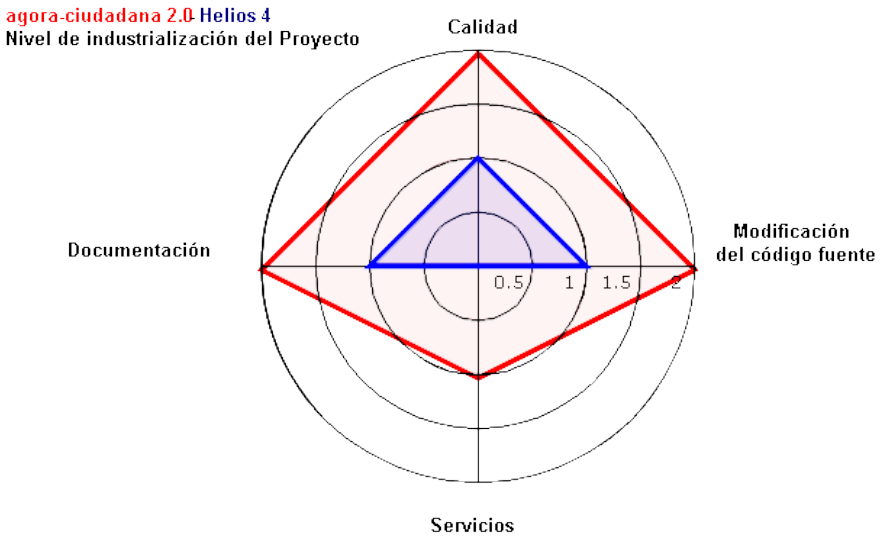


Ilustración 30: Servicios asociados a la aplicación

5.6 Características de Seguridad

En este apartado resulta claramente superior Helios ya está diseñado teniendo en cuenta las características de seguridad especificadas por lo que toda la información se almacena cifrada y existen un procedimiento de auditoría (Privacy, Verifiability).

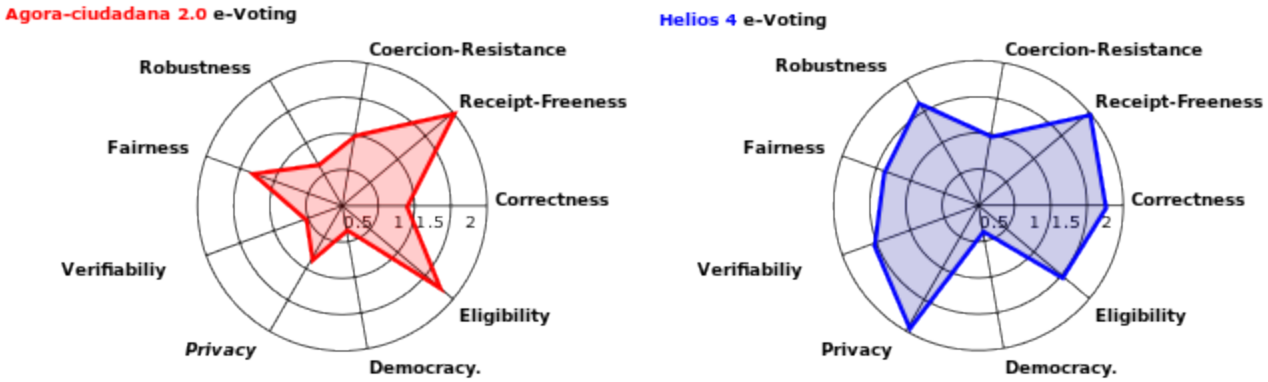


Ilustración 31: Características de Seguridad Ágora - Helios

En ambas aplicaciones es posible depositar más de un voto aunque únicamente se contabiliza el último por lo que en principio no se cumple con la característica Democracy. También es posible añadir votantes una vez iniciada la votación y, aunque se comprueba la identidad de los votantes, no existe ningún mecanismo que valide la procedencia de este listado (Eligibility).

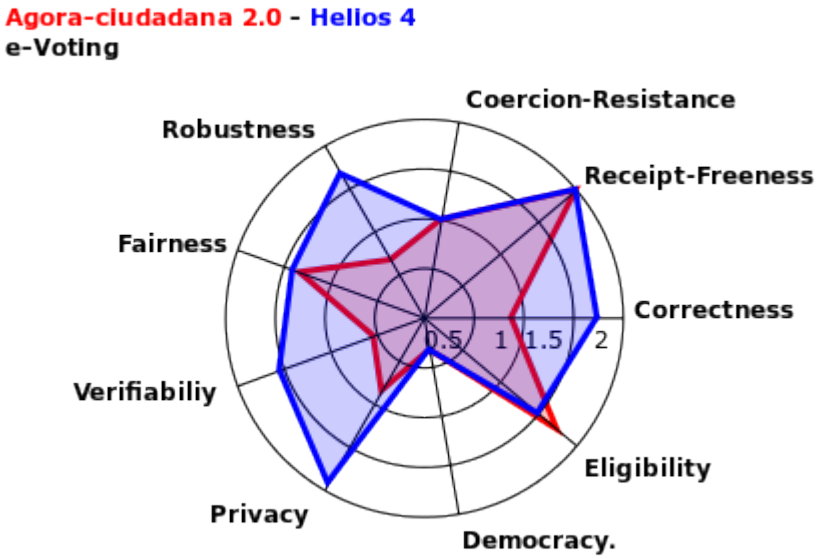


Ilustración 32: Comparación de las Características de Seguridad

En el caso de la aplicación Helios únicamente está disponible en Inglés y no dispone de ninguna adaptación que permita su internacionalización de forma sencilla a diferencia de Ágora que dispone de diferentes idiomas de presentación.

Finalmente ninguno de ellos implementa el estándar recomendado EML .

En general puede decirse que ninguna de las dos aplicaciones representa una alternativa operativa pero ambas presentan aspectos a tener en cuenta en el desarrollo de una aplicación de consultas.

HELIOS:

Cumple las especificaciones indicadas por parte del IMI y con gran parte de los estándares e implementa por tanto las especificaciones de seguridad.

Se distribuye íntegramente bajo licencia GPLV3.

Existe algún antecedente de votación oficial y vinculante con un censo de un tamaño considerable.

La versión actual no está diseñada para permitir su internacionalización o su traducción a otras lenguas.

Aunque existe un sistema de auditoría está limitado al administrador del sistema y no pueden acceder otros usuarios para el seguimiento del proceso sin revelar información sobre los votantes.

Está desarrollada por una fundación y los principales responsables en la creación del código mantienen una sólida actividad académica. Dispone de una pequeña comunidad pero con gran actividad.

ÁGORA:

Cumple las especificaciones indicadas por par te del IMI, aunque el sistema de votación no está, de momento, pensado para cumplir con los estándares especificados y por tanto presenta un comportamiento peor en relación con las características de seguridad.

Como en el caso anterior se distribuye bajo licencia GPLV3 aunque parte de las funcionalidades más avanzadas se ofrecen como servicios adicionales por parte de la empresa desarrolladora.

Dispone de versiones en varias lenguas y su implementación ya contempla la internacionalización.

Se ha realizado alguna experiencia práctica aunque ninguna de gran tamaño o vinculante[14] con esta aplicación. Resultan particularmente interesantes las capacidades de entorno de colaboración social de que dispone que permite su uso por asociaciones o grupos de discusión.

Dispone de una comunidad muy activa: Esta aplicación presenta un desarrollo muy rápido y está previsto implementar alguna de las funcionalidades necesarias.

