



TESIS DOCTORAL

Una Arquitectura de Apoyo a la Participación Electrónica

Autor: César Alfaro Gimeno

Directores: David Ríos Insua y Javier Martínez Moguerza

*Departamento de Estadística e Investigación Operativa
Universidad Rey Juan Carlos*

Abril de 2013



Don **David Ríos Insua**, Catedrático del Departamento de Estadística e Investigación Operativa de la Universidad Rey Juan Carlos y Don **Javier Martínez Moguerza** Profesor Titular del Departamento de Estadística e Investigación Operativa de la Universidad Rey Juan Carlos.

AUTORIZAN:

La presentación de la Tesis Doctoral titulada

UNA ARQUITECTURA DE APOYO A LA PARTICIPACIÓN ELECTRÓNICA

Realizada por Don César Alfaro Gimeno bajo la dirección y supervisión de Don David Ríos Insua y Don Javier Martínez Moguerza y, que el Departamento de Estadística e Investigación Operativa ha dado su conformidad para que sea presentada ante la Comisión de Doctorado.

En Fuenlabrada, a 5 de Abril de 2013.

Fdo.: D. David Ríos Insua

Fdo.: D. Javier Martínez Moguerza



Don **Antonio Alonso Ayuso**, Profesor Titular de Universidad y Director del Departamento de Estadística e Investigación Operativa de la Universidad Rey Juan Carlos.

INFORMA:

Que la presente Tesis Doctoral titulada

UNA ARQUITECTURA DE APOYO A LA PARTICIPACIÓN ELECTRÓNICA

Ha sido realizada por Don César Alfaro Gimeno bajo la dirección y supervisión de Don David Ríos Insua y Don Javier Martínez Moguerza y, que el Departamento de Estadística e Investigación Operativa ha dado su conformidad para que sea presentada ante la Comisión de Doctorado.

En Fuenlabrada, a 5 de Abril de 2013.

Fdo.: D. Antonio Alonso Ayuso

Dedicado a la memoria de mi madre,
a la que echo de menos cada día.

Agradecimientos

En primer lugar quiero agradecer a mi padre por ser el pilar fundamental de todo lo que soy, tanto académica como personalmente. Ha sido un ejemplo de que con lucha y superación todo puede salir adelante. Y a mi hermano del cual aprendí de aciertos y de momentos difíciles.

A Bevi, por la comprensión y apoyo en aquellos momentos en los que mis estudios y trabajo ocuparon la mayor parte de mi tiempo.

Quiero agradecer a mi director, David Ríos Insua, la oportunidad de poder realizar esta Tesis, siendo el principal responsable de que la finalización de este trabajo haya sido posible. Su apoyo continuo, sus ideas, correcciones y conocimientos recibidos han resultado muy enriquecedores.

Debo hacer mención especial también, a Javier Martínez Moguerza, cotutor de la Tesis, por sus brillantes aportaciones al contenido del trabajo y los buenos consejos ofrecidos.

A Javi Gómez y José María Lavín por la ayuda en la redacción de la Tesis y los buenos momentos vividos durante estos años. Muchas gracias.

A Javi Cano Cancela, Ángel, Andrés, JuanJo, Felipe, Javi Cano Montero, Pablo, Alberto y Javi Guerrero, por su ayuda incondicional para resolver cualquier problema que me impedía avanzar en el trabajo.

Extendiendo el agradecimiento a todo el Departamento de Estadística e Investigación Operativa que me acogió cálidamente desde el primer momento y, en especial, a su director Antonio Alonso Ayuso, por toda la ayuda ofrecida.

Esta tesis se ha realizado con el apoyo de los proyectos e-Colabora, Agora.net, Riesgos – CM, ESF – COST Algorithmic Decision Theory, MyUniversity, Democracy4All, Corporate Community, Content Intelligent y EDUCALAB.

Índice general

Abstract	XV
Resumen	XVII
1. Principios de participación	1
1.1. Introducción	1
1.2. Diferentes tipos de democracia	2
1.3. Mecanismos de participación	7
1.4. Democracia electrónica	12
1.5. Experiencias de participación electrónica	15
1.6. Conclusiones	21
2. Una aproximación genérica a la participación electrónica	23
2.1. Introducción	23
2.2. Tareas participativas	24

2.3. Algoritmo genérico	29
2.4. Arquitectura genérica	32
2.5. Implementación	40
2.6. Conclusiones	42
3. ePML: Un lenguaje XML para la e-Participación	45
3.1. Introducción	45
3.2. Comienzos de XML	46
3.3. Lenguajes de esquema	49
3.4. Esquemas XML para una arquitectura de participación electrónica	51
3.5. Conclusiones	80
4. ePBPM: Un lenguaje gráfico de definición de procesos participativos	81
4.1. Introducción	81
4.2. Estándares y herramientas BPM	84
4.3. ePBPMN: Diseño de procesos participativos	93
4.4. Conexión con XML	101
4.5. Conclusiones	103

5. Fundamentos para un módulo de explicación basado en textos en una arquitectura de e-participación	105
5.1. Introducción	105
5.2. Revisión	106
5.3. Análisis de los mensajes	109
5.4. Metodología	113
5.5. Presentación y resultados del experimento	118
5.6. Conclusiones	127
6. Conclusiones y cuestiones abiertas	129
6.1. Conclusiones	129
6.2. Mejoras futuras	132
Bibliografía	137

Abstract

In recent years, there has been a raise in the desire to participate in governmental decisions by citizens. This has motivated the emergence of hundreds of participatory instruments that facilitate the transmission of citizen concerns and desires. In this thesis, we present a methodology that allows to model the participatory problem through Information and Communications Technology (ICT) and Decision Analysis methods.

Similarly, there has been an increased use of information and communication technologies allowing citizens to contact and provide their preferences to the political class. We develop an architecture to support distributed decision making, allowing to construct and implement most participatory processes available. It is based on a standard that enables information exchange among various phases of the process, using XML and Web services.

We also develop a BPM based methodology for designing participatory processes in a fashion simple, only filling a simple form and editing the process through a graphic interface. It comprises a methodology that suggests an optimal process for a given context.

Finally, we illustrate how to combine supervised machine learning algorithms and unsupervised learning techniques for sentiment analysis and opinion mining purposes. We describe a multi-stage method for the automatic detection of different opinion trends. We use these to support explanation systems that help to explain citizens the conclusion of the adopted decisions. This may contribute to improve user perceptions about trust, satisfaction and increase levels of acceptance of the proposed architecture.

Resumen

Durante los últimos años, se viene promoviendo la necesidad de realizar procesos de participación ciudadana en los que se permita identificar e incorporar las preocupaciones, necesidades y valores de los distintos actores involucrados en la toma de decisiones. Así, han surgido movimientos ciudadanos como el 15-M o Democracia Real Ya! que solicitan un cambio en el modelo político actual, en el que los ciudadanos desempeñen un papel más importante y se tengan en cuenta sus demandas.

El objetivo de esta tesis es describir una metodología que, con el uso de las Tecnologías de la Información y la Comunicación (TIC) y los métodos del Análisis de Decisiones, permita abordar el problema de la participación dotando a los ciudadanos de una herramienta para tomar decisiones sobre los temas que más les afectan.

En primer lugar, analizamos diversos tipos de democracia, la participación ciudadana, cuáles son sus objetivos y veremos en qué puntos puede ayudar el uso de las nuevas tecnologías para mejorar la participación. Veremos algunos mecanismos físicos de participación utilizados en todo el mundo y cómo cada vez va teniendo más acogida la participación remota frente a la presencial.

Seguidamente, pasamos a estudiar algunos mecanismos de participación, observando que todos ellos se pueden descomponer en tareas participativas que, combinadas entre sí, dan lugar a la mayoría de mecanismos que existen actualmente, pudiendo además crearse nuevos instrumentos. De este modo, se propone una arquitectura que permite implementar

cualquier instrumento participativo, dividida en módulos cada uno de los cuales se encarga de dar solución a una o varias tareas participativas.

Debido a esta separación en módulos de la arquitectura, a continuación, utilizaremos XML como mecanismo de intercambio de información entre los anteriores módulos debido a la gran cantidad de ventajas que aporta. Presentaremos varias formas de crear las reglas sintácticas por las que se rige un documento XML, los motivos que nos llevan a utilizar esquemas XML como lenguaje de esquema y las partes más características del esquema XML diseñado para participación.

Una vez presentada la arquitectura y el mecanismo de intercambio de información, parece evidente que el paso siguiente debe ser el diseño de un proceso participativo. Por ello, pensamos que sería necesario que éste fuera lo más sencillo posible, permitiendo a cualquier usuario construir un proceso. Basándonos en los sistemas de gestión de procesos de negocio y, especialmente, en el estándar de notación BPM (BPMN), proponemos un método que permita diseñar procesos participativos mediante la comunicación de una serie parámetros permitiendo, posteriormente, su modificación a través de una interfaz gráfica.

Con todo esto, consideramos que uno de los puntos más importantes de la participación es la explicación de las decisiones tomadas a los ciudadanos independientemente de su nivel cultural o socio-económico. Basándonos en los sistemas de explicación y diversas técnicas de minería de textos, presentamos una metodología multifase que nos permita extraer opiniones y generar informes explicativos de forma comprensible para cualquier individuo.

Finalizamos con las conclusiones obtenidas y problemas abiertos como futuras posibles mejoras que se podrían implementar.

Capítulo 1

Principios de participación

1.1. Introducción

La participación ciudadana puede aportar importantes beneficios a la sociedad. Para ello, debe adoptar una serie de principios que generen valor, y no desconfianza, entre los ciudadanos. Estos principios no indican una forma única de participación ciudadana: el mecanismo a utilizar dependerá de cada situación concreta y de las necesidades de ese momento. Para que el proceso sea efectivo, debe informar correctamente de manera que todos los ciudadanos conozcan objetivamente y con datos exactos cada uno de los proyectos y alternativas propuestas. Así, éste debe ser transparente, de modo que la información se encuentre disponible en todo momento y exista constancia del por qué se adoptaron o rechazaron los proyectos. Los ciudadanos pueden expresar opiniones, observaciones o sugerencias, pero las autoridades son las que finalmente toman las decisiones. El proceso de participación debe planificarse y documentarse correctamente con el fin de tener una referencia concreta para exigir el cumplimiento de los compromisos acordados. Es importante, además, conseguir un alto índice de participación en los procesos, por lo que se deben realizar esfuerzos para crear igualdad de oportunidades en los grupos más vulnerables de la sociedad. La responsabilidad en el cumplimiento de todas las acciones acordadas debe ser compartida entre las autoridades y los ciudadanos. La participación continua en los procesos de evaluación,

diseño y ejecución de proyectos permite que surjan nuevas ideas y se consiga una mejora considerable de los resultados.

Antes de comenzar exponiendo cualquier teoría, es necesario que nos preguntemos qué es la democracia, posiblemente una de las palabras que se haya utilizado con mayor ambigüedad a lo largo de la historia. Como dice Sartori (2003), definir la democracia es importante porque establece qué es lo que esperamos de ella. La definición que nos da de democracia es sencilla: poder del pueblo. Aunque, probablemente la definición más citada de democracia puede hallarse en el discurso de Lincoln en Gettysburg (1863), cuando la definió como “el gobierno del pueblo, por el pueblo y para el pueblo”, estando la clave de esta frase en “para el pueblo”.

De este modo, únicamente se describe un nombre, pero definir democracia es una cuestión que pasa a tener una dificultad considerable. Conocer el significado literal de la palabra no nos sirve de ayuda para entender cómo están construidas y funcionan las democracias posibles, porque entre el nombre y el objeto democracia hay una gran distancia. Así, vemos que el significado literal del término se corresponde poco y mal con su referente. Si bien la definición de democracia nos dice que el poder reside en el pueblo, en la sociedad actual esto dista de ser cierto. Los ciudadanos tienen la oportunidad de elegir al conjunto de personas que representarán sus intereses durante un tiempo determinado. Sin embargo, a partir de ese momento, la participación en la toma de decisiones públicas de cualquier ciudadano se ve limitada.

1.2. Diferentes tipos de democracia

La democracia ha estado siempre en evolución continua, de hecho, algunos de los aspectos fundamentales de las democracias contemporáneas, que ahora parecen esenciales, aparecieron hace unas pocas décadas. Con el modelo democrático actual, se ha conseguido que la política se aleje cada vez más de los ciudadanos reflejado en un descenso en los índices de participación. Por lo tanto, no se puede afirmar que las diferentes formas de

democracia no pueden continuar evolucionando, principalmente porque los conocimientos tecnológicos han prosperado satisfactoriamente, produciendo profundos cambios sociales y culturales, motivando en la ciudadanía movimientos para solicitar más presencia en la toma de decisiones. Estas circunstancias facilitan que la participación ciudadana sea posible.

La participación ciudadana se relaciona principalmente con la democracia deliberativa. De esta manera, intenta integrar al ciudadano en el proceso de toma de decisiones, para que pueda aportar ideas y proyectos sin tener que formar parte del gobierno ni de ningún partido político. En contraposición, están las concepciones tradicionales que conciben al gobierno como un mero ente prestador de servicios, como una simple Administración. Las principales variables para analizar el tipo de relación entre el Estado y la sociedad son la legitimidad y la eficacia para constituir una estabilidad política. De este modo, se puede entender la participación ciudadana como un proceso que configura su deliberación entre el Estado y la ciudadanía. La autoridad estatal se forma en el espacio de la sociedad civil. Es por eso que se requiere la construcción de una intermediación entre el Estado y la sociedad: la democracia deliberativa. El término democracia deliberativa fue acuñado por Bessette (1980). La democracia deliberativa se remonta a la época ateniense, donde los ciudadanos deliberaban en la asamblea antes de tomar cualquier decisión y consideraban las ventajas y desventajas de las distintas alternativas. En contraposición con la forma tradicional de votar, los teóricos de la democracia deliberativa argumentan que las decisiones sólo pueden ser legítimas si llegan como consecuencia de una deliberación pública por parte de la ciudadanía. Así, políticos, técnicos y opinión pública deliberan entre sí en un circuito democrático. En la actualidad, en algunos parlamentos, especialmente en Escandinavia, los sondeos deliberativos son una herramienta de los legisladores para consultar a la ciudadanía. La democracia deliberativa, también llamada en ocasiones sustancial, es un proceso de discusión pública en el que los participantes buscan un bien colectivo y permite reconocer diferencias y similitudes entre los miembros de una comunidad.

Para Elster (1998) la definición de democracia deliberativa se divide en dos partes: toma de decisiones con la participación de los afectados y por medio de argumentos propuestos por los participantes. Para Cohen (1998), el objetivo de la deliberación es tomar decisio-

nes. Insiste en que *”este proceso debe hacerse por individuos inscritos en el marco de la libertad”*. A estas definiciones, podemos añadir la de Habermas (1991), que recupera como norma de legitimación democrática la libre discusión. Los intereses particulares se vuelven generales y la voluntad se convierte en racionalidad comunicativa dirigida a la consecución del entendimiento entre dos o más sujetos. Según Habermas (1999, 2005), podemos distinguir dos vías para la implementación de la política deliberativa: *”la que implica la institucionalización de los procedimientos y presupuestos comunicativos y otra que permite la interacción con opiniones públicas construidas de manera informal”*.

Aunque la democracia deliberativa se relaciona con la participación ciudadana, podemos encontrar otros tipos de democracia entre los que destacan la democracia representativa, directa y participativa. De hecho, la democracia deliberativa incorpora elementos de las democracias directa y participativa.

La democracia directa pretende que los ciudadanos decidan directamente sin ningún tipo de representación. Las iniciativas propuestas se votan en referéndum y cada ciudadano emite su voto a favor o en contra de dicha iniciativa. Sin embargo, una de las objeciones más importantes que se suelen imputar a la democracia directa se refiere a su practicidad y eficiencia. El hecho de decidir sobre todos los temas de importancia pública mediante referéndum requiere del uso de abundantes recursos y logística que pueden provocar en el ciudadano apatía y fatiga, considerándose la democracia directa poco efectiva a la hora de aplicarse a gran escala.

La democracia representativa es el tipo de democracia más conocido. En ella, el pueblo elige al gobierno, que actúa en representación de los intereses de los ciudadanos, de forma periódica mediante elecciones. Su objetivo es eliminar la concentración de poder en una mayoría y lograr un equilibrio entre ejercer y controlar el poder, véase Arenilla (2010). Los problemas logísticos que aparecen en la democracia directa han sido uno de los principales motivos que han hecho que la democracia representativa sea casi universal.