14 En momento de redacción de este documento están planteadas diferentes iniciativas en este sentido.

Las puntuaciones obtenidas pueden verse en las siguiente tabla:

Community-based QSOS analysis
e-Voting Systems
Synthesis and dynamic comparison

Criterion	agora-ciudadana 2.0	Helios 4
Especificaciones IMI	**2**	**2**
Interfaces	2	2
Licencia	2	2
Redes Sociales	2	2
e-Voting (Características de Seguridad)	**1,04**	**1,46**
Eligibility	1,75	1,5
Democracy.	0,33	0,33
Privacy	0,85	1,92
Verifiabiliy	0,54	1,54
Fairness	1,33	1,4
Robustness	0,67	1,67
Coercion-Resistance	1	1
Receipt-Freeness	2	2
Correctness	0,88	1,75
Métodos de Votación	**2**	**2**
Desarrollo	**1,75**	**1,38**
Antecedentes del Proyecto	1,75	1,75
Actividad del Proyecto	2	1,75
Gobierno del Proyecto	1,5	1,25
Nivel de industrialización del Proyecto	1,75	0,75

Ilustración 33: Tabla de Puntuaciones obtenidas.

6 Conclusiones

6.1 Bases Teóricas para una aplicación de Consultas

No hay duda de que en un futuro el e-voting resultará más común y se introducirá en un mayor número de campos.

Las dificultades de comprobación y seguimiento del proceso son dos de los factores que puede generar desconfianza. Es por tanto muy importante fundamentar su diseño sobre criterios y estándares que permitan asegurar la corrección formal de una consulta realizada con este sistema.

En este sentido el Consejo Europeo ha elaborado una documentación de referencia(1) (4) que describe detalladamente tanto los pasos como los puntos principales a tener en cuenta a la hora de implementar o desarrollar una votación de este tipo. Estos documentos han sido utilizados como referencia en las experiencias realizadas en países de nuestro entorno.

En esta documentación de referencia se destaca la importancia de utilizar aplicaciones de código abierto, ya que permiten acceder a todo el código fuente para estudiar el comportamiento y funcionalidades, además de garantizar independencia del proveedor y abaratar los costes.

Partiendo de un proceso de consulta general y a partir de estándar EML se ha descrito el funcionamiento de una aplicación de E-voting Remoto y los principales conceptos teóricos implicados a partir de una aproximación no formal.

6.2 Sistemas de votación.

Resulta bastante complicado recopilar información sobre experiencias reales relacionadas con el voto electrónico e implementadas con sistemas de votación con licencia Free/Open Sofware.

Pese a las características de observación y comprobación que ofrecen este tipo de aplicaciones, en la mayoría de los casos el código fuente de los sistemas empleados no es público. En otros casos como en las experiencias realizadas en Noruega, es posible acceder al código pero no modificarlo ni ejecutarlo sin restricciones ya que está distribuido bajo una licencia privativa por lo que no puede considerarse de ninguna manera como Código Abierto.

Por otra parte es posible localizar un gran número de iniciativas relacionadas con el voto electrónico dentro del campo de las aplicaciones de código abierto, aunque una gran mayoría no continúan su desarrollo, otras presentan funcionalidades muy completas.

La parte más importante de aplicaciones corresponden a trabajos relacionados con la formalización de protocolos de encriptación y esquemas de votación. Así mismo existe abundante información sobre las características de seguridad necesarias para garantizar la corrección de la aplicación, su descripción y su formalización matemática.

Sin embargo existe una cierta desconexión entre este mundo más académico y la aplicación en Sistemas de Votación que sea posible utilizar en situaciones reales más allá de puntuales pruebas de concepto; con todo estas aplicaciones no están muy alejadas de las funcionalidades necesarias, y parece posible crear un Sistema de Consultas como el planteado en el proyecto.

Como puede verse por las aplicaciones localizadas existen grupos o comunidades con interés y capacidades aunque puede no ser lo suficientemente grande como para sostener un proyecto de forma continuada y en la mayoría de los casos las iniciativas existentes quedan dentro del ámbito académico.

En este sentido, el interés del Ayuntamiento de Barcelona, en desarrollar una herramienta de consultas puede resultar muy útil. Más allá de los recursos necesarios para la implementación del sistema puede permitir por un lado, la creación de una aplicación adecuada a los estándares Europeos y por otro permitir su utilización de forma continuada en entornos reales dentro de los procesos participativos del propio ayuntamiento permitiendo una experiencia y un conocimiento mejor sobre el comportamiento de este tipo de aplicaciones.

En ningún caso se pretende desarrollar una aplicación para agrandes procesos, estos implican un gran despliegue tecnológico y de medios de todo tipo, si no más bien de una aplicación que permita realizar una consulta de forma rápida y económica a grupos de usuarios que, por ejemplo, debido a su número[15] o por su distribución geográfica no puedan reunirse en una sola ubicación. Por otra parte debe tratarse de una herramienta formalmente correcta que asegure la validez de los resultados.

Además del Ayuntamiento de Barcelona existen organizaciones o grupos que pueden necesitar la realización de procesos con una cierta corrección formal pese a no disponer de los medios necesarios.

No se trata de grandes organizaciones que puedan acceder a empresas que prestan este tipo de

15 Reunir un grupo de unos pocos cientos de personad implica ya una cierta logística sin entrar a valorar la necesaria comprobación de identidades o, como ejemplo, la realización de consultas dentro de una empresa con diferentes ubicaciones geográficas.

servicios y que por tanto además de proporcionar la organización disponen de las aplicaciones necesarias, si no más bien, de situaciones en las que sea necesaria la celebración de consultas vinculantes, sin disponer de grandes infraestructuras, que garanticen que se cumple con unas normas y estándares. Como ejemplo, elecciones de colegios profesionales o de asociaciones de todo tipo.

Es importante considerar la utilización de la aplicación en otros ámbitos y su implementación o utilización con unos medios y conocimientos mínimos o, a lo sumo, con el apoyo de una pequeña empresa que aporte los conocimientos técnicos para la instalación y seguimiento de la aplicación.

Esto podría facilitar el conocimiento de los procesos de e-Voting y permitir, por tanto, aumentar la confianza en su utilización, lo que entra dentro de las recomendaciones de CE como un elemento a tener en cuenta dentro del proceso de introducción de e-voting.

6.3 Análisis y comparación de aplicaciones de E-Voting

Uno de los principales problemas que plantea el e-voting radica en el hecho de que no resulta fácil la comprobación y la auditoría de este tipo de aplicaciones.(35)

En los procesos de gran envergadura es posible disponer de los medios adecuados pero, también en este aspecto, puede resultar más complicado en consultas realizadas sobre grupos de menor tamaño; pese a ello y para garantizar la corrección es necesaria la existencia de un seguimiento por parte de todas las partes implicadas.

En este sentido parece necesario el desarrollo de herramientas que permitan la auditoría de una consulta de e-voting por parte de personas sin una gran preparación técnica y en los que, debido a su pequeño volumen, no sea posible disponer de grandes medios.

Esta herramienta de auditoría debe ser independiente del sistema de votaciones, aunque, como parte del desarrollo, se deba tener en cuenta las necesidades de comunicación entre ambas aplicaciones.

Finalmente, dentro del proceso de desarrollo sería necesario tener muy en cuenta e incluso contar con la participación de otros campos académicos en los que la cuestión del voto electrónico está también presente[16].

16 Valgan como ejemplo el interés que pueda generar en campos como las Ciencias Políticas o el Derecho.

7 Propuesta de Desarrollo

A partir de los datos recopilados puede comprobarse que las funcionalidades de las aplicaciones analizadas están muy avanzadas y cumplen, aunque sea en parte, con muchos de los estándares de Consejo Europeo; aun así no parece que ninguna de las dos consideradas más maduras (Ágora y Helios) se encuentren en un estado de desarrollo que permita su implantación de forma inmediata.

Sin embargo y teniendo en cuenta las puntuaciones obtenidas en las pruebas comparativa, la aplicación más correcta formalmente es Helios ya que en su origen implementa las características de seguridad consideradas necesarias. Parece importante primar este aspecto a fin de generar confianza en la aplicación.

Es también importante tener en cuenta que el núcleo de desarrolladores de Helios pertenece sobre todo al mundo más académico y por tanto utilizar esta aplicación puede permitir conectar con entornos reales iniciativas que tienen una aplicación práctica muy limitada y aumentar el conocimiento sobre casos reales.

Por estas razones y sin entrar a definir el modelo de relación o colaboración con la organización que actualmente desarrolla Helios definiremos una propuesta basada en la creación de una nueva versión a partir de la última liberada Helios V4.

Es posible seguir las siguientes alternativas.

A) Crear una aplicación totalmente nueva; tomando como ejemplo la última versión liberada, continuar con la definición formal y desarrollar un sistema a partir de código nuevo.

B) Crear una aplicación a partir de la última versión liberada; en este sentido existirían dos posibilidades.

   - Utilizar el código existente para un desarrollo independiente. La licencia permitiría la reutilización y modificación con la condición de distribuir el nuevo código generado en las mismas condiciones. Como ejemplo sería posible crear una versión traducida o la adaptación para el uso del estándar EML si se considera necesaria su implementación.
   - Colaborar con los equipos de desarrollo existente creando una nueva versión independiente, o bien incluyendo las funcionalidades que se consideren necesarias en la siguiente versión prevista.

7.1 Helios versión Barcelona

El código correspondiente a la última versión (V4) se encuentra disponible bajo la licencia GPLV3 o posterior por lo que es posible utilizarlo con la única condición de distribuirlo bajo esta misma licencia.

Tal como se ha apuntado en las conclusiones no se trata en ningún caso de una aplicación destinada a la celebración de consultas de gran formato si no a procesos definidos como “Low- coercion”(4) es decir elecciones donde el secreto del voto es importante, pero el hecho de votar remotamente es garantía suficiente de privacidad aunque el voto no se realice en entornos controlados.

A partir del estudio comparativo se ha detectado la necesidad de desarrollar las siguientes características:

- Implementar el estándar EML para permitir el intercambio de toda la información de forma estructurada, segura y auditable.
- Se debe implementar funcionalidades que permitan su uso por parte de personas con discapacidades.
- Debe estar disponible en los idiomas necesarios para la consulta y facilitar información que permita la configuración del navegador para que se muestre en el idioma escogido. La herramienta de traducción debe ser independiente.
- Es necesario generar vistas adecuadas para dispositivos móviles como Smartphones o Tablets.
- Crear o mejorar un entorno de auditoría que permita el seguimiento por las personas designadas asegurando qué, disponiendo de información y privilegios dentro del sistema, no comprometan la corrección del proceso.
- Facilitar instrucciones que permitan eliminar toda la formación temporal que se haya generado localmente durante el proceso de votación.
- Disponer de diferentes sistemas de validación de usuarios y de certificación del origen del censo.

Se deberá diseñar el sistema informático que permita alojar la aplicación. Será necesario estudiar las implicaciones que los estándares recomendados y las características de seguridad tienen en la definición de este entorno.

- Sistema Operativo: Dentro de las distribuciones existentes valorar cuál se adapta mejor y crear una configuración específica para el Sistema de Votación.
- Aplicaciones auxiliares: Estudiar que aplicaciones son necesarias para el funcionamiento y seguridad del sistema de votación como:
  - Clientes de Correo
  - Servidores
  - Seguridad

Finalmente sería necesaria la realización del test de comportamiento a fin de dimensionar un sistema que permita pruebas reales. Se implementarían dos sistemas de votación.

- Sistema Institucional: Destinado a consultas promovidas por el propio Ayuntamiento u organizaciones relacionadas.
- Sistemas Público: Se trataría de un sistema abierto que permita la creación de consultas por grupos o entidades.

Distribución de la versión

Tal como se ha apuntado en las conclusiones la necesidad de una herramienta que permita realizar consultas dentro de una corrección formal puede ir más allá de las cubiertas por este proyecto por lo que debería contemplarse la posibilidad de distribuir la versión generada.

A partir del sistema generado para la realización de consultas, se crearían dos soportes de distribución:

- Imagen auto instalable: Esta imagen permitiría su instalación ya sea a partir de una memoria externa o mediante un CD.
- Imagen para entornos de visualización[17]: Esta imagen permitiría su ejecución sobre aplicaciones de entornos virtuales, bien sea un servidor dedicado (Kemu o similar) o sobre entornos facilitados por un proveedor de servicios.

En ambos casos estas imágenes deben ser totalmente operativas y funcionales facilitando la información necearía para su instalación, así como para el uso, auditoría y seguimiento de los procesos de consulta que se realicen.

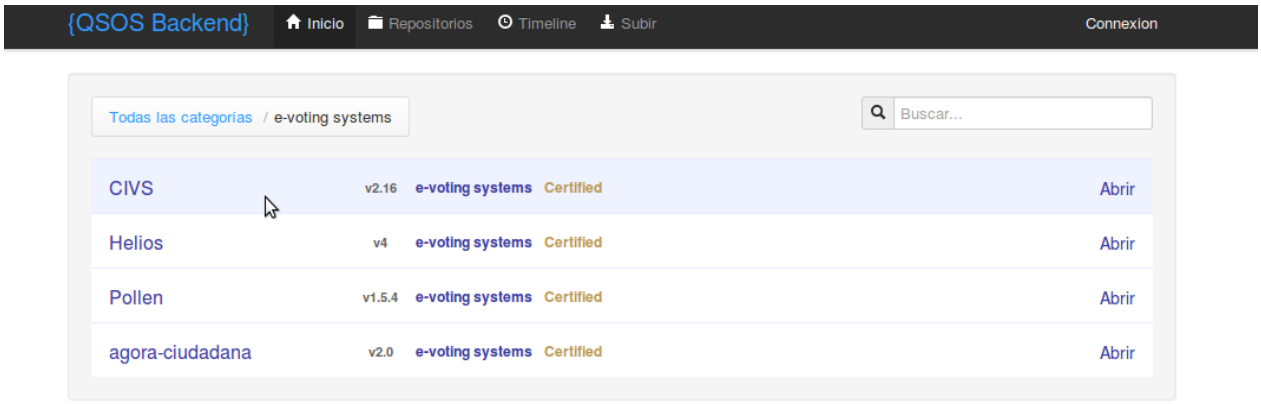
Con estas herramientas debería ser posible, con unos conocimientos medios o con asesoramiento, implementar un sistema de consultas dentro de los estándares del Consejo Europeo, asegurando el cumplimiento de las principales características de seguridad necesarias.