En la democracia participativa, los ciudadanos tienen mayor oportunidad de tomar las

decisiones que les parezcan más oportunas. El Estado comparte con los ciudadanos el poder establecer criterios para las políticas públicas, además de permitir que los ciudadanos se agrupen de modo que puedan ejercer influencia en la toma de algunas decisiones públicas. Pero para que esto funcione, es necesario que el poder público sea transparente e informe de todas sus acciones. Esto no es nada sencillo, por lo que hay que cuestionar de qué manera el pueblo está capacitado para gobernar. Hay varias razones por las que la democracia participativa es importante:

- Aprovecha las experiencias y la capacidad de todos. Se beneficia del conocimiento que tienen los ciudadanos sobre los temas que afectan a nivel local.
- Promueve la legitimidad. Gobierno e instituciones ganarán mayor apoyo y confianza por parte de los ciudadanos si éstos pueden formar parte de las decisiones que les afectan y se sienten parte activa del proceso.
- Desarrolla nuevas capacidades. Con la participación se consigue que los ciudadanos colaboren entre sí y que los proyectos se lleven a cabo. Ofrece al ciudadano la capacidad de participar en la toma de decisiones y aportar nuevas ideas y proyectos.

Sin embargo, según Sartori, sea cual sea el nombre de la democracia, el problema reside en qué medida se traspa el poder del pueblo hacia el gobierno, cuando afirma que *“una cosa es la titularidad y otra cosa totalmente distinta es el ejercicio del poder. El pueblo soberano es titular del poder. Pero, ¿de qué manera el mismo pueblo se torna también capacitado para ejercer el poder?”*

En muchas democracias occidentales, la desilusión con el gobierno central y las instituciones, ahora llamado déficit democrático (Steffek et al., 2008), ha conducido a la consulta y la participación en las decisiones públicas. Esto se debe a que los representantes elegidos periódicamente toman las decisiones con escasa participación por parte de los ciudadanos, lo que ha conducido a que los instrumentos de participación prosperen en todo el mundo. El ciudadano no se ve representado por las instituciones y, aunque no siente afecto por la forma en que funciona la democracia representativa, considera el régimen democrático como

el más adecuado para vivir en sociedad. Este desafecto por parte de los ciudadanos hacia las instituciones políticas se debe a que los gobiernos son reticentes a rendir cuentas y atender las demandas de los ciudadanos. Así, los ciudadanos ven a los gobiernos y los partidos políticos como agencias particulares que gestionan los intereses de una serie de grupos.

La motivación de muchos políticos por promover este movimiento hacia la participación probablemente se relacione más con el hecho de obtener la aceptación de los ciudadanos que por ideales democráticos. Esto podemos verlo reflejado especialmente en el hecho de que los políticos están intentando mejorar su relación con el ciudadano mediante cartas de los ciudadanos, códigos éticos o códigos de buen gobierno. Aun así, independientemente de la causa, es cierto que están creciendo los métodos de participación en las decisiones públicas, desde los jurados populares a los sondeos deliberativos, pasando por los presupuestos participativos (Rowe y Frewer, 2005). Aunque algunos de ellos únicamente consisten en informar a los políticos de las preferencias de los ciudadanos, otros permiten a éstos tomar sus propias decisiones.

La participación ciudadana presenta una serie de beneficios potenciales como acercar las decisiones a los ciudadanos, eliminar la apatía y la alienación o aumentar la legitimación y transparencia de las decisiones adoptadas. Es frecuente encontrar que el objetivo que se desea alcanzar con la participación sea confuso, por lo que puede acabar dominando la idea de que la participación es buena y necesaria, sin buscar otros objetivos.

Así, en los últimos años, las experiencias participativas a nivel local han sufrido un gran auge con objetivos muy diversos. La participación y la deliberación parecen haberse convertido en «imperativos» de las políticas públicas, en particular de aquellas de ámbito urbano (Blondiaux y Sintomer, 2002). Los procesos de toma de decisiones cada vez son más complejos y requieren la cooperación de diferentes instituciones estatales y partenariados público-privados, en los que, en ocasiones, también participan representantes de la sociedad civil.

1.3. Mecanismos de participación

La creación de mecanismos de participación puede entenderse como un intento por ganarse la confianza de los ciudadanos, evitar el sentimiento de desilusión que actualmente éstos tienen con los políticos e intentar disminuir los índices de abstención. Se han dado distintos conceptos para describir esta evolución: gestión de proximidad, nueva administración pública, modernización de la gestión local o democracia participativa. Se pueden encontrar una gran cantidad de sistemas físicos de participación ciudadana utilizados en todo el mundo y que dan una idea de lo vibrante que es este campo, véase, entre otros, Lavin et al. (2013); Rowe y Frewer (2005). Entre ellos mencionamos:

- **Agenda 21 local.** Surge como técnica para garantizar el desarrollo sostenible en las ciudades, como principales núcleos de desarrollo para el siglo XXI. La Agenda 21 local incluye a: técnicos, agentes sociales, ciudadanos, y el gobierno local, El procedimiento de elaboración se inicia con técnicas deliberativas guiadas por los técnicos. A continuación, se diseñan las líneas principales de desarrollo y, finalmente, se crean una serie de indicadores de consenso para seguir el cumplimiento de los compromisos. El área de aplicación de la Agenda 21 local es amplia, aunque las principales áreas de acción son las que se refieren a la planificación urbanística, la actividad económica, desarrollo industrial y a la movilidad urbana.

En la actualidad, gran cantidad de municipios en todo el mundo han desarrollado esta técnica, aunque sólo unos pocos logran cumplir sus acuerdos y objetivos.

- **Consultas ciudadanas.** A través de una serie de preguntas concretas y bien definidas, se trata de conocer la opinión de los ciudadanos acerca de futuras acciones públicas. Este mecanismo se utiliza cuando existen varias soluciones, pero ninguna es preferible. Su principal objetivo es conocer la opinión de los ciudadanos respecto a una serie de iniciativas. Para poder llevarlas a cabo, es necesario un amplio despliegue informativo, para que todos los ciudadanos conozcan en detalle cada una de las acciones que se pueden aplicar y, así, puedan elegir con criterio la que más les interesa. Esto

supone un elevado coste, por lo que no es recomendable utilizarlas asiduamente. Con todo ello, el rasgo más característico de este mecanismo es que permite la participación de personas que, típicamente, no podrían participar en otros mecanismos, por ejemplo, menores de dieciocho años o inmigrantes.

- **Foros de barrio.** Son reuniones abiertas y públicas donde se invita a todos los vecinos que lo deseen, sin ningún tipo de requisito, ni obligación. Tienen un carácter consultivo y participativo intentando fomentar el diálogo de los ciudadanos. Los temas a tratar son flexibles, no siguiéndose un guión preestablecido, donde cada vecino puede exponer sus problemas. De este modo, se tiene información de primera mano sobre los asuntos concretos que debe tratar el ayuntamiento. No hay que olvidar que se trata de un órgano consultivo, donde las decisiones sobre la gestión de los barrios sigue perteneciendo a los técnicos municipales, pudiendo aceptar o rechazar las propuestas de los ciudadanos. Con las nuevas tecnologías se han creado foros de barrio “online” que complementan o sustituyen a los presenciales, donde el principal problema son los aspectos logísticos que limita la participación de los vecinos.
- **Jurados ciudadanos.** Buscan soluciones concretas a un problema que genera conflicto. Son mecanismos consultivos. El número de personas que compone un jurado suele ser reducido, normalmente entre diez y doce personas. Durante las sesiones, expertos y técnicos exponen la información de la que disponen acerca del asunto a tratar. Una vez que el jurado ha recibido dicha información deliberará sobre el problema, ofreciendo una solución o varias alternativas. Este mecanismo es ventajoso en zonas donde no existe mucha confianza en los sistemas tradicionales de toma de decisiones. Las experiencias con este mecanismo de participación comenzaron en EEUU y Alemania, extendiéndose posteriormente a otros muchos países.
- **Peticiones ciudadanas.** Se pueden encontrar tres tipos de peticiones ciudadanas, dependiendo de su motivación: para tratar algún aspecto novedoso del cual no se ha encargado la administración, para aprobar algo que la administración haya rechazado o para impedir alguna medida tomada por la administración. Para que se acepte una petición ciudadana, ésta deberá presentarse por escrito, con las firmas de los ciudadanos

que la presentan, un texto de votación, una propuesta de financiación y los nombres de los representantes legales de la petición. La recogida de firmas podrá realizarse en cualquier lugar que los organizadores determinen, y deberá ser de, al menos, una parte de la población con derecho a voto, tomando como referencia el censo de las últimas elecciones. El país más activo en el uso de este mecanismo es Alemania y, más concretamente, el Estado de Baviera.

- **Planes de barrio.** A través de la cooperación de todos los vecinos se inician proyectos ambiciosos que mejoran la calidad de vida. Se desarrollan en un área territorial reducida que cuenta con la ventaja de tener comunicación e interacción directa con los vecinos. Permite adaptar el plan a las necesidades de estos y, que ofrezcan y desarrollen iniciativas propias. Pero, por otra parte, al ser las áreas de actuación más reducidas, la financiación de los planes no la costea únicamente el Ayuntamiento, sino que es necesaria la aportación de empresarios de la zona o asociaciones de vecinos. Este instrumento permite al Ayuntamiento, además de tomar las medidas oportunas para aumentar la satisfacción de la población a nivel de barrio, conocer las preferencias de cada uno de los barrios y realizar acciones a nivel de municipio.
- **Presupuestos participativos.** La ciudadanía toma parte, junto con el gobierno municipal, en la decisión de cómo se gastan algunas partidas presupuestarias del ayuntamiento. El objetivo principal es la participación directa de la ciudadanía para establecer sus principales demandas, priorizando las de mayor interés y realizando un seguimiento de los objetivos alcanzados, véase Alfaro et al. (2010). Los presupuestos participativos permiten a los ciudadanos tomar sus propias decisiones, además de que hay una mayor transparencia en la gestión municipal, mejora la comunicación entre la administración y los ciudadanos. Los presupuestos participativos intentan implicar al ciudadano en la marcha de la ciudad, muestran a la administración de manera más cercana y se inicia una democracia más participativa en la que se posibilita a los ciudadanos exponer sus opiniones, necesidades y propuestas, así como establecer sus prioridades.
- **Referéndums.** Se cede la responsabilidad de la decisión a los ciudadanos que serán

los encargados de decidir, mediante una votación y por mayoría, qué propuestas se llevarán a cabo y cuales no. Mediante el referéndum se intenta conocer cuál es la opinión de los ciudadanos sobre asuntos concretos lo que requiere una amplia campaña de información para obtener respuestas cercanas a la realidad. La aplicación de los referéndum debe limitarse, puesto que un uso masivo de éstos podría provocar cansancio en la población y, como consecuencia, aumentar el índice de abstención, lo que daría lugar a que el proceso se viera alterado por la falta de legitimidad. Por lo tanto, es recomendable utilizar otro tipo de instrumentos como las encuestas o los sondeos, para conocer la opinión de los ciudadanos de manera aproximada.

Del mismo modo que encontramos diferentes formas de participar, también se pueden diferenciar las experiencias dependiendo del tipo de participantes.

El primer modelo se apoya en la representación de la sociedad. Representa a los ciudadanos por sectores sociales, más allá del ámbito económico e intentan tener en cuenta la representación de grupos específicos. En esta perspectiva, las iniciativas de participación local permiten a las mujeres y hombres, jóvenes, niños, personas de tercera edad y discapacitados estar representados como tales. Un segundo tipo de mecanismo se refiere a aquellas asociaciones cuyas actividades se centran en llevar a cabo tareas relacionadas con sus ámbitos de competencias. En el tercer tipo de mecanismo, los ciudadanos son voluntarios y no es necesario que formen una organización. El riesgo de este mecanismo es que existan demasiados participantes que no representen verdaderamente al resto de la población. En el cuarto tipo de mecanismo los ciudadanos se eligen por sorteo, por lo que se evita que un grupo de personas monopolice el poder, ya que se trata de estructuras no permanentes. El último tipo de mecanismo está dirigido a asociar a la totalidad de los ciudadanos. Cuando la participación es suficiente, la legitimidad del referéndum es casi siempre intachable e, incluso, claramente superior a la de la democracia representativa clásica.

Hay tres factores que permiten distinguir los modelos participativos como son la voluntad política, el factor de tipo procesal y los conceptos sociopolíticos. Dados estos tres factores, podemos diferenciar hasta cinco modelos de participación ciudadana, véase Sinto-

mer (2007) para más detalles:

- **El modelo “managerial”**. En este modelo, las decisiones se toman por una cooperación pública/privada, donde los ciudadanos juegan un papel secundario. Los objetivos sociales son prácticamente inexistentes y las intervenciones participativas están despolitizadas, por lo que el poder local es débil.
- **Modelo de “modernización participativa”**. El Estado tiene una fuerte presencia y desempeña un papel importante. Los ciudadanos son consumidores de servicios y tienen poco poder de decisión. Las reglas de participación son claras: transparencia, calidad de los servicios y acceso garantizado a la información.
- **Modelo de “democracia de proximidad”**. En este modelo, la palabra proximidad tiene doble sentido. Geográficamente, la participación es eficaz sobre todo a escala micro local, puesto que se gestiona mejor gracias al conocimiento que tienen los ciudadanos de su propio barrio. Políticamente, la proximidad desarrolla el diálogo entre políticos y ciudadanos. Las reglas de participación son poco claras, sobre todo en lo que se refiere a la forma en que debe establecerse la síntesis de los debates.
- **Modelo de empowerment**. Se caracteriza por una disociación fuerte entre las estructuras participativas del Estado y la política institucional. La sociedad dispone de una fuerte autonomía y los mecanismos de participación tienen gran capacidad decisoria. Los responsables políticos juegan un papel secundario. Los ciudadanos participan en la decisión y, muy a menudo, también en la gestión y la realización de los proyectos.
- **Modelo de democracia participativa**. Las instituciones de la democracia representativa se articulan sobre las estructuras de la democracia directa o semidirecta. La eficacia gestora de la democracia participativa depende mucho de la modernización del Estado.

La implantación que se está haciendo actualmente de la participación ciudadana no representa un riesgo para las instituciones y los gobiernos, ya que el poder que pierden es aceptable por la imagen de mayor sensibilidad que reflejan.

1.4. Democracia electrónica

El reciente interés por Internet y las redes informáticas ha reavivado el debate sobre la democracia. Por un lado, algunos políticos hablan entusiasmados sobre la posibilidad de ejercer una política virtual en la que se recree el ideal griego de participación directa, lo que derivaría en el nacimiento de una nueva forma de democracia. En la otra cara de la moneda, están los que temen que se pueda originar su destrucción.

Las ventajas que aportan las nuevas tecnologías están creando las condiciones necesarias para la implementación de una democracia electrónica. El desarrollo de las redes de telecomunicaciones y de las nuevas tecnologías de la información, permitiría que la población pueda votar por Internet, informarse de la política gubernamental o interrogar a sus representantes. Aparecen así los ciudadanos activos e implicados, protagonistas del sueño de una democracia real.

La participación electrónica puede considerarse como “toda actividad voluntaria de los ciudadanos encaminada a influenciar en la selección de los gobernantes o en la toma de decisiones públicas que se canaliza o se produce a través de medios electrónicos” (Verba et al., 1995). Dada esta definición, podemos encontrar distintos niveles de participación, dependiendo del grado de implicación por parte del ciudadano, o del impacto en las elecciones o en otras decisiones políticas. Por tanto, para clasificar las distintas experiencias electrónicas de participación, resulta útil la idea de la escalera de participación (Arnstein, 1971) que incluye cinco niveles de participación que van de menor a mayor participación (Figura 1.1):

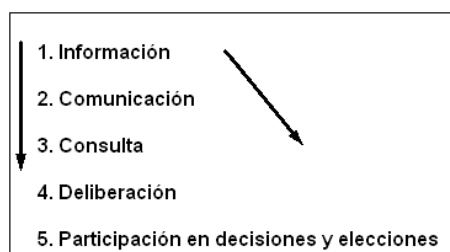


Figura 1.1: *La escalera de la participación*

El contenido y los elementos característicos más habituales de cada nivel de participación electrónica pueden verse en el Cuadro 1.1:

Niveles de participación	Características
Información	Divulgación de información a través de webs y correos electrónicos. Elementos informativos: documentos varios, enlaces web, comunicados, convocatorias, anuncios, avisos, noticias, etc . . .
Comunicación	Relación de contactos de manera bidireccional a través del correo electrónico y de espacios de comunicación habilitados en las webs. Elementos comunicativos: preguntas, sugerencias, demandas, quejas, comentarios, cartas, organización de convocatorias y reuniones, etc.
Consulta	Formas que emplean los gobiernos, administraciones, y organizaciones diversas para saber las opiniones de los ciudadanos o de sus miembros. Elementos consultivos: referendos, encuestas, sondeos, etc.
Deliberación	Procesos de examen, evaluación, reflexión, debate y discusión sobre las decisiones, opciones y valores que impregnan cualquier tema o problema sociopolítico. Elementos deliberativos: foros, chats, espacios de debate, etc.
Participación en decisiones y elecciones	Participación en elecciones: voto electrónico. Participación en una actividad orientada a la toma de decisiones; es decir, el resultado final es vinculante para las autoridades; referendos o encuestas vinculantes, recogida de firmas para iniciativas legislativas o para iniciar consultas ciudadanas en los ayuntamientos, etc.

Cuadro 1.1: *Niveles de participación y características*. Fuente: Borge (2005)

Sin embargo, el concepto de democracia electrónica no sólo es producto de la “fiebre” por las redes informáticas. Forma parte también de extensas investigaciones sobre la aplicación de las tecnologías de telecomunicaciones a la democracia, que se han ido desarrollando de un modo paralelo. Mientras que la tecnología que podría hacer posible la democracia electrónica es nueva, las cuestiones filosóficas que hace referencia son antiguas y de difícil solución. Este hecho se ha evidenciado especialmente por la acalorada discusión que se ha producido en los últimos años sobre la aplicación de las nuevas tecnologías a la política. Existe un consenso general que dice que, nos guste o no, estamos en el umbral de una nue-

va era de la democracia electrónica. Con el advenimiento de los ordenadores en los años 60, y la consiguiente proliferación de las nuevas tecnologías de la información, el concepto de democracia electrónica dejó de ser materia de especulaciones para convertirse en motivo de una considerable experimentación. Hoy, la cuestión no debería ser si es posible la democracia electrónica sino, más bien, quién la va a desarrollar y organizar, y con qué fines.

En estos últimos años, se ha producido una explosión de conocimiento en los campos de la informática y la telemática que están suponiendo un gran impulso para que la participación electrónica se convierta en realidad. Términos como e-gobierno, e-participación, e-democracia, etc. . . suenan cada vez con más fuerza, permitiendo un acercamiento entre los ciudadanos y los políticos, gracias también, en parte, a la aparición de las redes sociales que permiten un contacto fluido entre unos y otros. El ejemplo más claro es el de las elecciones a la presidencia de Estados Unidos en el año 2008, donde Barack Obama apoyó gran parte de su campaña electoral en las redes sociales, creando incluso su propia red social MyBO, cuestión reflejada aún más en las elecciones de 2012, véase (Rodríguez, 2012).

A pesar de los grandes avances de los últimos años en el área de la informática y el uso de Internet, un gran problema con el que se encuentra la participación electrónica es que el uso de las TIC no es universal, incluso en los países más desarrollados. Esta situación ha recibido el término “brecha digital”, aunque es posible disminuir tal brecha con políticas de accesibilidad web, para que todas las personas, independientemente de sus limitaciones físicas o derivadas del entorno, puedan utilizar Internet. Esto ha hecho que el uso de las TIC en la participación electrónica se reduzca a unas pocas experiencias en las que se emplean foros de discusión y votación electrónica a pesar de disponer de la tecnología necesaria para realizar otras tareas más complejas, como por ejemplo (MyUniversity, 2012). A pesar de ello, es bastante probable que en un futuro próximo estas barreras se vayan desvaneciendo y aumenten de forma considerable las ocasiones en las que se pueda participar a través de la Web. Además, el uso de las TIC podría potenciar de forma considerable las ventajas de la participación.

1.5. Experiencias de participación electrónica

Podemos dividir las experiencias de participación ciudadana en dos grandes bloques: los sistemas de voto electrónico y otras formas de participación que se basan en Internet. Estos dos bloques presentan amplias diferencias en cuanto a su grado de innovación en el diseño de mecanismos participativos. En el voto electrónico, únicamente se añadiría un nuevo canal para desarrollar la función de elección de los representantes, mientras que las nuevas formas de participación vía Internet son innovaciones participativas significativas, aunque no afecten a muchos ciudadanos.

1.5.1. Voto electrónico

En un principio, el voto electrónico fue empleado en el marco de las votaciones privadas, principalmente en el contexto de asambleas de accionistas o de asociaciones profesionales. Más tarde, tuvieron lugar algunas experiencias piloto en organismos públicos, que se han multiplicando recientemente. Son muchos, los países que, al menos, han comenzado el estudio de la posible implantación del voto electrónico como medio para aumentar la participación ciudadana en las elecciones y facilitar el sistema de recuento, además de para intentar reducir el fraude electoral y mitigar posibles problemas derivados del analfabetismo.

Hay dos tipos de sistemas de voto electrónico: los presenciales y los remotos. Los primeros se desarrollan de manera muy similar al voto que denominamos tradicional, porque los ciudadanos se desplazan a un colegio electoral para votar in situ. Por el contrario, el sistema remoto se basa en una red de comunicaciones que permite enviar el voto desde el lugar en el que está el votante hasta una urna digital remota.

El primer país donde se ha empleado el sistema de votación remoto para elecciones a gran escala ha sido Estonia, aunque otros países como Suiza, Estados Unidos, Irlanda y el Reino Unido se encuentran en fase de prueba de sistemas para observar los resultados que

se obtienen en términos de satisfacción ciudadana y potencial uso. En cambio, los sistemas presenciales están siendo empleados ya en países como Bélgica, Holanda, Filipinas, Brasil y la India. Debe incidirse que, incluso, en algunos países, entre ellos España, sería necesario modificar la legislación para poder incorporar este tipo de sistemas al régimen electoral en un futuro próximo.

Una de las experiencias más importantes de voto electrónico es la desarrollada en Estonia. En el año 2001, el Ministerio de Justicia hizo una propuesta para introducir la posibilidad de votación electrónica en las elecciones al Parlamento del año 2003. El plan proponía que los ciudadanos se registraran como e-votantes y firmaran sus papeletas electrónicamente con firma digital, pudiendo hacer esto, a través de Internet, desde sus domicilios. Para emplear el sistema, los ciudadanos sólo necesitaban un ordenador dotado de un lector electrónico de datos, encargado de leer el documento de identidad electrónico, una firma electrónica y una contraseña. Por último, una vez elegidos los candidatos, era necesario introducir una segunda contraseña, con la que el voto se encripta antes de enviarse al servidor, véase Madise y Maaten (2010). En concreto, en las elecciones locales de octubre de 2005, unos 10.000 electores pudieron votar por Internet en Bancos, Oficinas estatales y municipales y a través de Compañías de Telecomunicaciones. Desde entonces, el número de electores que han emitido su voto por Internet ha ido creciendo con el paso de los años. Así, en las elecciones parlamentarias de 2007 esa cifra ascendió hasta los 30.000, en las locales de octubre del año 2010 los votantes internautas fueron 104.000 y en las elecciones parlamentarias de Marzo de 2011, algo más de 140.000 personas emitieron su voto por Internet lo que representa más de un 15 % del electorado que estaba llamado a las urnas y algo más del 24 % de los votantes, véase VVK (2011). A medida que han pasado los años se ha incrementado el número de personas que han emitido su voto a través de la red, como podemos ver en la Figura 1.2:

	Elecciones locales 2005	Elecciones parlamento 2007	Elecciones parlamento europeo 2009	Elecciones locales 2009	Elecciones parlamento 2011
Votantes censados	1 059 292	897 243	909 628	1 094 317	913 346
Votantes participan	502 504	555 463	399 181	662 813	580 264
Votos electrónicos	9 317	30 275	58 669	104 413	140 846
Porcentaje de voto electrónico / votantes censados	0,9%	3,4%	6,5%	9,5%	15,4%
Porcentaje de voto electrónico / participantes	1,9%	5,5%	14,7%	15,8%	24,3%

Figura 1.2: *Estadísticas generales.*

En España, actualmente se dan las condiciones tecnológicas necesarias para implantar el voto electrónico, aunque, todavía no se dan las circunstancias culturales ni legales que lo hagan posible. Un alto porcentaje de la población se muestra todavía reticente al voto por Internet, por creerlo “poco fiable”. La única Comunidad Autónoma que posee regulación legislativa de voto electrónico, paso previo indispensable para su implantación, es el País Vasco. La primera prueba experimental de voto por Internet celebrada en España se realizó en 2003. La localidad en la que se hizo realidad la iniciativa fue El Hoyo de Pinares (Ávila), donde los 1.900 ciudadanos censados en el Ayuntamiento fueron los primeros españoles en participar en un proyecto de estas características.

Una experiencia piloto a destacar, realizada en 2004, fue Madrid Participa. Esta prueba piloto de consulta ciudadana, no vinculante jurídicamente, tenía como fin consultar a

los ciudadanos sobre varias cuestiones relacionadas con posibles actuaciones y mejoras en equipamiento a promover por el Ayuntamiento en sus barrios con respuestas cerradas, sólo pudiendo elegir una respuesta por pregunta. Esta consulta, con un número de potenciales votantes de más de 136.000, podía tener lugar mediante acceso remoto con móviles y ordenadores, o mediante acceso presencial con el uso de Tablet PCs u ordenadores personales. El índice de participación fué bajo, siendo de un 0,65 %. De esta experiencia pudieron sacarse datos importantes como por ejemplo, la valoración por parte de los participantes de las tecnologías utilizadas (Figura 1.3), o el sistema de votación preferido (Figura 1.4).

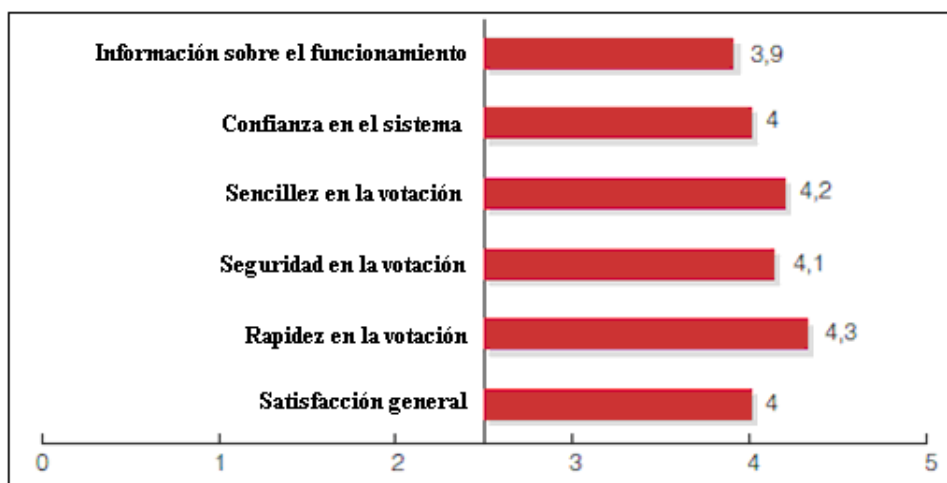


Figura 1.3: Valoración de la tecnología de voto electrónico

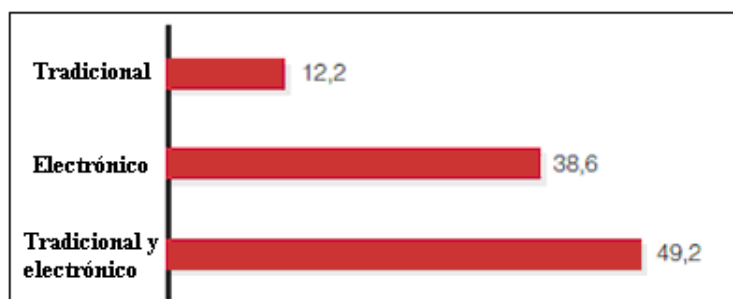


Figura 1.4: Valoración del sistema de votación preferido

Por lo que respecta a comunidades autónomas, la experiencia más reciente fue la de las elecciones autonómicas de 2011 donde cuatro provincias, Castellón, Huesca, Mérida y Ceuta, contaron con mesas electrónicas para emitir los votos. En el resto de España, el voto electrónico también se ha experimentado en diversas instituciones. Así, por ejemplo, en el año 2009, la Universidad Rovira i Virgili de Tarragona realizó las elecciones a la Junta de la Escuela Técnica Superior de Ingeniería mediante votación electrónica. Fueron las primeras elecciones celebradas en una universidad de España a través de Internet y con efecto vinculante. En el año 2010, la Universidad de Alcalá utilizó por primera vez voto electrónico en las elecciones de Claustro a través de pantalla táctil situando los terminales electrónicos en diversas facultades. En ese mismo año, el partido político Unión Progreso y Democracia (UPyD) eligió sus órganos de dirección utilizando el voto electrónico en Madrid, con diversas mesas auxiliares instaladas por los municipios de la comunidad.

En la Guardia Civil se han celebrado ya tres procesos electorales (2002, 2004, 2009) para seleccionar a los miembros del Consejo Asesor de Personal, que es el instrumento de participación de los guardias civiles en determinadas decisiones del cuerpo. En las elecciones realizadas en el año 2009, la participación fue del 34,16 %, siendo un resultado satisfactorio teniendo en cuenta que en los dos procesos anteriores, lo había hecho menos del 1 % del censo.

Otra experiencia importante es la de MyUniversity, que se trata de una plataforma para permitir la participación en los procesos de toma de decisiones de diversas universidades europeas. En MyUniversity se trata de incrementar la participación de la comunidad universitaria ofreciendo herramientas electrónicas que faciliten la toma de decisiones e involucre a todos en la política de la universidad. Para ello, hace uso de una plataforma avanzada de participación electrónica que ofrece diferentes herramientas interactivas basadas en la tecnología Web 2.0.

1.5.2. Presupuestos participativos

En el año 2005, la Junta de El Rabal (Zaragoza) inicia el proceso de “Presupuestos Participativos Vía Internet”. Se trata de un programa piloto del Ayuntamiento y del Grupo de Decisión Multicriterio de Zaragoza (GDMZ), de la Universidad de Zaragoza, que permitió conocer, a través de la red, la opinión de los vecinos sobre la distribución de los presupuestos del barrio. El proyecto consistió en distribuir la partida presupuestaria entre cuatro alternativas planteadas (supresión de barreras arquitectónicas, zonas deportivas, zonas verdes, limpieza viaria) a través del análisis de las preferencias de los participantes de este proyecto entre unos criterios económicos, sociales y ambientales. Los resultados obtenidos fueron reveladores en el sentido de que se observó que, a los ciudadanos, les preocupaban más los problemas sociales que los económicos. La experiencia se repitió en el año 2006 aumentando la participación respecto al año precedente.

El proceso de presupuestos participativos realizado en Sector III, barrio de la ciudad de (Getafe, 2008), es otro claro ejemplo de que la participación a través de Internet, se está convirtiendo en algo fundamental para mejorar la participación ciudadana. Este proceso, incrementó la participación respecto al año anterior, realizado únicamente de forma presencial, obteniendo un 42 % de participación de modo telemático.

En la ciudad de Bogota (2011), se realizó un proceso de presupuestos participativos en el cuál, se permitía por primera vez, el voto electrónico a través de Internet. En dicho proceso se decidió en que gastar el presupuesto disponible de inversión. Los ciudadanos, además de tener la posibilidad de emitir sus opiniones a través del método tradicional (voto en urna), pudieron hacerlo a través de Internet accediendo con sus credenciales a la web creada a tal efecto. Los resultados obtenidos demostraron que el tema que más interés suscitó entre los votantes fue la seguridad y que algo más del 2 % de los ciudadano votó a través de Internet, cifra más que interesante teniendo en cuenta que es la primera vez que se permitía dicho voto.

1.6. Conclusiones

El concepto de participación se remonta a la Grecia antigua, pero es a partir del siglo XX cuando empieza a tomar un papel relevante en la sociedad, aportando importantes beneficios. Sin embargo, para ello, debe adoptar una serie de principios que generen valor y no desconfianza entre los ciudadanos.

La participación ciudadana está representada en las sociedades a través de la democracia, relacionándose principalmente con la democracia participativa, aunque existen otros tipos de democracia como la representativa, directa y deliberativa.

En los últimos años se ha desarrollado en la sociedad una creciente conciencia, tanto individual como colectiva, sobre la importancia de influir en las decisiones. De ahí, que hayan aparecido diferentes mecanismos de participación presencial, como pueden ser los jurados ciudadanos o los presupuestos participativos, con el fin de ganarse la confianza de los ciudadanos y evitar el sentimiento de desilusión actual.