17 Existe un magnifico ejemplo en http://stephane.glondu.net/helios/QUICKSTART.html

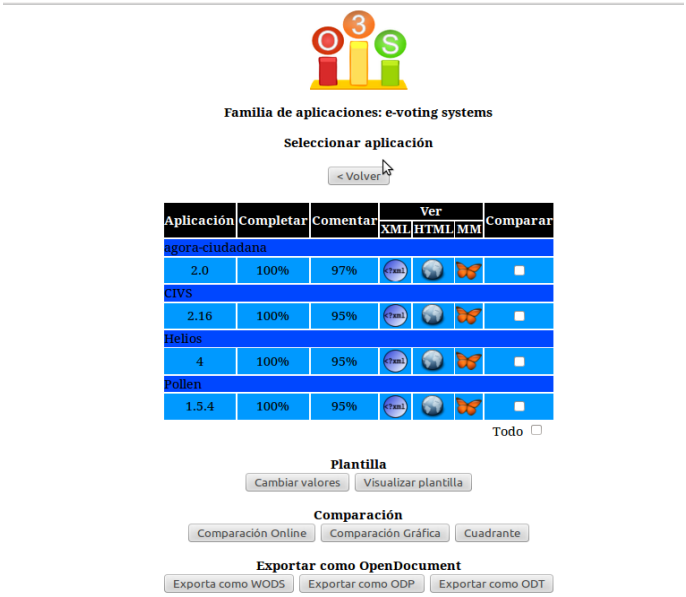
7.2 Herramienta de análisis

Además de las plantillas descritas en el apartado 5.1 el conjunto de herramientas QSOS dispone de dos aplicaciones que funcionan sobre un servidor HTTP.

Backends: Permiten el control de las versiones mediante el programa de control de versiones GIT[18]. En esta aplicación se depositan las diferentes valoraciones que son controladas por el Administrador



O3S: Permite la consulta las diferentes plantillas así como su exportación, facilita además aplicaciones para la comparación de las diferentes aplicaciones y la generación de informes en diferentes formatos y gráficos comparativos como los utilizados en el presente documento.



En conjunto estas herramientas han permitido analizar cada una de las aplicaciones de forma fácil y eficiente.

En el presente proyecto se ha limitado la adaptación de las plantillas a los estándares y características relacionadas con el código de una aplicación de consultas, por lo cual, sería necesario ampliar dicho proyecto al resto de estándares no relacionados de forma estricta con el código de el

18 http://git-scm.com/

sistemas de votación.

Esta herramienta debe servir en un primer momento como parte de la definición formal de la aplicación permitiendo controlar que se cumpla con las especificaciones.

Sin embargo se debería considerar su utilización como parte del sistema de auditoría desarrollando diferentes plantillas que permitieran a terceras partes[19] analizar y certificar su comportamiento ampliando sus funcionalidades al análisis de todo el sistema:

- La instalación del sistema.
- Análisis de riesgos.
- Seguimiento de el proceso de votación.
- Validación de los resultados finales.

Toda esta información debe ser accesible por todas las partes encargadas del seguimiento añadiéndose la generada por el sistema referente a errores e incidencias que puedan producirse a lo largo de las diferentes fases. Para ello será necesario tener en cuenta los actores y fases definidos en el proyecto.

Para la comunicación con el Sistema de Votación se utilizará el estándar EML y debe ser posible la instalación del sistema de auditoría de forma distribuida e independiente.

En todo el proceso de desarrollo, uno de los parámetros fundamentales a tener en cuenta es que la aplicación debe ser utilizada por personas sin una gran preparación técnica. Para ello parece necesario potenciar la transversalidad, de manera que participen posibles usuarios que se tengan presente aspectos legales o formales más allá de los contemplados en el presente proyecto.

19 Durante la realización de las evaluaciones correspondientes al presente documento se han realizado pruebas en las que personas sin una preparación informática especializada han evaluado el comportamiento de las aplicaciones sin grandes dificultades.

8 Bibliografía

1. COUNCIL OF EUROPE and COMMITTEE OF MINISTERS. *Legal, operational and technical standards for e-voting: recommendation Rec(2004)11 adopted by the Committee of Ministers of the Council of Europe on 30 September 2004 and explanatory memorandum.* Strasbourg : Council of Europe Pub., 2005. ISBN 9287156352 9789287156358. 00000

2. BALZAROTTI, Davide, BANKS, Greg, COVA, Marco, FELMETSGER, Viktoria, KEMMERER, Richard, ROBERTSON, William, VALEUR, Fredrik and VIGNA, Giovanni. Are Your Votes Really Counted ? Testing the Security of Real-world Electronic Voting Systems. [online]. 2007. [Accessed 1 April 2013]. Available from: dl.acm.org/citation.cfm?id=1390630.13906600000Cited by 0000

3. E-Voting: Big Risks for Small Gains. [online]. [Accessed 23 June 2013]. Available from: http://www1.cs.columbia.edu/~unger/articles/e-voting1-11-07.html00000

4. CAARLS, Susanne and COUNCIL OF EUROPE. *E-voting handbook: key steps in the implementation of e-enabled elections*. Strasbourg : Council of Europe Pub., 2010. ISBN 9789287169488 9287169489. 00009

5. U.S. ELECTION ASISTANCE COMISION. *A Survey of Internett Voting*. 14 September 2011. Cited by 0000

6. Web Accessibility Initiative (WAI) - home page. [online]. [Accessed 2 December 2013]. Available from: http://www.w3.org/WAI/00000

7. EML v5.0 Process & Data Requirements. [online]. [Accessed 30 June 2013]. Available from: http://docs.oasis-open.org/election/eml/v5.0/EML-Process-Data-Requirements-v5.0.htmlCited by 0000

8. *OASIS | Advancing open standards for the information society* [online]. [Accessed 2 December 2013]. Available from: https://www.oasis-open.org/

9. SCIENCE, Information and AUGUST, Updated. Civitas: A Secure Remote Voting System. 2007. P. 1–47. 000000000

10. CLARKSON, Michael R, CHONG, Stephen and MYERS, Andrew C. Civitas: Toward a Secure Voting System. 2008. Vol. 7875, no. August 2007. 013800154

11. SAMPIGETHAYA, Krishna and POOVENDRAN, Radha. A framework and taxonomy for comparison of electronic voting schemes. *computers & s e c u rity* [online]. 8 November 2005. P. 1 3 7 – 1 5 3. [Accessed 6 June 2013]. Available from: http://www.journals.elsevier.com/computers-and-securityJCS-05.pdf00054

12. FOUARD, Laure, DUCLOS, Mathilde and LAFOURCADE, Pascal. *Survey on Electronic Voting Schemes*. 00004

13. Asymmetric-Key Cryptography. [online]. [Accessed 2 December 2013]. Available from: http://www.cs.cornell.edu/courses/cs513/2007fa/TL04.asymmetric.html00008

14. *Meier Andreas The ElGamal Cryptosystem.pdf* [online]. [Accessed 3 April 2013]. Available from: http://wwwmayr.in.tum.de/konferenzen/Jass05/courses/1/presentations/Meier%20Andreas%20The%20ElGamal%20Cryptosystem.pdf

15. QUISQUATER, Jean-Jacques, QUISQUATER, Myriam, QUISQUATER, Muriel, MICHAËL QUISQUATER, GUILLOU, Louis, GUILLOU, Marie Annick, GUILLOU, Gaïd, GUILLOU, Anna, GUILLOU, Gwenolé and GUILLOU, Soazig. How to explain zero-knowledge protocols to your children. In : *Advances in Cryptology—CRYPTO'89 Proceedings* [online]. 1990. p. 628–631. [Accessed 16 July 2013]. Available from: http://link.springer.com/chapter/10.1007/0-387-34805-0_600132