El auge de las nuevas tecnologías y las redes sociales están permitiendo que la participación electrónica sea una realidad. Esta, facilita el acercamiento entre ciudadanos y políticos. Pero, a pesar de los beneficios que aporta la participación electrónica, ésta tiene un obstáculo que debe solventar y es que el uso de Internet no es universal. Tal situación ha recibido el nombre de “brecha digital” que ha derivado en que el uso de las TIC en la participación se reduzca a unas pocas experiencias en las que se emplean tareas relativamente sencillas. Las experiencias más destacadas han sido las de voto electrónico, de las cuales existen dos tipos, los sistemas presenciales y los remotos, aunque, en los últimos años, es frecuente que se utilicen otras formas participar a través de la red.

Este interés creciente, por parte de la ciudadanía, en formar parte de las decisiones que más les afectan, nos impulsó a proponer el desarrollo de una metodología que permita construir procesos participativos. Para ello, implementamos una arquitectura que facilite la ejecución de procesos y consiga aumentar la transparencia y aceptación de las decisiones.

Capítulo 2

Una aproximación genérica a la participación electrónica

2.1. Introducción

El desarrollo de la administración pública contemporánea se está caracterizando por el incremento de los procesos de reforma local, acompañados, en muchas ocasiones, de la implementación de mecanismos de participación ciudadana (Dollery et al, 2008). Esta dinámica está relacionada con la idea cada vez más extendida de que la participación ciudadana entre elecciones es un factor que impacta positivamente en las democracias. Muchos países europeos han transformado recientemente sus legislaciones para incentivar la apertura de los gobiernos locales a través de formas de consulta y democracia directa. Así el año 2002, Francia introdujo su Ley de la Democracia de Proximidad y en España, en el 2003, se promulgó la Ley de Medidas para la Modernización del Gobierno Local que implementan reformas que permiten aumentar la transparencia del poder de los ejecutivos locales. Estas leyes tienen como objetivo establecer una mayor rendición de cuentas e incrementar el diálogo entre los gobernantes y la ciudadanía regulado mediante instrumentos de participación.

Por otro lado, la expansión en los últimos años de las TIC puede permitir que el uso de

dichos instrumentos participativos aumente involucrando a los ciudadanos en la democracia. Las TIC han aumentado la capacidad de comunicación y de distribución de la información, lo que permite que los ciudadanos tengan mayor conocimiento de los problemas y que las decisiones sean de mayor calidad. Además, podrían suponer un canal de comunicación entre ciudadanos y políticos para aumentar la transparencia y la participación, directa y colectivamente, de los ciudadanos en cuestiones políticas.

Estas razones, nos impulsaron al desarrollo de una arquitectura genérica que facilite permita con el objetivo de facilitar el diseño y ejecución de los procesos participativos ya existentes y nuevos que pudieran surgir. Para ello, permite la combinación de las tareas comunes identificadas en muchas de las experiencias participativas analizadas, cumpliendo una serie de restricciones.

En la sección 2, veremos las etapas en las que se dividen los instrumentos participativos y las tareas comunes que hemos identificado en los mismos. En la Sección 3, describimos un algoritmo genérico de apoyo a procesos participativos. A continuación, mostraremos una arquitectura genérica que da soporte a cualquier instrumento, facilitando su uso a los ciudadanos y consiguiendo transparencia. En la sección 5, explicaremos detalles de la implementación de dicha arquitectura. Por último, presentaremos algunas conclusiones.

2.2. Tareas participativas

Aunque todavía con limitaciones, en los últimos años ha habido un uso creciente de los mecanismos de participación ciudadana en la administración pública. Tratando de atender las peticiones de los ciudadanos, existen diversos tipos de experiencias, con funcionamiento y resultados diferentes, que pretenden fomentar la participación de los ciudadanos en las decisiones de ámbito público.

Una vez que hemos identificado y descrito las fases comunes a cada uno de los instrumentos de participación ciudadana más extendidos, podemos ver que, básicamente, todos

los instrumentos de participación se dividen en tres etapas, como puede verse en la Figura 2.1:

- Fase inicial: En esta fase se identifican el problema a tratar, los conflictos asociados, las necesidades y las expectativas de la población.
- Fase de elaboración: Se procede a la recogida de ideas, propuestas y objetivos, y la toma de decisiones de grupo.
- Fase final: Se procede a la presentación de los documentos con los resultados del proceso participativo.

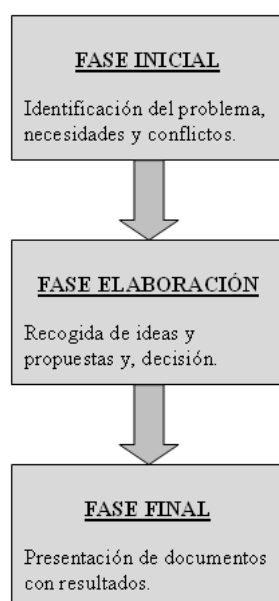


Figura 2.1: *Momentos comunes de un proceso participativo.*

Existe una gran cantidad de instrumentos participativos, superior a los cien, véase por ejemplo Rowe y Frewer (2005). A través de las experiencias analizadas en todo el mundo, se han identificado varias tareas comunes que aparecen en los distintos procesos participativos, de

modo que, podemos considerar un instrumento participativo como una serie completa de actividades e interacciones entre autoridades, expertos y ciudadanos, desde la exploración inicial hasta la elaboración de informes finales. Las tareas identificadas son:

1. **Muestreo de participantes:** En muchos mecanismos, la participación de toda la ciudadanía es compleja, bien por razones físicas o logísticas. En estas ocasiones, se elige a un grupo de personas, que puede ser elegido al azar, pueden ser voluntarios, . . . , dependiendo del problema a tratar, que representen al resto de la población.
2. **Elección de representantes y/o expertos:** En ocasiones, se eligen representantes o expertos en debates públicos o votaciones por el resto de participantes. Estos se encargan de expresar las opiniones colectivas, permitiendo agilizar la tarea y mejorando la eficiencia del proceso. Los expertos se encargan de aportar conocimiento técnico sobre las propuestas.
3. **Uso de cuestionarios:** Ayudan a conocer los principales intereses y preocupaciones de la población y, de este modo, trabajar en la medida de lo posible realizando propuestas o buscando soluciones que satisfagan dichos problemas.
4. **Preparación de documentos:** Antes del comienzo de cualquier proceso de participación, es necesaria la generación de documentos que informen a los participantes sobre el proceso que se va a iniciar. Estos documentos contienen información referente al problema a tratar, así como la descripción del proceso. Del mismo modo, se generan documentos una vez que ha finalizado el proceso participativo que contienen información relevante de cada una de las fases por las que ha pasado el proceso, así como los resultados obtenidos, un resumen de la fase de debate, en caso de que exista dicha fase, y las decisiones finalmente adoptadas. Ambos tipos de documentos suelen ser escritos por los representantes o expertos.
5. **Distribución de información:** Es uno de los elementos más importantes de un proceso de participación. Cualquier tipo de dato teórico, práctico o de experiencias pasadas

es fundamental y debe estar disponible para facilitar a los participantes el proceso de toma de decisiones.

6. **Intercambio de información:** Del mismo modo que es importante que la información se distribuya, los participantes deberían intercambiar con el resto la información de que dispongan para que, de este modo, aumente el conocimiento sobre el problema y sea posible tomar la mejor decisión.
7. **Estructuración del problema:** En ocasiones, los problemas no se formulan fácilmente y es necesario dividirlos en partes para que sean comprensibles por los ciudadanos. Así, en la estructuración de problemas, los expertos se encargan de determinar los criterios de evaluación, elaborar una lista de posibles soluciones al problema propuesto, describir características técnicas de cada solución e identificar restricciones o dependencias entre soluciones.
8. **Generación de alternativas:** Los participantes se reúnen para proponer alternativas que solucionen los problemas planteados. Para ello, utilizan técnicas como la tormenta de ideas (brainstorming) que favorece la participación y la creatividad de los ciudadanos. Estas reuniones están coordinadas, típicamente, por uno o varios técnicos para aportar experiencias pasadas.
9. **Comunicación de preferencias:** Los participantes expresan sus preferencias sobre las alternativas que pueden dar solución al problema planteado. Las preferencias pueden comunicarse a través de comparaciones entre pares o mediante funciones de valor o de utilidad. Dichas preferencias servirán para conocer de primera mano la opinión de los participantes y, de este modo, ofrecer soluciones que satisfagan a los mismos.
10. **Exploración individual del problema:** En ocasiones, cada participante se toma un tiempo para estudiar el problema en profundidad y encontrar su alternativa preferida.
11. **Debates:** Algo importante para la participación ciudadana es el intercambio de ideas, ya que, sin éstas, las decisiones tomadas son pobres y carentes de imaginación. Los debates se proponen para que los participantes discutan sobre las alternativas ofrecidas, así como sobre cada uno de los proyectos. Del mismo modo, los participantes

pueden proponer nuevos proyectos o alternativas, que serán supervisadas por técnicos especializados que, en caso de considerar que aportan algo positivo, las añadirán a la lista final de propuestas.

12. **Negociación:** En caso de que los participantes no lleguen a un acuerdo sobre las alternativas preferidas, la negociación puede ser una buena técnica para intentar resolver el conflicto y que todas las partes interesadas obtengan beneficios. Para esta fase, podrían utilizarse métodos como POSTING o el método de incrementos equilibrados (BIM), entre muchos otros.
13. **Arbitraje:** Puede presentarse la situación de que no se alcance ningún acuerdo en las fases de debate o negociación, y que las partes no deseen moverse de su posición. En estas ocasiones, puede utilizarse la figura del árbitro que se encarga de tomar la decisión final una vez que conoce las opiniones y preferencias de los participantes. El árbitro debe ser lo más justo posible para no perjudicar a ningún individuo o colectivo.
14. **Votación:** La votación es, en muchas ocasiones, el último paso en el proceso de toma de decisiones y se utiliza como último recurso en caso de que no haya sido posible llegar a un acuerdo en las fases anteriores. Se pueden usar diferentes reglas como son la de mayoría simple, el voto por aprobación, el conteo de Borda, entre muchas otras, véase Nurmi (2010).
15. **Explicación a los ciudadanos:** Los documentos con los resultados obtenidos deben estar disponibles para todos los participantes. De este modo, los participantes pueden comprobar que el proceso ha transcurrido correctamente y no existe ningún tipo de incongruencia. También puede ser necesaria una explicación de las decisiones tomadas por parte de las instituciones, o por medio de los representantes a través de discursos públicos.

Se pueden distinguir, además, tres perfiles de usuario distintos:

- La **autoridad**, que es la encargada de proponer el problema, estructurarlo (describir proyectos y características técnicas), y publicarlo.

- Los **ciudadanos**. Son los actores principales del proceso y aportan ideas en el proceso de toma de decisiones, dando opiniones, sugerencias y comunicando sus preferencias sobre el problema a resolver.
- Los **técnicos o expertos**, que son los encargados de controlar que el proceso se desarrolle correctamente. También realizan labores de mantenimiento en la estructuración del problema y resuelven las dudas y sugerencias recibidas por parte de los ciudadanos.

2.3. Algoritmo genérico

Los instrumentos participativos surgen con el objetivo de lograr un acercamiento entre la clase política y la ciudadanía. Del mismo modo, tratan de permitir que los ciudadanos participen en las decisiones públicas que les afectan, consiguiendo aumentar los niveles de participación y una mayor satisfacción de las acciones realizadas por parte de éstos. Sin embargo, a pesar de aportar potencialmente numerosos beneficios, cada instrumento presenta sus ventajas e inconvenientes, como pueden ser la cantidad de recursos que requieren, bien sean de tipo espacial, temporal o financieros.

Por ello, hemos observado que, se podría diseñar un proceso genérico de participación intercambiando las distintas tareas para crear nuevos procesos que combinen las ventajas de cada instrumento participativo, en función de las necesidades de cada ocasión. Dichas tareas podrían ejecutarse en el orden en que el diseñador del proceso considere oportuno, pudiéndose repetir varias de ellas a lo largo del proceso. Sin embargo, deberían cumplir una serie de restricciones como, por ejemplo, que algunas tareas no pueden aparecer a continuación de otras o no pueden comenzar o finalizar el proceso participativo. Así, por ejemplo, la tarea de generación de documentos no puede aparecer en primer lugar, debido a que no existen datos para generar los documentos.

Tomando como referencia las tres fases principales de los instrumentos participati-

vos, podemos construir un algoritmo genérico separando las tareas según en qué etapa del proceso pueden aparecer. Así, en la fase inicial, podrían utilizarse las tareas de muestreo, elección de representantes, preparación de documentos, estructuración de problemas, generación de alternativas o debate. En la fase de elaboración, las tareas disponibles serían uso de cuestionarios, distribución e intercambio de información, comunicación de preferencias, exploración individual del problema, debate, negociación, arbitraje y votación. Por último, el proceso podría finalizar con las tareas de explicación o de preparación de documentos. Dentro de cada uno de las fases del proceso participativo podrían existir restricciones entre las tareas porque realizan la misma actividad o porque son incompatibles. Por ejemplo, en la fase inicial no podrían aparecer en un mismo proceso la tarea de muestreo y la de elección de representantes porque, ambas, tienen la misma finalidad que es la de minimizar el número de participantes para agilizar el proceso. Al mismo tiempo, existen tareas que pueden aparecer en distintas fases porque sean útiles en distintas etapas del proceso. Es necesario destacar que todas las tareas podrían repetirse sin restricciones, pero teniendo en consideración que, la repetición de una misma tarea o tareas que se alarguen en el tiempo, pueden disminuir los índices de participación.

Un posible ejemplo de proceso participativo creado a partir del algoritmo genérico podría ser el de la Figura 2.2. En primer lugar, hay una fase de estructuración del problema en la que los técnicos incorporan los datos del problema a resolver. A continuación, los ciudadanos debaten sobre dichos datos añadiendo o eliminando los que piensan que tienen menos relevancia. Seguidamente, los técnicos modifican el problema, teniendo en cuenta los acuerdos alcanzados por los ciudadanos en la fase de debate. En la fase de comunicación de preferencias, los ciudadanos expresan, mediante diferentes técnicas, los aspectos que mayor o menor importancia tienen, para cada uno de ellos, para resolver el problema. En caso de alcanzar un acuerdo, se implementa la solución con mayor aceptación. En caso contrario, el proceso avanza a la fase de negociación. En dicha fase, los ciudadanos presentan propuestas al resto que pueden ser aceptadas o rechazadas por los mismos. Al igual que en la fase de comunicación de preferencias, si una propuesta alcanza un nivel de aceptación mayor o igual que un umbral, determinado al comienzo del proceso, se implementa dicha propuesta;

en caso contrario, los ciudadanos votan para intentar alcanzar un acuerdo. Una vez finalizada la votación, el sistema comprueba si la propuesta más votada es socialmente óptima. En caso afirmativo, se implementa dicha solución; si no, el proceso volvería a la fase de negociación, repitiéndose el ciclo “negociación-votación” hasta que los ciudadanos lleguen a un acuerdo definitivo. Para finalizar el proceso, los técnicos preparan los documentos finales que explican a los ciudadanos las decisiones adoptadas. Cabría la posibilidad de, una vez que no se ha conseguido un nivel de aceptación que supere el umbral, reducir dicho umbral, para que no sea necesario repetir en múltiples ocasiones el bucle “negociación-votación”, que podría ocasionar cansancio en los ciudadanos y, que éstos dejen de participar.

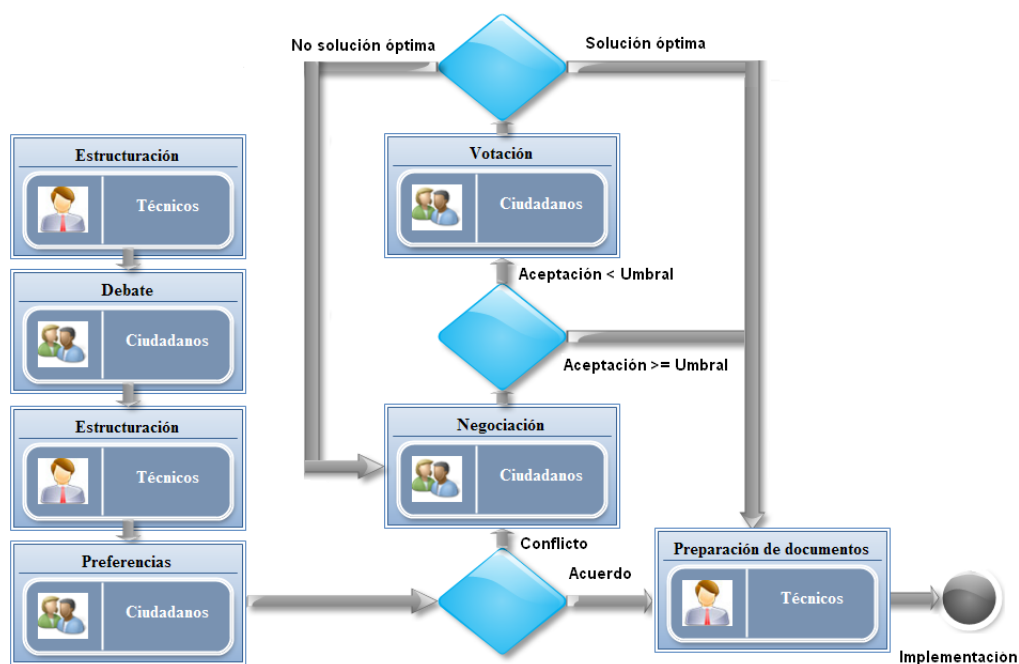
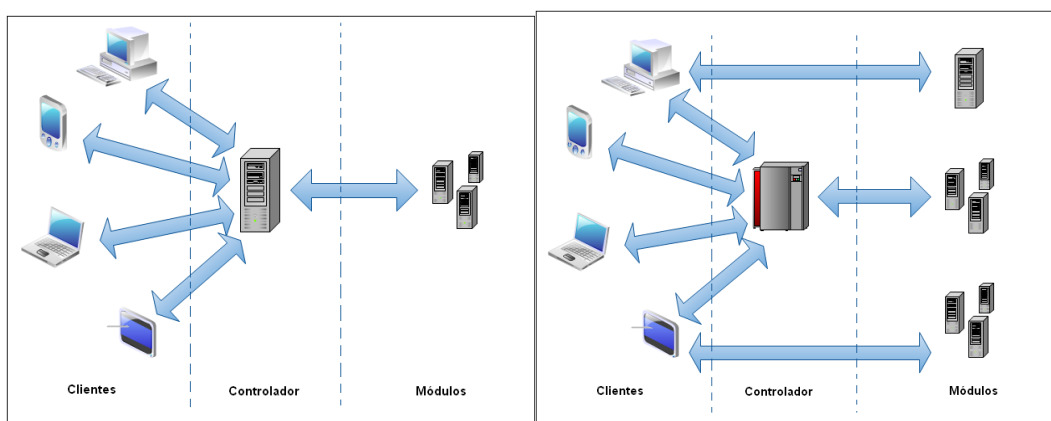


Figura 2.2: Ejemplo de un proceso participativo genérico.

2.4. Arquitectura genérica

A continuación, se describe una arquitectura general que puede dar soporte a una herramienta para fomentar el uso de la participación electrónica y, al mismo tiempo, facilitar su uso a los ciudadanos y conseguir transparencia durante el proceso. La arquitectura está pensada para, combinando cada uno de los módulos, poder construir cualquiera de los instrumentos de participación descritos en los puntos anteriores, así como, poder crear nuevos instrumentos dependiendo de las necesidades del proceso que se vaya a realizar. El esquema sigue la arquitectura SOA (Arquitectura Orientada a Servicios), donde se definen servicios, principalmente servicios web, para dar soporte a los requisitos de negocio.

Una primera implementación podría haber sido centralizar el acceso a la aplicación a través del controlador ofreciendo mayor seguridad al sistema pero, con la desventaja de que si éste fuera inaccesible, el sistema entero dejaría de funcionar. Por lo tanto, nos decantamos por una arquitectura distribuida donde el peso recaiga sobre los proveedores de servicios y, si alguno deja de estar disponible, otro que realice la misma tarea pueda reemplazarlo. En la Figura 2.3 podemos ver un ejemplo de las posibles soluciones de implementación.



(a) Arquitectura centralizada a través del Controlador.

(b) Arquitectura distribuida.

Figura 2.3: Posibles soluciones

Implementar un sistema orientado a servicios, utilizando servicios web, implica el uso

de una serie de estándares y protocolos, para que los servicios se encuentren accesibles en la Red, tales como SOAP, WSDL y UDDI que, básicamente, se utilizan para definir, localizar, implementar y hacer que un servicio web interactúe con otro. SOAP (2011) es un protocolo que permite el intercambio de mensajes basados en XML entre dos procesos, independientemente del protocolo de transporte utilizado, el tipo de comunicación establecido o las reglas de interpretación del mensaje. El lenguaje WSDL, basado en XML, se utiliza para describir lo que hace un servicio web, dónde se encuentra y la forma de ser invocado, véase WSLD (2011). Por último, UDDI (2011) también basado en XML, se utiliza para registrar y describir servicios y negocios con diferente tipo de información sobre los servicios web de un sistema. UDDI se compone de dos partes:

- Descripción de negocios que se almacenan utilizando tres categorías: páginas blancas (información de contacto), páginas amarillas (información de la industria) y páginas verdes (información técnica).
- Registro de servicios.

Una vez descritos los estándares y protocolos asociados a los servicios web, en la Figura 2.4 podemos ver los pasos a seguir para registrar un servicio y cómo un posible cliente haría uso del mismo.

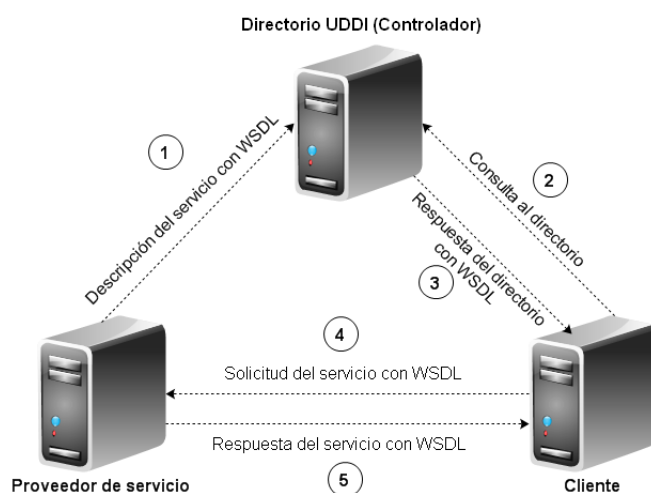


Figura 2.4: *Modelo básico del funcionamiento de un servicio web.*

En primer lugar, el agente proveedor del servicio (negociación, votación, etc . . .) envía un mensaje SOAP con la descripción del servicio que ofrece al directorio de servicios (UDDI) utilizando para ello WSDL. Una vez que el servicio ha sido registrado, está disponible para que cualquier cliente pueda hacer uso del mismo. Para ello, el cliente solicita al directorio el servicio que necesita y éste responde con un mensaje SOAP la dirección donde se encuentra el servicio solicitado y los parámetros requeridos para un correcto funcionamiento del mismo. A continuación, el cliente envía una petición al proveedor con los parámetros indicados para ejecutar la tarea solicitada. Por último, el proveedor del servicio responde al cliente con el resultado correspondiente a la ejecución.

Con todo esto, el esquema general de la arquitectura tendría un aspecto como el de la Figura 2.5:

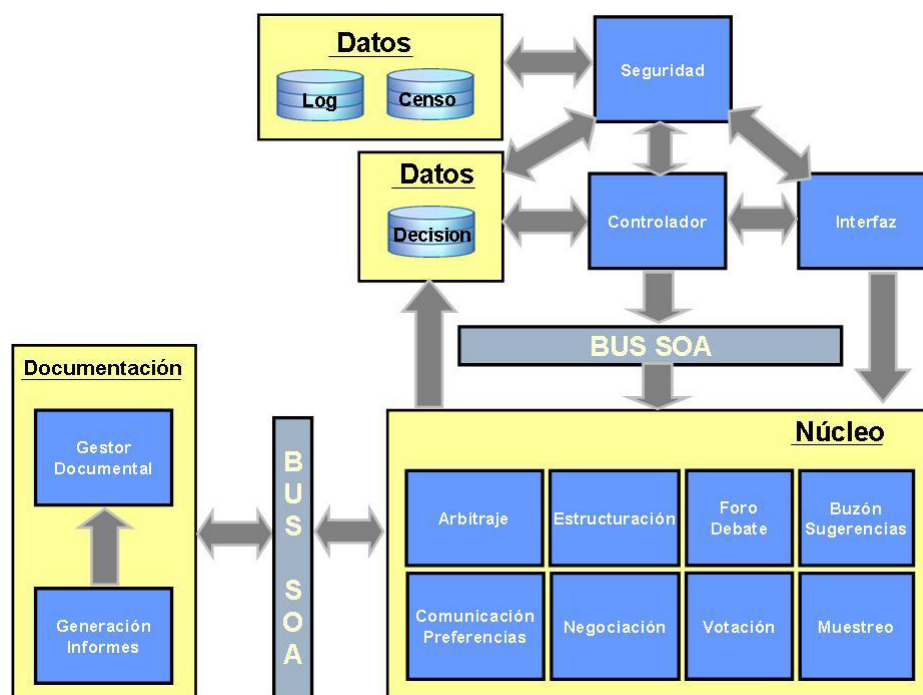


Figura 2.5: Arquitectura general de los distintos módulos.

La arquitectura incluiría tres bases de datos principales: una para almacenar el censo de la población; otra para almacenar la información y datos generados durante el proceso por parte de los participantes, que permitirán tomar la mejor decisión; y, otra, para almacenar los movimientos que realizan en el sistema los participantes y, al mismo tiempo, sirve para detectar si se ha producido alguna entrada no deseada. El acceso a las bases de datos debería estar restringido a los diferentes módulos. Las bases de datos quedarían de la siguiente forma:

- **Censo.** En la base de datos Censo se encuentran almacenados todos los datos de los participantes, sus roles dentro de la aplicación, los contenidos a los que tiene acceso, etc... Únicamente se permite acceder a ella a través del módulo de seguridad.
- **Log.** En esta base de datos se almacenan todos los movimientos realizados en el

sistema, lo que sirve para que cualquier usuario con permisos pueda verificar que todo está correcto, permitiendo de este modo aumentar la confianza en el sistema.

- **Decisión.** Almacena los datos de cada proceso, el estado en el que se encuentra, las preferencias de los usuarios, mensajes enviados, proyectos y soluciones adoptadas en caso de que el proceso haya finalizado.

En lo referente a módulos, podemos encontrar cinco principales que, unidos de forma adecuada, permitirían construir cualquier proceso de toma de decisiones.

- **Controlador:** Es uno de las partes más importantes del sistema. Se encarga de controlar que los procesos pasen por todas las fases, es decir, de coordinar al resto de subsistemas. Para ello, se encarga de intercambiar información entre los módulos del Núcleo y de indicar en qué momento debe ejecutarse cada fase. Utiliza el protocolo UDDI para registrar los servicios web ofrecidos por los componentes del sistema. Además, cuando se realiza cualquier acción en algún módulo del núcleo, graba dicha acción en la base de datos Log.
- **Interfaz.** Encargada de interactuar con los usuarios. Con el fin de mitigar la brecha digital, debería ser gráfica y de fácil uso. Se podría considerar el uso de distintos tipos de interfaces según la fase del proceso o el tipo de instrumento participativo. Debería cumplir con los estándares de accesibilidad Web Accessibility Initiative (WAI), para que el contenido llegue al mayor número posible de personas.
- **Módulo de Seguridad.** Debería impedir accesos no deseados a la aplicación. Para tal fin, utilizando la información de las bases de datos Censo y Decisión, comprueba si un usuario tiene permisos suficientes y le asigna su rol dependiendo de los contenidos a los que tenga acceso.
- **Módulo de Documentación.** Se encuentra dividido en dos submódulos. Generación de informes, que se encarga de generar documentos con datos obtenidos a través del

módulo del Núcleo que haya solicitado la generación del informe; y gestor documental, que almacena los informes generados por el módulo anterior. En caso de que cualquier módulo del Núcleo lo requiera, encontraría el documento en el gestor.

- **Núcleo.** Es uno de los componentes más importantes de la arquitectura y donde se aloja la lógica de negocio. Se encuentra dividido en varios módulos, que se irán ejecutando en el orden que establezca el “Controlador”. Cuando el controlador determine que un módulo debe ejecutarse, utilizará la base de datos Decisión para obtener la información necesaria para su correcto funcionamiento, a medida que los usuarios realicen las peticiones. Los principales módulos en que se encuentra dividido el núcleo en la arquitectura son:

1. **Control del Proceso Participativo.** Da soporte a la creación, especificación y estructuración del proceso, definiendo las fases a incluir, las herramientas permitidas en cada fase, cuánto tiempo permanecemos en cada fase, . . . Una vez que se especifica, acuerda y secuencia el proceso, este módulo se encargaría del control de la correcta ejecución del mismo. Dependiendo del proceso participativo, podría tener distintos estados como adquiriendo información, estructurando el problema, debatiendo, negociando, votando, . . .
2. **Estructuración de problemas.** Es el encargado de ayudar a estructurar el problema, es decir, de ayudar a identificar los proyectos que se pueden realizar, los criterios de evaluación de los proyectos o las restricciones que se pudieran dar entre los mismos. El encargado de estructurar el problema debería ser un técnico especializado o un administrador del sistema.
3. **Modelización de Preferencias.** Se utilizaría para que los participantes comuniquen de forma sencilla e intuitiva sus preferencias sobre los criterios que evalúan cada proyecto. De este modo, el sistema conoce de primera mano los intereses de cada participante y puede ofertar soluciones óptimas para la resolución de los posibles conflictos que se presenten.
4. **Gestor de Votaciones.** Este módulo permite resolver el conflicto mediante la votación de los participantes a los proyectos o representantes. Se apoya con

distintas reglas de votación.

5. **Gestor de negociaciones.** Se emplea para resolver conflictos a través de la negociación de los distintos actores que participan en el proceso de toma de decisiones. Se encarga de gestionar métodos de negociación, como puede ser POSTING, en el que los participantes envían propuestas, y el resto de participantes votan a favor o en contra de dicha propuesta, hasta métodos como BIM.
6. **Gestor de Arbitrajes.** Un árbitro, en este caso el sistema, se encarga de ofrecer la solución más justa y que más beneficios aporta a los participantes. Se pueden aplicar varios métodos de arbitraje.
7. **Gestor de debates.** Es el encargado de permitir a los participantes modificar la lista inicial de proyectos o criterios de evaluación, mediante su opinión en un foro de discusión. Si una gran mayoría de los participantes desean realizar alguna modificación, el administrador del problema la realizará en caso de considerarla viable y oportuna, consolidándose, de este modo, la lista final de propuestas. Esta es una cuestión que adquiere cada vez más importancia con el auge de las redes sociales en Internet y la Web 2.0.
8. **Buzón de sugerencias:** Permite a los participantes de los distintos procesos realizar consultas o sugerencias al administrador de un proceso sobre cuestiones relacionadas con el mismo, o posibles mejoras que se podrían realizar en el sistema. Estas sugerencias podrán tener carácter público quedando accesibles a todos los ciudadanos, o privado, siendo los destinatarios de dichas solicitudes los únicos que podrán verlas.

La idea de que el sistema haya sido dividido en módulos es porque aporta una mayor escalabilidad, pudiendo añadir nuevos módulos, aumentando de este modo la funcionalidad sin necesidad de realizar ninguna modificación importante. Del mismo modo, se pueden añadir nuevas soluciones a cualquier tarea registrando el servicio en el controlador. Un ejemplo de servicio Web implementado es el que se muestra en el Código 2.1:

```
1 <wsdl:definitions
2     targetNamespace="http://publics.services.problemStructuring.participation.es">
3 <wsdl:types>
4     <xsd:schema
5         targetNamespace="http://model.option.problemStructuring.participation.es">
6         <xsd:complexType name="Option">
7             <xsd:sequence>
8                 <xsd:element name="description" type="xsd:string"/>
9                 <xsd:element name="name" type="xsd:string"/>
10            </xsd:sequence>
11        </xsd:complexType>
12    </xsd:schema>
13 </wsdl:types>
14
15 <wsdl:message name="getOptionRequest">
16     <wsdl:part name="parameters" element="xsd:string" />
17 </wsdl:message>
18
19 <wsdl:message name="getOptionResponse">
20     <wsdl:part name="parameters" element="Option" />
21 </wsdl:message>
22
23 <wsdl:portType name="remotePSServicePortType">
24     <wsdl:operation name="getOption">
25         <wsdl:input name="getOptionRequest" message="tns:getOptionRequest" />
26         <wsdl:output name="getOptionResponse" message="tns:getOptionResponse" />
27     </wsdl:operation>
28 </wsdl:portType>
29
30 <wsdl:binding name="remotePSServiceHttpBinding" type="remotePSServicePortType">
31     <wsdl:operation name="getOption">
32         <wsdlsoap:operation soapAction="" />
33         <wsdl:input name="getOptionRequest">
34             <wsdlsoap:body use="literal"/>
35         </wsdl:input>
36         <wsdl:output name="getOptionResponse">
37             <wsdlsoap:body use="literal"/>
38         </wsdl:output>
39     </wsdl:operation>
40 </wsdl:binding>
41
42 <wsdl:service name="remotePSService">
```

```
43 <wsdl:port name="remotePSServiceHttpPort" binding="remotePSServiceHttpBinding">
44   <wsdlsoap:address
45     location="http://company.com/problemStructuring/services/remotePSService"/>
46 </wsdl:port>
47 </wsdl:service>
48 </wsdl:definitions>
```

Código 2.1: Ejemplo de servicio web para solicitar una opción por su identificador.