16. HENRY, Kevin. The Theory and Applications of Homomorphic Cryptography. 2008. 000300004

17. ZAGORSKI, Filip. *Introduction to Electronic Voting ´ rski Filip Zag o*. 2012. 000000000

18. RJA\VSKOVÁ, Zuzana. *Electronic voting schemes* [online]. Diplomova praca. Bratislava : Comenius University,, 2002. [Accessed 4 June 2013]. Available from: http://people.ksp.sk/~zuzka/elevote.pdf00000

19. GREEN-ARMYTAGE, James. A Survey of Basic Voting Methods. *A Survey of Basic Voting Methods* [online]. [Accessed 30 May 2013]. Available from: http://www.econ.ucsb.edu/~armytage/voting/survey.htm00004

20. CHEVALLIER-MAMES, DAVID POINTCHEVAL, JULIEN STERN and JACQUES TRAORE. *On Some Incompatible Properties of Voting Schemes*. 00028

21. ADIDA, Ben, MARNEFFE, Olivier De, OLIVIER PEREIRA and JEAN-JACQUES QUISQUATER. Electing a University President using Open-Audit Voting : Analysis of real-world use of Helios. 2009. No. i, p. 1–15. 006100077

22. BARCELÓ, Bartolomé. Sistemas Electorales. *Materials Matemátics*. 4 July 2007. Vol. 2007, no. 7, p. 24. El problema que se trata en este artıculo es elde como interpretar los resultados de una elecciono, incluso antes de tener ningun resultado, el de responder,si tenemos que elegir de entre varios candidatos,a ¿como se puede disenar un procedimientopara escoger "el mejor"?.00000

23. Voting Methods (Stanford Encyclopedia of Philosophy). [online]. [Accessed 30 May 2013]. Available from: http://plato.stanford.edu/entries/voting-methods/

24. http://www.vvk.ee/voting-methods-in-estonia/engindex/. 00000

25. Hjem - Dokumentasjon. [online]. [Accessed 5 January 2014]. Available from: https://brukerveiledning.valg.no/Dokumentasjon/Kildekode.aspx

26. Fundación Copyleft ¿Qué es copyleft? [online]. [Accessed 2 December 2013]. Available from: http://fundacioncopyleft.org/es/9/que-es-copyleft00000

27. El Proyecto FreeBSD. [online]. [Accessed 2 December 2013]. Available from: http://www.freebsd.org/es/00000

28. La Llicència Pública General de GNU v3.0 - Projecte GNU - Free Software Foundation (FSF). [online]. [Accessed 2 December 2013]. Available from: http://www.gnu.org/licenses/gpl.html00000

29. The Open Source Initiative | Open Source Initiative. [online]. [Accessed 2 December 2013]. Available from: http://opensource.org/00043

30. TrustTheVote – An OSDV Foundation Project » The Project. [online]. [Accessed 2 December 2013]. Available from: http://www.trustthevote.org/background00000

31. AULET, CARLES VIÑAS. *Consultes populars electròniques Sant Bartomeu del Grau*. 2002. 0000Cited by 0000

32. JFreeVote Project. [online]. [Accessed 13 April 2013]. Available from: http://web.dit.upm.es/~jantonio/voto_electronico/jfreevote-1.0/

33. Una docena de forjas para el desarrollo de software libre y colaborativo en la Administración - una docena de. [online]. [Accessed 18 May 2013]. Available from: http://unadocenade.com/una-docena-de-forjas-para-el-desarrollo-de-software-libre-y-colaborativo-en-la-administracion/

34. Collaborative technological watch. [online]. [Accessed 2 December 2013]. Available from: http://www.qsos.org/00000

35. PROCEEDINGS OF THE SEMINAR ON NETWORK SECURITY. [online]. [Accessed 16 June 2013]. Available from: http://www.coe.int/t/dgap/democracy/Activities/GGIS/E-voting/00000

36. DIAZ, Jose Angel, PACHECO, Manuel Velardo, GONZALEZ-BARAHONA, Jesus M. and FELIPE ORTEGA SOTO. *Forjas: entornos de desarrollo colaborativo.* [online]. 2009. ©2009 Cenatic. [Accessed 18 May 2013]. Available from: http://comunidad.cenatic.es00000

37. File:CriptografiaAsimetrica.png - Wikimedia Commons. [online]. [Accessed 2 December 2013]. Available from: http://commons.wikimedia.org/wiki/File:CriptografiaAsimetrica.png00000

9 Anexos

9.1 Ejemplo Métodos de Votación

Partiendo de unos datos como los de la siguiente tabla veremos el comportamiento de alguno de los métodos considerados como más representativos.

	nº de votos en miles		
	6	5	4
Posición 1	A	C	B
Posición 2	B	B	C
Posición 3	C	A	A

Se presentan tres Opciones (A,B,C) que es posible ordenar en la papeleta según el orden de preferencia adjudicada a cada una de ellas de mayor a menor. En esta consulta se han emitido un total de 15.000 (6+5+4=15) votos repartidos según se indica en la primera fila y con las opciones ordenadas de la manera que se refleja en cada columna.

Método Mayoritario.

Si utilizamos el método **Mayoritario** y tomamos como referencia el primera posición de los votos emitidos, la opción ganadora sería la A con 6.000 votos.

	nº de votos en miles		
	6	5	4
Posición 1	A	C	B

Es importante destacar que en este caso la opción ganadora únicamente ha sido escogida por el 40% de los votantes, es decir menos de la mitad. Para mejorar este comportamiento es posible realizar una segunda vuelta en la que únicamente participan los dos primeros clasificados en la primera votación.

Métodos Preferenciales

Los métodos llamados **Preferenciales** permiten escoger entre diferentes opciones teniendo en cuenta el orden (preferencia) en que se escogen. Entre este tipo de métodos consideraremos únicamente dos de los más conocidos.

Método Condorcet

Este método fue enunciado por **Marie Jean Antoine Nicolas de Caritat, Marques de Condorcet** del que toma el nombre. De forma resumida el ganador es el que, enfrentados los candidatos dos a dos, gana a todos los otros candidatos.

Con los datos de la tabla de ejemplo si contabilizamos únicamente los votos del ganador en enfrentamientos entre las opciones:

A contra C ==> ganador C (6-9)

	nº de votos en miles		
	6	5	4
Posición 1	A	C	
Posición 2			C
Posición 3	C	A	A

Opción	votos Col 1	votos Col 2	votos Col 3	Total
A	6	0	0	**6**
C	0	5	4	**9**

A contra B ==> ganador B (6-9)

	nº de votos en miles		
	6	**5**	**4**
Posición 1	**A**		**B**
Posición 2	**B**	**B**	
Posición 3		**A**	**A**

Opción	votos Col 1	votos Col 2	votos Col 3	Total
A	6	0	0	**6**
B	0	5	4	**9**

B contra C ==> ganador B (10-5)

	nº de votos en miles		
	6	**5**	**4**
Posición 1		**C**	**B**
Posición 2	**B**	**B**	**C**
Posición 3	**C**		

Opción	votos Col 1	votos Col 2	votos Col 3	Total
B	6	0	4	**10**
C	0	5	0	**5**

Según lo expuesto ganaría la Opción B, ya que vence en la comparación con cada una de las otras dos opciones individualmente.

No siempre existe un ganador Condorcet y puede darse el caso de ninguna de las opciones haya ganado a todas las otras y se presenten empates múltiples.

Método Borda

Recibe el nombre del matemático francés **Jean-Charles Borda.** Se puntúan las opciones según el lugar que ocupan en las listas; así en el ejemplo:

- 1º puesto 3 puntos.
- 2º puesto 2 puntos.
- 3º puesto 1 punto.