Un servicio web se divide en dos partes: parte abstracta y parte concreta. La parte abstracta define la interfaz del servicio, qué hace a través de los mensajes que envía y recibe. Engloba las etiquetas “type”, “messages” y “portType”. La parte concreta define la implementación del servicio y engloba las etiquetas “binding” y “service”. En el servicio web del ejemplo se han eliminado algunas partes que no eran imprescindibles para comprender su funcionamiento. Dicho servicio se encarga de devolver una opción dado su identificador. Para ello, se indica que debemos utilizar el método “getOption”, que recibe el identificador de la opción en una cadena de caracteres, retornando un objeto de tipo Opción con los datos del mismo.

2.5. Implementación

Para implementar la arquitectura descrita en la sección 2.4 hemos utilizado una plataforma de programación de software libre como es Java EE, que permite utilizar arquitecturas de N capas distribuidas y se apoya en componentes modulares que se ejecutan sobre un servidor de aplicaciones. Para su implementación, nos hemos basado en el patrón MVC (Modelo, Vista, Controlador) que permite separar la lógica de negocio (modelo) y la presentación (vista).

Al sistema sólo pueden acceder usuarios registrados mediante la identificación con nombre de usuario y contraseña. Una vez dentro, si el usuario tiene permisos podrá construir un nuevo proceso participativo. Para ello, deberá indicar las fases que forman dicho proceso,

así como el orden en que se ejecutarán indicando las fechas de inicio y de fin de cada una de ellas. Una vez que el usuario ha diseñado el proceso, el sistema lo identifica como dueño del proceso que ha creado pudiendo acceder a zonas del mismo que no serán visibles para el resto de usuarios.

Además, el sistema incluye el correspondiente soporte de información a través de un esquema de base de datos relacional y una interfaz de usuario basada en la web para gestionar la interacción de los usuarios con el mismo. Como se ha comentado, las páginas a las que puede acceder un usuario dependerán de si es un participante o el dueño del problema y del estado en que se encuentre el proceso al que accede en ese momento. Por ejemplo, si se está estructurando el proceso, los participantes no podrán comunicar sus preferencias para calcular su función de valor, porque el vínculo a dicho módulo estará bloqueado.

En la Figura 2.6 podemos ver la pantalla en la que, una vez que se han añadido las alternativas y los criterios al problema, el técnico encargado de estructurarlo debe introducir la relación que existe entre los mismos. Dicha relación, se representa en una tabla en la que las filas representan las alternativas y las columnas representan los criterios de evaluación del problema. Cada celda representa la relación que existe entre la alternativa y el criterio. En el ejemplo mostrado, la alternativa “Espectáculos” tiene una relación de 20 con el criterio “Gasto total”, mientras que la alternativa “Parque” tiene una relación de 400 para el mismo criterio.

Project	Efectividad	Abarcamiento	Ingresos-manten	Gasto total	Precio
Agua	1.0	1.0	-50.0	300.0	300.0
Árboles	1.0	1.0	-30.0	150.0	150.0
Biblioteca	1.0	0.5	-33.0	1200.0	700.0
Carril bici	1.0	0.2	0.0	80.0	80.0
Deporte	0.75	0.4	0.0	25.0	25.0
Escuela	1.0	0.6	-120.0	1200.0	700.0
Espectáculos	0.3	0.3	0.0	20.0	20.0
Estatua	0.3	0.3	-15.0	50.0	50.0
Estranjera	1.0	0.15	-50.0	100.0	100.0
Parque	0.75	0.55	-20.0	400.0	250.0
Pastorizar	0.3	0.3	200.0	10.0	10.0
Pipican	0.75	0.1	-10.0	10.0	10.0
Poli-deportivo	0.75	0.6	100.0	2100.0	1200.0
Seguridad	1.0	0.0	-150.0	120.0	110.0
Teatro	0.3	0.3	0.0	350.0	350.0

Figura 2.6: Relación entre las alternativas y los criterios.

2.6. Conclusiones

A pesar de existir más de un centenar de mecanismos de participación en todo el mundo, a través de su estudio, detectamos que la mayoría de ellos surgen de la combinación de tareas comunes que se ejecutan en distinto orden, dando lugar a un nuevo proceso participativo. Así, podemos crear nuevos instrumentos coordinando diferentes tareas y adaptar el proceso a las necesidades requeridas en cada momento. Para ello se ha creado un algoritmo genérico en el cual, las tareas se pueden repetir el número de veces que se desee y aparecer en cualquier orden, que siempre cumplen una serie de restricciones.

Como hemos visto, para utilizar un servicio ofrecido por la arquitectura, el cliente o usuario final debe consultar el catálogo de servicios UDDI para localizar la dirección donde se encuentra el servicio y cómo debe utilizarlo. Todo este intercambio de información debe realizarse mediante mensajes SOAP.

A partir de la creación del algoritmo genérico, surgió la idea de diseñar una arquitectura para conseguir que la creación de nuevos instrumentos de participación sea lo más sencilla y rápida posible. Se compone de varios módulos interconectados para ofrecer todas las herramientas necesarias para crear dichos instrumentos. Cada uno de los módulos puede tener una localización geográfica diferente permitiendo, de este modo, que se puedan añadir nuevas soluciones independientemente de su localización. Por ejemplo, dos tareas de un proceso participativo podrían estar localizadas una en Barcelona y otra en Estocolmo, pero formar parte del mismo proceso. El componente principal de la arquitectura es el “Núcleo”, en el que se encuentran los módulos que implementan cada una de las fases comunes identificadas. Desde un principio se pensó en que el sistema pueda ser modificado añadiendo nuevas funcionalidades. Por ello, la arquitectura ha sido dividida en partes, para lograr un bajo acoplamiento entre los distintos componentes y conseguir que el mantenimiento sea simple.

A continuación, veremos la sintaxis de los documentos XML que facilitarán el intercambio de información entre los módulos de la arquitectura y, que permitirán, manejar la

información de forma estructurada.

Capítulo 3

ePML: Un lenguaje XML para la e-Participación

3.1. Introducción

Existe en la actualidad un gran auge en el uso de herramientas de toma de decisiones, muchas de ellas basadas en la Web. Así, nos encontramos con que un gran número de los sistemas actuales de participación son plataformas genéricas disponibles a través de Internet. De este modo, sería interesante crear un estándar que pueda utilizarse en la *e-participación*, aprovechando el papel que pueden desempeñar las herramientas Web en el apoyo a la toma de decisiones y la participación ciudadana, así como las características de XML y su aplicación en diversos campos. Dicho estándar, permitiría diseñar un modelo con las opiniones o preferencias de cualquier participante y ayudaría a modelizar las características del problema y sus posibles formas de resolución.

A pesar del auge en la utilización de los instrumentos de participación para la toma de decisiones son pocas las iniciativas en *e-democracia* y *e-gobierno* que aprovechan los beneficios que proporciona XML. Una de ellas es el proyecto ITALO, véase Oficina Virtual (2011), iniciado por el gobierno español para hacer posible el intercambio de informa-

ción administrativa entre administraciones públicas, a través de ficheros electrónicos. Otro ejemplo sería el proyecto GovTalk, véase CabinetOffice (2011), iniciado por el gobierno británico. Aunque ambas propuestas son pioneras en su concepción, no tienen en cuenta aspectos esenciales para hablar de un sistema de participación electrónica. Esto se debe a que en ambos proyectos se demuestra la utilidad de XML dentro del ámbito del gobierno electrónico, pero en ningún momento se ofrece apoyo a la toma de decisiones por parte de los ciudadanos.

Introducimos en este capítulo EPML (e-Participation Markup Language), un lenguaje de marcado que permite y facilita el intercambio de información entre los distintos módulos de la arquitectura descrita en el Capítulo 2 de esta tesis, cuyo fin es apoyar la toma de decisiones por parte de los ciudadanos. Los procesos de *e*-participación pueden adoptar distintos caminos para la resolución de un problema concreto. Por consiguiente, nuestro esquema deberá, del mismo modo, almacenar toda la información referente al problema, así como las fases a seguir.

Comenzaremos hablando de los inicios de XML: cómo surgió, cómo se ha ido desarrollando, su uso y ventajas, así como aplicaciones en las que se utiliza XML. A continuación, se muestran distintos lenguajes de esquema que existen para crear las reglas sintácticas que definen las etiquetas de los documentos XML. También expondremos las razones que han llevado a la elección de los esquemas XML para definir dichas reglas, así como las ventajas que ofrecen al utilizarlos en la participación. Seguidamente, veremos los esquemas XML asociados a nuestra arquitectura, explicando detalladamente las características más importantes de cada uno de ellos. Por último, se exponen algunas conclusiones.

3.2. Comienzos de XML

El lenguaje XML (eXtensive Markup Language), véase XML (2011), proviene del lenguaje GML (General Markup Language) para organizar documentos técnicos con etiquetas estructuradas, inventado por IBM en los años 70. Surgió por la necesidad de manejar gran

cantidad de información sobre diferentes temas: gracias a este lenguaje se podía clasificar la información de forma correcta y luego procesarla adecuadamente. En 1986, la ISO (International Organization for Standardization), creó el SGML (Standard General Markup Language) para normalizar el lenguaje GML y buscar un estándar. A pesar de ser un buen lenguaje de marcado, los intérpretes encontraban serios problemas a la hora de analizar los documentos escritos en SGML, debido a que no se podían combinar elementos de diferentes vocabularios. Para solucionar dichos problemas surgió XML que permite:

- Mezclar elementos de diferentes lenguajes, es decir, es extensible.
- Crear analizadores simples, sin ninguna lógica especial.
- Rechazar documentos con errores de sintaxis.

XML se ha convertido en un estándar internacional desarrollado por el World Wide Web Consortium (W3C) que permite crear un conjunto de marcas para el procesamiento de la información. Una de sus principales características es su versatilidad, pudiendo manejar gran cantidad de información de distinto tipo. Como metalenguaje, XML define la sintaxis y los requisitos del lenguaje de marcado que se utiliza, lo que hace que la información se procese de manera estructurada y se gestione más eficientemente. En contraste con el lenguaje HTML, que pone el énfasis en el interfaz de usuario, los lenguajes basados en XML proporcionan medios para representar el contenido, la semántica y el esquema de los datos. Por estas razones, XML se utiliza en el desarrollo de diversos procesos como el planteamiento de la información que se quiere obtener, el almacenamiento, actualización y transmisión de dicha información y el filtrado de la misma.

Además, su uso puede influir en los siguientes parámetros:

- La calidad de la información, que determinará su utilidad.
- La precisión de la información obtenida, así como la calidad de los resultados obtenidos.

- La velocidad y el coste de obtener la información.
- La capacidad de comunicación entre procesos remotos.

XML permite la definición de un conjunto de etiquetas específicas que pueden usarse para el marcado de datos en distintas aplicaciones. Al facilitar el intercambio de datos utilizando un formato no propietario, XML es particularmente valioso para facilitar la interoperabilidad en entornos heterogéneos, como puede ser el caso de Internet. El tremendo crecimiento de las aplicaciones XML como lenguaje para representar e intercambiar datos, se ve reflejado en varios dominios de problemas. Algunos ejemplos son MathML (Mathematical Markup Language), véase Carlisle et al. (2009), que permite describir anotaciones matemáticas, manteniendo su estructura y contenido o CML (Chemical Markup Language), que permite describir moléculas junto con sus características físico-químicas y sus reacciones.

En el contexto de modelos de optimización, se han propuesto diversos lenguajes basados en XML como OSil (Optimization Service Instance Language), para representar problemas de optimización, entre ellos problemas de programación lineal, entera, cuadrática y, de manera muy general, problemas de programación no lineal, véase Fourer et al. (2010), o LPFML para representar modelos de programación lineal, que facilita la interoperabilidad entre lenguajes de modelización reduciendo el número de controladores necesarios, véase Fourer et al. (2005). Un proyecto muy importante que también utiliza XML para el intercambio de información es Decision Deck. El objetivo de dicho proyecto es desarrollar herramientas software que sirvan de ayuda en la toma de decisiones con criterios múltiples. Para ello, ha desarrollado un estándar XML denominado XMCD, ver Bisdorff et al. (2009), y definido por un esquema XML.

3.3. Lenguajes de esquema

Una vez que hemos revisado las ventajas que ofrece XML, nos encontramos con distintas formas para crear las reglas sintácticas que definirán las etiquetas de un documento XML en el ámbito de la participación electrónica.

Los métodos más extendidos para crear tales reglas son DTD (Document Type Definition), esquemas XML, RELAX NG y Schematron, aunque son más populares los dos primeros. Todos ofrecen ventajas e inconvenientes que mencionaremos brevemente a continuación:

1. **DTD.**

Es el formato de esquema más antiguo para validar documentos XML. Utiliza una sintaxis que no es XML para definir la estructura de un documento XML válido. Es el método más sencillo usado para validar los documentos XML. Por esta razón presenta varias limitaciones, ya que posee un lenguaje propio de escritura que causa problemas a la hora del aprendizaje, puesto que no sólo es necesario aprender XML sino que, además, hay que aprender el lenguaje de las DTD. No permite el uso de espacios de nombres, que permite tener dos elementos con el mismo nombre dentro del mismo contexto siempre y cuando tengan distinto prefijo. Tiene un tipado de documentos muy limitado, no pudiendo añadir nuevos tipos. Finalmente, su mecanismo de extensión es complejo y frágil.

2. **Esquema XML.**

Es el tipo de lenguaje más extendido. Los esquemas XML son una evolución de las DTDs y reciben el nombre de XSD (XML Schema Definition). Es un lenguaje de esquema complejo y potente que utiliza sintaxis XML, por lo que tiene gran cantidad de tipos de datos y permite especificar de forma detallada tipos de datos nuevos definidos por el usuario, llamados comúnmente “arquetipos”. Su característica más importante es la de poder extender arquetipos de un modo específico, es decir, permite lo que, en términos de orientación a objetos, recibe el nombre de herencia.

El problema de los esquemas XML aparece a la hora de validar el modelo, ya que su utilización supone un gran consumo de recursos y tiempo, debido a su gran especificación y a su complejidad en la sintaxis.

3. **RELAX NG.**

Apareció como la fusión de dos iniciativas, TREX y RELAX, en un intento de simplificar y potenciar la utilidad de las DTDs, en general, y los esquemas XML, en particular. RELAX NG es un lenguaje de esquema basado en gramática, muy intuitivo y fácil de usar, pero con el inconveniente de que las etiquetas que utiliza aportan poco significado a los componentes, además de contar con sólo dos tipos de datos definidos.

4. **Schematron.**

A diferencia de los anteriores, se basa en afirmaciones en vez de en la gramática. Al basarse en una serie de reglas, utiliza expresiones de acceso para definir lo que se permite en un documento XML. Si el documento cumple estas reglas, entonces es válido.

Con el propósito que se buscaba de gestionar y validar la información en *e*-participación, se ha decidido utilizar esquemas XML para crear un lenguaje que nos permita generar documentos XML válidos para la participación electrónica, debido a la facilidad de tratamiento de los esquemas que se deriva de la escritura en XML de los mismos, su potencia de trabajo, capacidad de ampliación, su uso extendido y la gran cantidad de documentación disponible.

Para la implementación del esquema del lenguaje EPML, se ha definido un conjunto de etiquetas válidas y las restricciones que afectan a las mismas. También se ha descrito la semántica del lenguaje para dar significado a cada etiqueta. El uso del esquema XML es recomendable puesto que las definiciones informales suelen ser ambiguas e incompletas, y una definición formal nos da la ventaja de poder extender el uso del lenguaje. Permite validar automáticamente los documentos XML creados, lo que permite alcanzar una fácil estandarización. Nos permite restringir los valores de los datos a un tipo de dato apropiado,

así como la estructura del mismo, con lo cual, únicamente se permite la construcción de documentos válidos. Pueden extenderse en un futuro para añadir características no presentes en un principio.

Dentro del campo de la participación ciudadana, el uso de esquemas XML también ofrece algunas ventajas específicas:

- Facilita el almacenamiento de toda la información relacionada con el proceso de participación, ofreciendo, además, una mejor integración con la infraestructura computacional.
- Son muy utilizados para el intercambio de datos entre servicios web, por lo que se facilita la integración y funcionamiento de los sistemas de participación a través de dichos servicios web.
- Los estándares de encriptación como “XML Encryption”, véase Lautenbach (2004), están emergiendo para los datos XML. Esta opción es muy importante para las aplicaciones que contienen datos confidenciales, como sería en nuestro caso.