Según este método también sería la opción B la ganadora.

	nº de votos en miles		
	6	**5**	**4**
Posición 1	**A**	**C**	**B**
Posición 2	**B**	**B**	**C**
Posición 3	**C**	**A**	**A**

				Total
A	3	1	1	5
B	2	2	3	**7**
C	1	3	2	6

9.2 Búsqueda de Aplicaciones

En la investigación preliminar y a partir de los datos considerados relevantes de los posibles programas de votación localizados se recopilan los siguientes datos:

General	**Nombre**	**JFreevote**
	Web	http://web.dit.upm.es/~jantonio/voto_electronico/jfreevote-1.0/
	Caso Real	si
	Cumple Espc	si
Código	**Licencia**	GPL
	Web descarga	http://sourceforge.net/projects/freevote/files/JFreeVote/
	Descarga	si
	Versión	1.0.b1
	Fecha última Actualización	2006
Tecnología	**Funcional**	no
	Lenguaje	Java
	Traducción	Si Es.ES
	Móviles	no
	Redes Sociales	no
Organización	**Estructura**	Desarrollado por Hispalinux
	Comunidad	Lista de Correos
	Tamaño	no
	Activa	si
	Soporte a Usuario	si

9.2.1 Experiencias de Voto

Únicamente se ha podido documentar dos casos de consulta vinculante realizadas con un sistema implementado sobre Software de Código Abierto.

Una es la elección del Presidente de la Universidad de Lovaina (Marzo del 2009) (21). El sistema utilizado es Helios e-Votin System y se detalla como parte de los sistemas recopilados mediante búsqueda directa.

La segunda forma parte de los casos recopilados por parte de la U.S. Election Assistance Commission y ser refiere a la experiencia realizada en el Distrito de Columbia durante las elecciones generales del año 2010 para facilitar el voto a personas desplazadas. Se utiliza la aplicación facilitada por la Open Source Digital Vote Foundation y que se encuentra paralizado.

En nuestro entorno más cercano y entre las experiencias sobre las que se dispone de información, únicamente se ha podido documentar un solo caso, resaltado en verde en la tabla, en el que se empleó con una aplicación de código abierto Java FreeVote.

TÍTULO DE LA CONSULTA	ORGANISMO	FECHA	CONSULTA/ELECCIÓN	VINCULANTE	SISTEMA	CÓDIGO ABIERTO
Ajuntament d'Amposta	Ajuntament D'Amposta	2009	Consulta		Desconocido	desconocido
Elecciones de Rep de los Colectivos Universitarios a la Junta de la Escuela Técnica Superior de Ingeniería (ETSE)	Universitat Rovira i Virgili	2009	Elección	si	Scytl	no
Reforma de la Diagonal	Ayuntament de Barcelonoa	2010	Consulta	si	Scytl/Indra	no
Elecciones de los Órganos Colegiados de la Universidad	UPC	2008	Elección	si	Scytl	no
Elecció de l'Ajuntament Jove de Llavaneres	Ajutament Llavaners	2008	Elección	si	Scytl	no
Madrid participa	Ayuntamiento de Madrid	2004	Consulta	si	Scytl, Accenture, Oracle, Telefónica, HP, Intel)	no
Consulta Independentista (Alella)	Alella Decideix	2010	Consulta	no	Desconocido	desconocido
Elección de representantes	Colegio Ingenieros Telecos Valencia	2008	Elección	si	ODEC	no
Junta de Gobierno	Colegio de Ingenieros Técnicos Industriales de Barcelona (CETIB)	2005	Elección	si	scytl	no
Varias Elecciones	UAB	2010/12	Elección	si	Scytl	no
Consulta Ajuntament Sant Batomeu de Grau	Ajuntament	2001	Consulta	no	FreeVote	SI

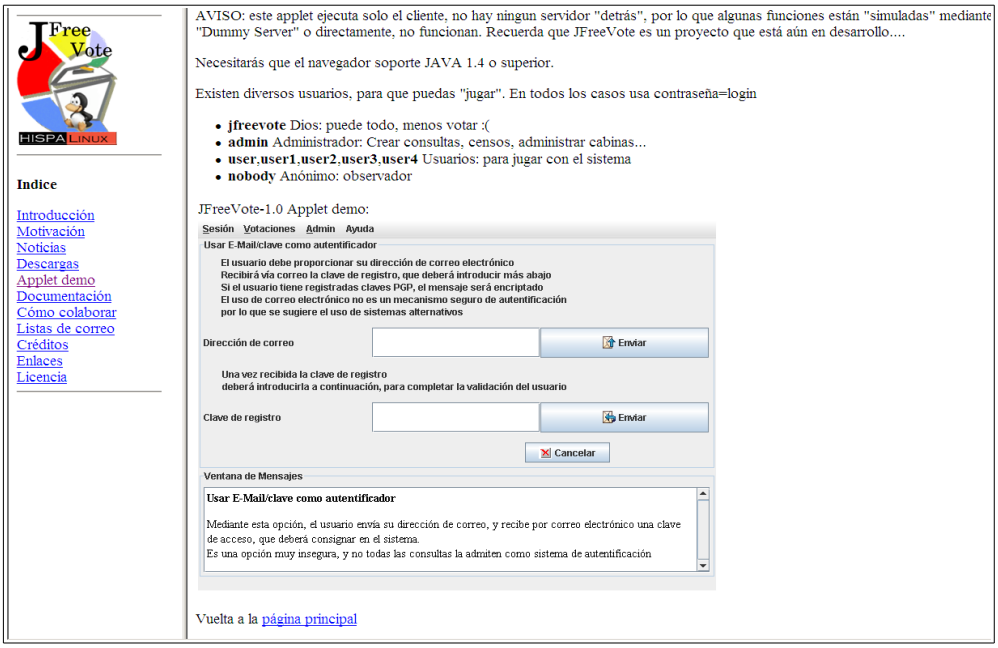
Jfreevote

Únicamente se hizo servir en algunas pruebas de concepto como el caso de la Sant Bartomeu del Grau. (31)

Desarrollada en Java la versión disponible es del año 2006 por lo que está totalmente obsoleta. Sin embargo hay que destacar que se trata de una aplicación con unas funcionalidades muy completas tanto con una configuración remota como en configuración *Cabina* es decir en voto presencial.

Existe una instancia de pruebas como Applet Java en:

http://web.dit.upm.es/~jantonio/voto_electronico/jfreevote-1.0/



9.2.2 Búsqueda Directa

Los proyectos de código abierto se desarrollan de forma colaborativa y, aunque en su inicio puedan iniciarse de forma individual, se estructuran en algún tipo de comunidad, para ello se utilizan los entornos colaborativos conocidos como Forjas (Forges) (36). Este tipo de aplicaciones permiten almacenar y compartir el código, mantener un control de cambios o versiones así como la comunicación y coordinación de los desarrolladores.

Se han realizado búsquedas en las forjas (33) consideradas como las más importante[20]s:

- Google Code (http://code.google.com/intl/es/)
- Freecode (http://freecode.com/)
- Github (https://github.com/)
- Lunchpad (https://code.launchpad.net/)
- Free Sofware Directory (http://directory.fsf.org/wiki/Main_Page)

20 Las búsquedas se desarrollan durante los meses de Marzo/Abril 2013

Los términos utilizados son los siguientes:

- Voting
- e-Voting
- election
- Poll

El total de resultados obtenidos en cada caso son:

	Google	Freecode	Github	Lauchpad	The Free Software Directory
E-Voting	18	0	41	1	20
Voting	573	46	1515	39	17
Election	187	10	591	14	
Poll	169	61	1453	40	
Total	947	117	3600	94	37

Así mismo se ha realizado búsquedas en las siguientes forjas de tamaño menor:

- Joinup (Comisión Europea)
- Savannah (GNU)
- Gna! (GNU)

Los resultados se han filtrado con los siguientes parámetros:

- **Actualizados durante este año 2012/2013:** Debe tratarse de proyectos activos en los que el proceso de desarrollo y mejora sea constante.
- **Sistemas de votación (consultas):** Debe tratarse de aplicaciones completas que permitan todo el proceso y deben permitir la realización de consultas.
- **Funcionales y no experimentales:** Debe tratarse de proyectos orientados a la utilización práctica y no a la experimentación de esquemas o algoritmos de votación.
- Los sistemas encontrados se detallan en la siguiente tabla:

General	Código			Tecnologia
Nombre	Licencia	Versión	Fecha última actualización	Funcional
Civis Condorcet	BSD	2.16	2012	si
eVote/Clerk	eVote®/Clerk License	2.52	2013	no
VoteEngine	no (BSD)	0.99	2013	Experimental
Helios	GNU	3.0	2013	si
Pi-Vote	Pyrate Partei (MIT)	1.4	2013	no
Ulyssesvoting	GNU	0.1	2013	Experimental
Glosowania	WIN	0.2 alpha	2013	Experimental
VotingEngine	ninguna		2012	
Halalan	GNU	1.6.1	2012	si
Opavote	Open STV licence		2012	
Votorola	MIT	0.2.3	2011	
Pollen	GPL	1.5.3	2013	si
Agora Ciudadana	LGPL	2.0	2013	si
I-election	EllisLab, Inc.	1.0	2013	
Evote	3-clause BSD License	0.1	2013	Experimental
Solon-voting	GNU	0.1	2013	Experimental
Onlinevoting	ninguna	0.1	2013	Experimental

Los sistemas resaltados en rojo presentan licencias que no son compatibles ni reconocidas como Open/Free Software o, en algún caso, no figura ningún tipo de licencia:

General	Código			Tecnologia
Nombre	Licencia	Versión	Fecha última actualización	Funcional
Glosowania	WIN	0.2 alpha	2013	Experimental
VotingEngine	ninguna		2012	
Opavote	Open STV licence		2012	
I-election	EllisLab, Inc.	1.0	2013	
Onlinevoting	ninguna	0.1	2013	Experimental

Los sistemas resaltados en amarillo son aplicaciones experimentales o, pese a ser sistemas de votación, con funcionalidades limitadas:

General	Código			Tecnologia
Nombre	Licencia	Versión	Fecha última actualización	Funcional
eVote/Clerk	eVote®/Clerk License	2.52	2013	no
VoteEngine	no (BSD)	0.99	2013	Experimental
Pi-Vote	Pyrate Partei (MIT)	1.4	2013	No Cliente
Ulyssesvoting	GNU	0.1	2013	Experimental
Evote	3-clause BSD License	0.1	2013	Experimental
Solon-voting	GNU	0.1	2013	Experimental

Finalmente, los sistemas resaltados en verde son los qué se han analizado con más detalle ya que se trata de aplicaciones con una licencia dentro de los términos especificados y con unas funcionalidades, en principio, correctas.

General	Código			Tecnologia
Nombre	Licencia	Versión	Fecha última actualización	Funcional
Civis Condorcet	BSD	2.16	2012	si
Helios	GNU	3.0	2013	si
Halalan	GNU	1.6.1	2012	si
Votorola	MIT	0.2.3	2011	
Pollen	GPL	1.5.3	2013	si
Agora Ciudadana	LGPL	2.0	2013	si

9.2.3 Sistemas Evaluados

Con cada uno de los sistemas se ha seguido el siguiente procedimiento:

- Descarga del código y comprobación de la licencia.
- Instalación: Se ha efectuado una instalación mínima con una prueba de comunicación local.
- Test de las funcionalidades, mediante alguna votación, de los elementos de acceso a la aplicación especificados:
    - Interfaz de Administración.
    - Interfaz de Usuario Votación.
    - Interfaz de Usuario Resultados.
    - Redes Sociales.

Civis Condorcet

General	Nombre	Civis Condorcet
	Web	http://www.cs.cornell.edu/andru/civs.html
	Caso Real	no
	Cumple Epec	si
Código	Licencia	BSD
	Web descarga	http://www.cs.cornell.edu/w8/~andru/cgi-bin/download.pl
	Descarga	si
	Versión	2.16
	Fecha última Actualización	2012
Tecnología	Funcional	si
	Lenguaje	HTML CSS PERL
	Traducción	Si Es.ES
	Móviles	no
	Redes Sociales	no
Organización	Estructura	The Cornell Computer Science Department.
	Comunidad	Sin estructura Mail List
	Tamaño	no
	Activa	no
	Soporte a Usuario	Si Lista de Correo

Desarrollado en el Departamento de Computación de la Universidad de Cornell es una aplicación que permite realizar consultas sobre una cuestión o elecciones; el método de votación es el sistema Cordoncet[21].
Existe una instancia que permite probar la aplicación realizando consultas públicas. (http://www.cs.cornell.edu/w8/~andru/civs/).

Interfaz de Administración: Permite la creación de la consulta con diferentes características (privada, pública, detalle de los resultados) una vez creada se notifica a los votantes mediante correo electrónico.
Interfaz de usuario Votación: La notificación de la votación contiene un enlace a la votación.
Interfaz de usuario Resultados: Después de realizarse el voto es posible acceder a la página de resultados que permite seguir la evolución de la consulta.
Redes Sociales: No dispone de ninguna funcionalidad relacionada con redes sociales.
Instalación: Se realizan pruebas sobre una instancia de desarrollo; para la realización de pruebas más completas es necesaria su instalación sobre un servidor web (Apache o similar).

21 http://es.wikipedia.org/wiki/M%C3%A9todo_de_Condorcet

Helios

General	**Nombre**	**Helios**
	Web	http://heliosvoting.org/
	Caso Real	si
	Cumple Espc	si
Código	**Licencia**	GNU
	Web descarga	https://github.com/benadida/helios-server
	Descarga	si
	Versión	V.4
	Fecha última Actualización	2013
Tecnología	**Funcional**	si
	Lenguaje	Python
	Traducción	no
	Móviles	si
	Redes Sociales	si
Organización	**Estructura**	Fundación
	Comunidad	Lista de Correos
	Tamaño	Pequeña
	Activa	SI
	Soporte a Usuario	Si Wiky, manual de instalación mailist Google Groups

El principal desarrollador es Ben Adida[22]. Este sistema se ha empleado en 2009 en la elección del presidente de la Universidad de Lovaina. Existe numerosa documentación sobre otras pruebas realizadas. Está estructurado en una Fundación y dispone de una comunidad pequeña pero muy activa. Permite realizar consultas con múltiples preguntas y respuestas. El método de votación es únicamente mayoritario.

Existe una instancia que permite realizar consultas públicas en http://heliosvoting.org/.

Interfaz de Administración: Permite la creación de la consulta con diferentes características (privada, pública, detalle de los resultados) una vez creada se notifica a los votantes mediante correo electrónico.
Interfaz de usuario Votación: La notificación de la votación contiene un enlace a la votación. Permite la identificación de los usuarios mediante diferentes redes sociales y la verificación de la encriptación de la papeleta.
Interfaz de usuario Resultados: Son liberados por el administrador de la consulta y anunciados mediante correo electrónico.
Redes Sociales: Dispone de plugins para la publicación de los resultados.
Instalación: Se realizan pruebas sobre una instancia de desarrollo. Para la realización de pruebas más completas es necesaria su instalación sobre un servidor web (Apache o similar).