3.4. Esquemas XML para una arquitectura de participación electrónica

Una vez introducidos los aspectos relevantes de XML en nuestro contexto, pasamos a comentar la estructura de los esquemas enfocados a los distintos componentes que forman nuestra arquitectura.

Todos los componentes de la arquitectura se basan en la misma estructura general, en la que se describe el problema y el perfil de los usuarios que pueden participar en cada fase. Sin embargo, cada módulo, puede tener unas características únicas que no aparecen en el resto. Esto es así porque, por ejemplo, los datos de entrada de la fase de debate no son los

mismos que los de la fase de comunicación de preferencias. Así, en esta sección, explicaremos los detalles generales del esquema XML que están definidos en la estructuración de problemas ya que es, generalmente, la primera etapa por la que debe pasar cualquier proceso participativo para, a continuación, explicar los detalles característicos del resto de fases. En la Figura 3.1 se muestra un boceto de cómo se compone el esquema propuesto.

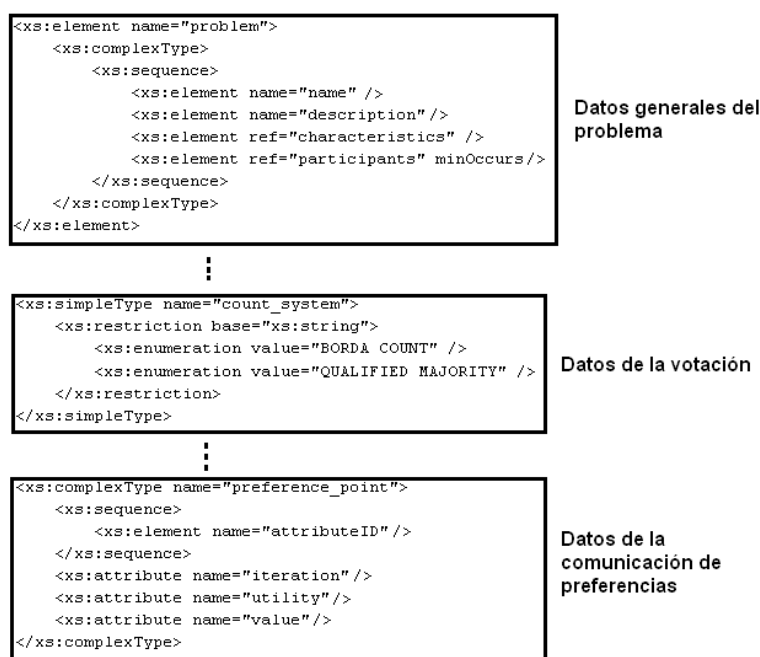


Figura 3.1: *Boceto del esquema XML propuesto*

Como comentamos en el capítulo anterior, el controlador hace las veces de catálogo de negocio, registrando los servicios (votación, negociación, etc ...) disponibles en la arquitectura, puesto que, distintos métodos de negociación pueden tener distintas localizaciones. Por ejemplo, el algoritmo de resolución de un método puede encontrarse en Finlandia y otro en Barcelona. De este modo, conseguimos que añadir un nuevo método de resolución resulte sencillo y se permite que cualquier usuario que desarrolle sus propios algoritmos pueda añadirlos al sistema, independientemente de la localización geográfica en la que se encuentre, únicamente, teniendo conexión a Internet.

3.4.1. Esquema XML del módulo de estructuración de problemas

El módulo de estructuración de problemas, como se expuso en el Capítulo 2, facilita la forma de organizar los problemas, identificar proyectos, restricciones y criterios de evaluación de los mismos. Su esquema XML puede dividirse en tres secciones principales:

- Datos básicos del problema (nombre, descripción,...).
- Características del mismo (alternativas, criterios, restricciones,...).
- Datos de los usuarios que participarán en el proceso de resolución del problema.

En estas tres estructuras, tenemos almacenada la información referente a la estructuración de un proceso de participación ciudadana. La definición del esquema XML del módulo de estructuración se muestra en la Tabla 3.1. El elemento `<problem>` se compone de cuatro elementos y un único atributo. El atributo `<id>` representa un identificador único para cada problema de participación que pudiera existir. Los elementos `<name>` y `<description>` contienen los datos básicos del problema. En el elemento `<characteristics>` almacenamos la información relevante del problema, como son los atributos, las opciones y las restricciones entre opciones. Por último, en el elemento `<participants>` tenemos los identificadores o “nicks” de los participantes que pueden acceder al problema.

```
1 <xs:element name="problem">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="name" type="xs:string" />
5       <xs:element name="description" type="xs:string" />
6       <xs:element ref="characteristics" />
7       <xs:element ref="participants" minOccurs="0" />
8     </xs:sequence>
9     <xs:attribute name="id" type="xs:string" use="required" />
10  </xs:complexType>
11 </xs:element>
```

Código 3.1: Estructura del elemento “problem”.

A su vez, algunos de los elementos descritos están formados por otros. Por ejemplo, el elemento `<characteristics>` se compone de `<options>`, `<attributes>` y `<constraints>`, que contendrán las opciones, atributos y restricciones del problema, respectivamente, consiguiendo, de este modo, tener las características más relevantes del problema unificadas en un solo elemento.

```
1 <xs:element name="characteristics">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element ref="options" minOccurs="0" maxOccurs="1" />
5       <xs:element ref="constraints" maxOccurs="1" />
6       <xs:element ref="attributes" maxOccurs="1" />
7     </xs:sequence>
8   </xs:complexType>
9 </xs:element>
```

Código 3.2: Estructura del elemento “characteristics”.

Dentro del elemento `<option>` cabe destacar el elemento `<attribute-option>` que representa la relación que existe entre un atributo y una opción. Dicho elemento se repetirá tantas veces como atributos contenga el problema y se compone del identificador de un atributo y de un valor.

```
1 <xs:element name="option">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="name" type="xs:string" />
5       <xs:element name="description" type="xs:string" />
6       <xs:element ref="attribute-option" maxOccurs="unbounded"/>
7     </xs:sequence>
8     <xs:attribute name="id" type="xs:string" use="required" />
9   </xs:complexType>
10 </xs:element>
11 <xs:element name="attribute-option">
12   <xs:complexType>
13     <xs:sequence>
14       <xs:element name="attribute" type="string_attribute" />
15       <xs:element name="value" type="xs:float" />
```

```
16 </xs:sequence>
17 <xs:attribute name="id" type="xs:string" use="required" />
18 </xs:complexType>
19 </xs:element>
```

Código 3.3: Estructura de los elementos “option” y “attribute-option”.

Otro elemento que es necesario destacar en el esquema son las restricciones. Entre las muchas posibles, en nuestro sistema hemos implementado dos tipos, las restricciones de número y las de dependencia. Las restricciones de número son aquellas en las que sólo se puede tomar un número determinado de opciones de todas las disponibles. Por ejemplo, cuando una gran multinacional necesita renovar sus equipos informáticos suele comprar todos de la misma marca. Si existen cuatro fabricantes, sólo se le compraría a uno. Por tanto, la restricción es de número porque hay que elegir un fabricante de entre cuatro. En el esquema, esta restricción se representa en el elemento `<numberConstraint>`, formado por dos elementos: `<numMax>`, que indica el número máximo de opciones que se pueden realizar, y tantos elementos `<option>` como opciones tenga el problema. En el caso del ejemplo mostrado en la Tabla 3.5, tendríamos el número 1 en el elemento `<numMax>` y cuatro elementos `<option>`.

```
1 <xs:complexType name="numberConstraint">
2 <xs:sequence>
3 <xs:element name="option" type="string_option" />
4 </xs:sequence>
5 <xs:attribute name="numMax" id="ID_numMax"
6 type="xs:nonNegativeInteger" />
7 </xs:complexType>
```

Código 3.4: Estructura del tipo “numberConstraint”.

Así, el XML resultante del ejemplo anterior sería:

```
1 <numberConstraint numMax="1">
2   <option id="1" />
3   <option id="2" />
4   <option id="3" />
5   <option id="4" />
6 </numberConstraint>
```

Código 3.5: Ejemplo del elemento “numberConstraint” con datos.

Las restricciones de dependencia, como su nombre indica, representan la dependencia de una opción respecto de otra. Por ejemplo, la construcción de un aparcamiento en un hospital, depende de la construcción previa de éste. Esta última restricción se representa en el elemento `<dependsConstraint>` formado por el elemento `<optionDepends>` que representa la opción de la que dependen las opciones que contiene. El identificador de dicho elemento se almacena en un atributo denominado “id”. De una opción pueden depender 0 o más opciones. En el ejemplo anterior, el elemento `<option>` sería el aparcamiento del hospital cuya construcción depende del elemento `<optionDepends>` que sería el propio hospital.

```
1 <xs:element name="constraint">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="numberConstraint" type="numberConstraint"
5         minOccurs="0" maxOccurs="unbounded" />
6       <xs:element name="dependsConstraint" type="dependsConstraint"
7         minOccurs="0" maxOccurs="unbounded" />
8       <xs:element name="budget" type="xs:float" minOccurs="0" />
9     </xs:sequence>
10  </xs:complexType>
11 </xs:element>
12 <xs:complexType name="dependsConstraint">
13   <xs:sequence>
14     <xs:element name="optionDepends">
15       <xs:complexType>
16         <xs:sequence>
17           <xs:element name="option" type="xs:string"
18             maxOccurs="unbounded" />
```

```
19     </xs:sequence>
20     <xs:attribute name="id" type="xs:string" />
21   </xs:complexType>
22 </xs:element>
23 </xs:sequence>
24 </xs:complexType>
```

Código 3.6: Estructura del elemento “constraint” y el tipo “dependsConstraint”.

Por último, otro elemento importante es la información relacionada con los participantes que intervendrán en el problema de participación creado. De cada participante se almacenará su nombre y apellidos, así como un número para identificarlo, como puede ser el Documento Nacional de Identidad (DNI) o el pasaporte. Estos datos estarán almacenados en la base de datos “Census”, véase Capítulo 2, accesible a todos los módulos del sistema. En el esquema XML, junto a cada participante, únicamente se almacenará un identificador de usuario o “nick”, que se utiliza para garantizar el anonimato del usuario durante todo el proceso, apareciendo dicho “nick” cada vez que el usuario realice alguna acción en el sistema.

```
1 <xs:element name="participant">
2   <xs:complexType>
3     <xs:attribute name="id" type="xs:string" use="required" />
4   </xs:complexType>
5 </xs:element>
```

Código 3.7: Estructura del elemento “participant”.

Cada elemento importante del esquema XML, como pueden ser atributos, opciones, restricciones o participantes, están incluidos en elementos más generales donde se indica el número de elementos que lo componen. Por ejemplo, el elemento <option> estará incluido dentro de otro denominado <options>, que tendrá un atributo que indica el número de elementos <option> que contiene. En el siguiente ejemplo, vemos cómo <options> está formado por dos elementos <option>. Así, el módulo encargado de recibir los datos en el documento XML puede conocer el número de elementos <option> que hay en el problema, únicamente accediendo al atributo “num_options”.

```
1 <options num_options="2">
2   <option id="1">
3     <name>option1</name>
4     <description>Description of option1</description>
5     <cost>1000</cost>
6   </option>
7   <option id="2">
8     <name>option2</name>
9     <description>Description of option2</description>
10    <cost>2000</cost>
11  </option>
12 </options>
```

Código 3.8: Ejemplo de documento XML con dos elementos “option”

3.4.2. Esquema XML del módulo de debate

El módulo de debate es, básicamente, un foro en el que los participantes pueden intercambiar ideas y mostrar sus opiniones respecto a los temas más relevantes del proceso de participación. El esquema XML de este módulo tiene una estructura jerárquica, puesto que el elemento raíz está formado por varios elementos, cada uno de los cuales contiene otros elementos y, así, sucesivamente. En el caso de este módulo, el elemento raíz es `<forum>` que representa un foro. Cada foro representará un problema de participación y tendrá la información relevante del problema, como son su nombre y descripción. También se almacenarán los participantes que podrán actuar en el proceso de debate y las categorías en que se divide el foro. Por tanto, el elemento `<forum>` tendrá la estructura representada en el Código 3.9.

```
1 <xs:element name="forum">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="name" type="xs:string" />
5       <xs:element name="description" type="xs:string" />
6       <xs:element ref="epar:categories" />
7       <xs:element ref="epar:participants"/>
8     </xs:sequence>
9     <xs:attribute name="id" type="xs:string" use="required" />
10  </xs:complexType>
11 </xs:element>
```

Código 3.9: Estructura del elemento “forum”.

Un foro se almacena con un identificador único, que lo diferenciará del resto de foros, representado por el atributo “id”. Dicho identificador es obligatorio para todos los elementos <forum>. Como vemos, cada foro puede tener varias categorías y varios participantes.

El elemento <category> está formado por varios elementos <topic>. Además, tendrá un nombre y descripción, así como un identificador único para diferenciarlo del resto de categorías.

```
1 <xs:element name="category">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="name" type="xs:string" />
5       <xs:element name="description" type="xs:string" />
6       <xs:element ref="epar:topics" />
7     </xs:sequence>
8     <xs:attribute name="id" type="xs:string" use="required" />
9   </xs:complexType>
10 </xs:element>
```

Código 3.10: Estructura del elemento “category”.

El siguiente elemento en la jerarquía es <topic>, que representa los temas que crean los participantes y sobre los que discutirán. El elemento <topic> se compone de un título, una

descripción, varios mensajes y un participante, que será quien ha creado el tema. Al igual que en los casos anteriores, tendrá un identificador para diferenciarlo del resto de temas.

```
1 <xs:element name="topic">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="title" type="xs:string" />
5       <xs:element name="description" type="xs:string" />
6       <xs:element name="participant" type="string_participant"/>
7       <xs:element ref="epar:messages" />
8     </xs:sequence>
9     <xs:attribute name="id" type="xs:string" use="required" />
10  </xs:complexType>
11 </xs:element>
```

Código 3.11: Estructura del elemento “topic”.

El último elemento del esquema XML es <message>, que almacena la información más relevante del módulo de debate, como son los mensajes enviados por los participantes, a partir de los cuales se sacarán las conclusiones correspondientes. Dicho elemento consta de un título, el cuerpo del mensaje, una fecha de creación del mensaje y el participante que lo creó. También tendrá como atributo un identificador único que hará que dicho mensaje aparezca en un tema, de una categoría, de un foro concreto.

```
1 <xs:element name="message">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="title" type="xs:string" />
5       <xs:element name="body" type="xs:string" />
6       <xs:element name="date" type="xs:date" />
7       <xs:element name="participant"
8         type="string_participant"/>
9     </xs:sequence>
10    <xs:attribute name="id" id="ID_message" use="required" />
11  </xs:complexType>
12 </xs:element>
```

Código 3.12: Estructura del elemento “message”.

Como en el caso del esquema XML de estructuración, los elementos `<category>`, `<topic>`, `<message>` y `<participant>` están englobados en elementos más generales que tienen como atributo el número de elementos que los componen. En el siguiente ejemplo, se ve cómo el elemento `<topics>` está constituido por tres elementos `<topic>`, uno de los cuales contiene un elemento `<messages>` formado por dos elementos `<message>` y, los otros dos, compuestos por un único elemento `<message>`.

```
1 <topics num_topics="3">
2   <topic id="1">
3     <title>topic1</title>
4     <description>Description of topic1</description>
5     <participant id="admin" />
6     <messages num_messages="2">
7       <message id="1" />
8       <message id="2" />
9     </messages>
10  </topic>
11  <topic id="2">
12    <title>topic2</title>
13    <description>Description of topic2</description>
14    <participant id="admin" />
15    <messages num_messages="1">
16      <message id="3" />
17    </messages>
18  </topic>
19  <topic id="3">
20    <title>topic3</title>
21    <description>Description of topic3</description>
22    <participant id="admin" />
23    <messages num_messages="1">
24      <message id="4" />
25    </messages>
26  </topic>
27 </topics>
```

Código 3.13: Ejemplo de documento XML con tres elementos “topic”.

El documento XML generado a partir de este esquema puede utilizarse como log de seguridad por sí durante el proceso participativo, o cuando finaliza, cualquier usuario o adminis-

trador necesitara realizar alguna consulta sobre lo sucedido durante el desarrollo de la fase de debate. También se podría generar dicho documento una vez finalizada esta fase, como posible copia de seguridad en un formato distinto al almacenado en la base de datos. Pero, sobretodo, este documento puede utilizarse para obtener la información más destacada de todos los temas discutidos durante el debate, por los usuarios, mediante la técnica de minería de textos, véase Tan (1999), lo que nos puede permitir acelerar el ciclo de decisiones al acortar el tiempo necesario para identificar, encontrar o descubrir información relevante, como se explica en el Capítulo 5.