22 http://www.linkedin.com/in/benadida

Halalan

General	Nombre	Halalan
	Web	http://code.google.com/p/halalan/
	Caso Real	si
	Cumple Espc	si
Código	Licencia	GNU
	Web descarga	http://code.google.com/p/halalan/
	Descarga	si
	Versión	1.6.1
	Fecha última Actualización	2012
Tecnología	Funcional	si
	Lenguaje	CSS
	Traducción	ENG
	Móviles	si
	Redes Sociales	si
Organización	Estructura	University of the Philippines Linux Users' Group (UnPLUG)
	Comunidad	SI
	Tamaño	Pequeña
	Activa	SI
	Soporte a Usuario	SI Wiky, Bolg, Maillit

Desarrollado en University of the Philippines Linux Users' Group (UnPLUG) se ha utilizado en la realización de diversas consultas en esa misma universidad.
Es un sistema adaptado específicamente para realizar elecciones.

Interfaz de Administración: Permite la creación de los diferentes elementos necesarios para realizar las elecciones; como candidatos, partidos o posiciones; una vez creada la elección se notifica a los votantes mediante correo electrónico.
Interfaz de usuario Votación: La notificación de la votación contiene un enlace a la votación.
Interfaz de usuario Resultados: Son liberados por el Administrador de la consulta y anunciados mediante correo electrónico.
Redes Sociales: No dispone de ninguna funcionalidad relacionada con redes sociales.
Instalación: Se realizan pruebas sobre una instancia de desarrollo, y para la realización de pruebas más completas es necesaria su instalación sobre un servidor web (Apache o similar).

Votorola

General	**Nombre**	**Votorola**
	Web	http://zelea.com/project/votorola/home.html
	Caso Real	no
	Cumple Espc	no
Código	**Licencia**	MIT
	Web descarga	http://zelea.com/var/backups/votorola/
	Descarga	si
	Versión	0.2.3
	Fecha última Actualización	**2011**
Tecnología	**Funcional**	java
	Lenguaje	no
	Traducción	no
	Móviles	no
	Redes Sociales	si
Organización	**Estructura**	
	Comunidad	si
	Tamaño	pequeña
	Activa	si
	Soporte a Usuario	si

Se trata de un grupo de programas para implementar plataformas de colaboración y deliberativas, entre sus funcionalidades se encuentra la capacidad de realizar consultas.

Se trata de un software realmente complejo ideado para estructurar diferentes áreas y sistemas de votación.

Una vez analizado más en profundidad se descarta su instalación ya que las funcionalidades de votación son únicamente una parte de todo el sistema más enfocado a la organización de grupos.

Pollen

General	**Nombre**	**Pollen**
	Web	http://maven-site.chorem.org/pollen/
	Caso Real	no
	Cumple CE	SI
Código	**Licencia**	GPL
	Web descarga	http://www.chorem.org/projects/pollen/files
	Descarga	si
	Versión	1.5.3
	Fecha última Actualización	**2013**
Tecnología	**Funcional**	java
	Lenguaje	java
	Traducción	Eng Fra
	Móviles	no
	Redes Sociales	si
Organización	**Estructura**	Empresa
	Comunidad	si
	Tamaño	pequeña
	Activa	si
	Soporte a Usuario	si

Desarrollado por la empresa Code Lutin es un sistema orientado a realización de consultas con múltiples preguntas y respuestas. Es utilizado de forma interna dentro de la empresa desarrolladora. Dispone de diferentes métodos de votación.

Existe una instancia que permite realizar consultas públicas en:
http://pollen.chorem.org/pollen/home

Interfaz de Administración: Permite la creación consultas con diferentes preguntas y diferentes respuestas por pregunta. Una vez creada la votación se notifica vía correo electrónico.
Interfaz de usuario Votación: La notificación de la votación contiene un enlace a la votación.
Interfaz de usuario Resultados: Pueden seguirse durante le proceso de votaciones.
Redes Sociales: No dispone de ninguna funcionalidad relacionada con redes sociales.
Instalación: Se realizan pruebas sobre una instancia de desarrollo, para la realización de pruebas más completas es necesaria su instalación sobre Apache Tomcat o Jetty.

Agora Ciudadana

General	Nombre	Agora Ciudadana
	Web	https://www.agoravoting.com/
	Caso Real	si
	Cumple Espc	si
Código	Licencia	LGPL
	Web descarga	https://github.com/agoraciudadana/agora-ciudadana
	Descarga	si
	Versión	
	Fecha última Actualización	2013
Tecnología	Funcional	sl
	Lenguaje	python
	Traducción	si Es.ES
	Móviles	si
	Redes Sociales	si
Organización	Estructura	Empresa
	Comunidad	si
	Tamaño	pequeña
	Activa	si
	Soporte a Usuario	si

Desarrollado por Wadabo (http://wadobo.com/) es un sistema orientado a la creación de grupos de discusión mediante la realización de consultas. Esta siendo evaluado por alguna plataforma ciudadana de forma experimental como herramienta de organización y discusión. Existe una instancia que permite realizar consultas públicas en: https://agoravoting.com /

Interfaz de Administración: Permite la creación de grupos (Ágoras) y dentro de cada uno de los espacios es posible crear consultas.

Interfaz de usuario Votación: Las diferentes votaciones existentes en el sistema son accesibles, bien sea a través del acceso con identificación o libremente dependiendo que la consulta sea pública o privada.

Interfaz de usuario Resultados: Pueden seguirse durante todo el proceso de votaciones.

Redes Sociales: Dispone de diversas funcionalidades que permiten publicar y seguir los resultados.

Instalación: Se realizan pruebas sobre una instancia de desarrollo, para la realización de pruebas más completas es necesaria su instalación sobre Apache.

9.2.4 Conclusiones de la Búsqueda

La primera valoración es que resulta bastante complicado recopilar información sobre experiencias reales relacionadas con el voto electrónico e implementadas con sistemas de votación con licencia Free/Open Sofware.

Pese a las características de observación y comprobación que ofrecen este tipo de desarrollos en la mayoría de los casos el código fuente de las aplicaciones no es público. En otros casos como en las experiencias realizadas en Noruega, es posible acceder al código pero no modificarlo ni ejecutarlo sin restricciones ya que está distribuido bajo una licencia privativa por lo que no puede considerarse de ninguna manera como código abierto.

Por otra parte es posible localizar un gran número de iniciativas relacionadas con el voto electrónico dentro del campo de las aplicaciones de código abierto, y aunque una gran mayoría no continúan su desarrollo, otras presentan funcionalidades muy completas. La parte más importante de aplicaciones corresponden a trabajos relacionados con la formalización de protocolos de encriptación y esquemas de votación.

Resulta también necesario aclarar que de ninguna manera puede garantizarse que no existan otras aplicaciones que, en lugar de estar alojadas en las forjas mas conocidas, mantengan su código en repositorios correspondientes a Facultades u otro tipo de Organizaciones.

Entre las aplicaciones localizadas hay cuatro que presentan alguna de las características consideradas necesarias:

- Civis Condorcet: Es una gran implementación del método Condorcet.
- Helios: De una gran corrección formal y académica.
- Pollen: Con unas funcionalidades muy completas.
- Ágora: Como en el caso anterior con grandes funcionalidades.

Todas ellas disponen de soporte y de comunidades activas.