3.4.3. Esquema XML del módulo de comunicación de preferencias

El módulo de comunicación de preferencias es uno de los más importantes de la arquitectura. Se basa en un modelo de función de valor aditiva, véase Von Winterfeldt y Edwards (1986), sin mucha pérdida de generalidad. Además, es uno de los más complejos debido a la forma de calcular los valores de cada opción, dadas las preferencias de los participantes. La forma de almacenar los datos en el documento XML se adapta al modelo de comunicación de preferencias de la arquitectura implementada en el Capítulo 2.

Una vez que el sistema ha recogido las preferencias, puede calcular la función de valor de cada una de las opciones del problema. Específicamente, de cada participante se obtiene el peso que le asigna a los diferentes criterios del problema y cinco pares de puntos que permitan obtener una función lineal a trozos para conocer la utilidad del criterio evaluado. De estos cinco puntos, dos deben que ser el mínimo y el máximo que puede tomar el criterio y que representan el peor y el mejor valor, o viceversa. Toda esta información se representa de la siguiente forma:

$$P(w_i, v_i(x_j, y_j))$$

donde $i \in \{1..n\}$ representa el i -ésimo atributo; $j \in \{1, \dots, 5\}$ representa el j -ésimo valor

en tal atributo; w_i representa el peso del atributo i -ésimo; $w_i \geq 0$; $\sum w_i = 1$; v_i representa el vector con cinco pares de puntos para el atributo i ; x_j representa el valor del atributo e $y_j \in [0, 100]$ representa el valor asignado a x_j . Entonces, para una alternativa o_k , con valor z_{ik} , que representa la relación entre un atributo y una alternativa, el valor para tal alternativa debería ser:

$$v(o_k) = \sum_{i=1}^n w_i \cdot d_i(v_i, z_{ik})$$

donde para calcular $d(v_i, z_{ik})$ se busca cuál es la posición del valor z_{ik} en el vector v_i y calculamos su valor con el siguiente algoritmo:

```

IF ( $z_{ik} \neq x_1$ )
  WHILE  $z_{ik} \geq x_j$ 
     $j=j+1$ ;
   $d_i(v_i, z_{ki}) = y_{j-1} + \left( \frac{(z_{ik}-x_{j-1}) \cdot (y_j-y_{j-1})}{(x_j-x_{j-1})} \right)$ 
ELSE
   $d_i(v_i, z_{ki}) = 0$ 

```

El esquema XML de este módulo es similar en organización al de estructuración de problemas. Está dividido en tres secciones principales: datos básicos del problema, características del mismo y participantes del proceso participativo. La principal diferencia respecto al módulo de estructuración es que para cada participante se almacena un perfil con sus preferencias respecto a cada atributo del problema y las utilidades obtenidas para cada opción del mismo. Por lo tanto, el elemento <participant> tendrá la estructura:

```
1 <xs:element name="participant">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="profile" type="profile"
5         minOccurs="0" />
6     </xs:sequence>
7     <xs:attribute name="id" type="xs:string" use="required" />
8   </xs:complexType>
9 </xs:element>
```

Código 3.14: Estructura del elemento “participant”.

El tipo complejo `<profile>` consta de dos elementos que se encargan de almacenar las preferencias que muestra el participante por cada atributo del problema y la función de valor calculada para cada opción o alternativa a partir de las preferencias comunicadas por el usuario. Los elementos que forman el tipo `<profile>` son `<preferences_participant>` y `<utility_function>`.

```
1 <xs:complexType name="profile">
2   <xs:sequence>
3     <xs:element ref="utility_function" minOccurs="0" />
4     <xs:element ref="preferences_participant" minOccurs="0" />
5   </xs:sequence>
6 </xs:complexType>
```

Código 3.15: Estructura del tipo “profile”.

En primer lugar, el participante comunica cuáles son su mejor y peor valores a través del máximo y el mínimo que puede tomar cada atributo del problema. También debe indicar el peso que le asigna a cada atributo. Toda esta información se encuentra almacenada en el tipo `<preference_participant>`, formado por cuatro elementos. Una referencia al elemento `<attribute>` para saber a qué atributo se le asignan los valores, un elemento `<best>` para almacenar el valor favorito del participante para ese atributo, un elemento `<worst>` que almacena el peor valor y el elemento `<weight>` que contiene el peso que asigna el participante a tal atributo. La estructura de estos elementos puede verse en el Código 3.16.

```
1 <xs:complexType name="preference_participant">
2   <xs:sequence>
3     <xs:element name="attribute" type="xs:string"/>
4     <xs:element name="worst" type="xs:long" />
5     <xs:element name="best" type="xs:long" />
6     <xs:element name="weighth">
7       <xs:simpleType>
8         <xs:restriction base="xs:decimal">
9           <xs:minInclusive value="0"/>
10          <xs:maxInclusive value="1"/>
11        </xs:restriction>
12      </xs:simpleType>
13    </xs:element>
14    <xs:element ref="preference_points"/>
15  </xs:sequence>
16 </xs:complexType>
17 <xs:simpleType name="range">
18   <xs:restriction base="xs:integer">
19     <xs:minInclusive value="0" />
20     <xs:maxInclusive value="100" />
21   </xs:restriction>
22 </xs:simpleType>
```

Código 3.16: Estructura de los tipos “preference_participant” y “range”.

A continuación, cada participante deberá dar información adicional al sistema para que éste pueda conocer sus preferencias y calcular la utilidad de cada opción del problema. Las preferencias del participante, se almacenan en una función de valor lineal a trozos, en la que, se deben indicar tres puntos entre el peor y el mejor valor que puede ser asignado a cada atributo. A cada uno de los puntos le asigna una utilidad distinta. Por ejemplo, si hubiera que valorar un atributo cuyo valor mínimo sea 20 y el valor máximo 100, el participante podría asignar la utilidad 25 al valor 40, la utilidad 50 al valor 60 y la utilidad 60 al valor 80. De este modo, el sistema cuenta con cinco puntos distintos para evaluar qué utilidad da el usuario a ese atributo. El resultado de dicho ejemplo puede verse en la Figura 3.2.

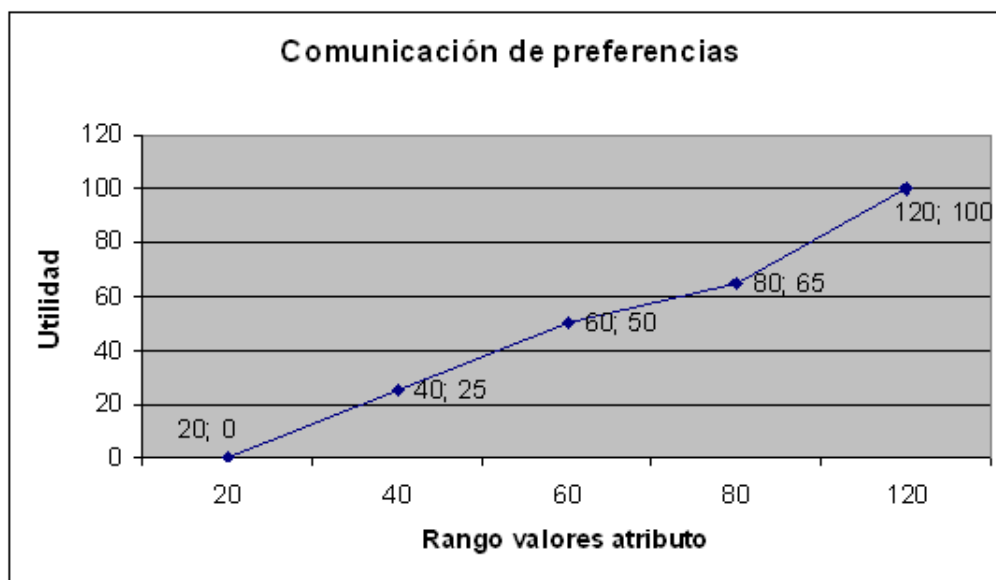


Figura 3.2: Gráfica obtenida una vez que el usuario ha indicado sus preferencias.

Toda esta información queda almacenada en el elemento `<preference_point>` que consta del atributo “utility” para indicar la utilidad de dicho punto, el atributo “value” que es un valor intermedio dentro del rango de valores que puede tomar el atributo y el atributo “iteration”, para identificar cada punto. Este último atributo es de tipo “iteration_value”, que es un número entero restringido a los valores uno, dos o tres, identificando el punto.

```

1 <xs:complexType name="preference_point">
2   <xs:attribute name="iteration" type="iteration_values" />
3   <xs:attribute name="utility" type="range" />
4   <xs:attribute name="value" type="xs:long" />
5 </xs:complexType>
6 <xs:simpleType name="iteration_values">
7   <xs:restriction base="xs:integer">
8     <xs:minInclusive value="1" />
9     <xs:maxInclusive value="3" />
10  </xs:restriction>
11 </xs:simpleType>

```

Código 3.17: Estructura de los tipos “preference_point” e “iteration_values”.

El documento XML que se genera a partir del ejemplo de la Figura 3.2 es:

```
1 <preference_points>
2   <point iteration="1" utility="25" value="40"/>
3   <point iteration="2" utility="50" value="60"/>
4   <point iteration="3" utility="65" value="80"/>
5 </preference_points>
```

Código 3.18: Documento XML generado con los datos del ejemplo de la Figura 3.2.

Por último, se calcula el valor del participante para cada una de las opciones. La forma de calcular esta función de utilidad es independiente del método de asignación de preferencias a pesar de que es necesario que el participante haya comunicado sus preferencias previamente. Esta utilidad se almacena en el elemento `<utility_function>` que consta de un atributo “id” para identificar esa utilidad y un elemento `<option>`, que es la opción que recibe el valor para ese participante con un atributo “id” para identificar la opción y un atributo “utility” con la utilidad de la opción para dicho participante.

```
1 <xs:element name="utility_function">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="option" maxOccurs="unbounded">
5         <xs:complexType>
6           <xs:attribute name="id" type="xs:string" />
7           <xs:attribute name="utility" type="xs:long" />
8         </xs:complexType>
9       </xs:element>
10    </xs:sequence>
11    <xs:attribute name="id" id="utility_functionID"
12      use="required" />
13  </xs:complexType>
14 </xs:element>
```

Código 3.19: Estructura del elemento “utility_function”.

3.4.4. Esquema XML del módulo de negociación

El módulo de negociación implementa distintos protocolos de negociación, entre ellos POSTING y BIM, véase Ríos y Ríos Insua (2008, 2010) para más información, cada uno con sus propias reglas de actuación. La negociación por POSTING es similar a las características ofrecidas por el módulo de debate, pero presenta ciertas diferencias respecto a éste. En primer lugar, en este módulo, los participantes conocen la utilidad de las alternativas del problema dadas sus preferencias, una vez que se ha pasado por el módulo de comunicación de preferencias. Esto es importante a la hora de sugerir una nueva oferta, puesto que, de un simple vistazo, el participante puede ver qué opciones son preferibles respecto de otras. Además, otra ventaja importante es que el participante puede publicar sugerencias que reciben votos por parte del resto de participantes. De este modo, el esquema XML del módulo de negociación adopta una forma similar a los esquemas de los módulos de estructuración de problemas y de debate, aunque con ligeras diferencias.

La primera se encuentra en la forma de almacenar los datos de los participantes. Estos contienen los mismos datos básicos que en los casos anteriores, pero añaden tres nuevos elementos: “participant”, “utility_option” y “offer”.

```
1 <xs:element name="participant">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element ref="utility_option" maxOccurs="1" />
5       <xs:element ref="optimal_solution" maxOccurs="1" />
6       <xs:element ref="offer" />
7     </xs:sequence>
8     <xs:attribute name="id" type="xs:string" use="required" />
9   </xs:complexType>
10 </xs:element>
11 <xs:element name="utility_option">
12   <xs:complexType>
13     <xs:sequence>
14       <xs:element name="option" maxOccurs="unbounded">
15         <xs:complexType>
16           <xs:attribute name="id" type="xs:string" />
17           <xs:attribute name="utility" type="xs:long" />
```

```

18     </xs:complexType>
19   </xs:element>
20 </xs:sequence>
21 </xs:complexType>
22 </xs:element>
23 <xs:element name="offer">
24   <xs:complexType>
25     <xs:sequence>
26       <xs:element ref="epar:options" />
27       <xs:element ref="epar:messages" />
28     </xs:sequence>
29   </xs:complexType>
30 </xs:element>

```

Código 3.20: Estructura de los elementos “participant”, “utility_option” y “offer”.

En los elementos `<utility_option>` y `<optimal_solution>`, se encuentran las utilidades que el participante obtiene de cada opción y la solución óptima de ese participante, respectivamente. Ambos elementos sólo toman valores en caso de que el proceso participativo pase previamente por el módulo de comunicación de preferencias. La etiqueta `<optimal_solution>` también puede aparecer en el documento final si ha concluido la fase de negociación y el sistema ha encontrado una solución aceptable para todos los participantes que han intervenido en la misma. Esta etiqueta consta de un elemento `<options>`, donde se guardan las opciones que forman parte de la solución, la utilidad de todas las opciones, al igual que en el caso anterior, siempre que el proceso haya pasado por una fase de comunicación de preferencias y el método de resolución utilizado, BIM o POSTING.

```

1 <xs:element name="optimal_solution">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element ref="epar:options" />
5       <xs:element name="utility" type="xs:float" />
6       <xs:element name="method" type="epar:methods_neg" />
7     </xs:sequence>
8   </xs:complexType>
9 </xs:element>

```

Código 3.21: Estructura del elemento “optimal_solution”.

Por último, el elemento `<offer>` consta de dos elementos. El elemento `<options>` está formado por un conjunto de opciones elegidas por el participante para realizar la oferta; el elemento `<messages>` tiene la misma estructura que en el módulo de debate. En este caso, almacena el conjunto de mensajes escritos por los participantes, mostrando su opinión acerca del conjunto de opciones que forman la oferta. Otro elemento diferenciador de este esquema XML es `<offer_vote>` donde quedan registrados los votos que reciben las ofertas realizadas por los participantes. Consta de los elementos `<offer>` que contiene la oferta con la lista de opciones que la conforma, el elemento `<participant>` para conocer el participante que ha emitido su voto y el elemento `<accept>`, de tipo “boolean”, que toma los valores “true”, si el voto emitido por el participante es a favor de dicha lista de opciones, o “false”, en caso de que el voto sea en contra.

```
1 <xs:element name="offer_vote">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element ref="epar:offer" />
5       <xs:element ref="epar:participant" />
6       <xs:element name="accept" type="xs:boolean" />
7     </xs:sequence>
8   </xs:complexType>
9 </xs:element>
```

Código 3.22: Estructura del elemento “offer_vote”.

Al igual que en los casos anteriores, cuando se tratan elementos concretos que pueden ser contenidos en elementos más generales, estos últimos guardan en un atributo el número de elementos que lo forman. Por ejemplo, es el caso de `<messages>`, que almacena en el atributo “num_messages” el número de mensajes que contiene.

```
1 <xs:element name="messages">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element ref="epar:message" maxOccurs="unbounded" />
5     </xs:sequence>
6     <xs:attribute name="num_messages"
7       type="xs:nonNegativeInteger"/>
8   </xs:complexType>
9 </xs:element>
```

Código 3.23: Estructura del elemento “messages”.

3.4.5. Esquema XML del módulo de votación

En el esquema XML de este módulo, el elemento <problem>, que es el raíz de todos los esquemas descritos anteriormente, añade un nuevo elemento, <voting>, que almacena la información importante de la votación. Esta información puede ser el tipo de conteo, el número máximo y mínimo de opciones marcadas por cada voto, el número máximo y mínimo de puntos asignados a cada opción, el porcentaje mínimo para ganar la votación en caso de que el sistema de conteo sea mayoría cualificada y, en caso de que haya finalizado la fase, la solución finalmente adoptada.

```
1 <xs:element name="voting">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="percentage" type="xs:decimal"
5         minOccurs="0"/>
6       <xs:element ref="epar:solution" />
7     </xs:sequence>
8     <xs:attribute name="id" type="xs:string" use="required" />
9   </xs:complexType>
10 </xs:element>
```

Código 3.24: Estructura del elemento “voting”.

En `<participant>`, se añade un nuevo elemento `<vote>` donde se guarda el voto emitido por tal participante. `<vote>` está formado por dos elementos `<vote_type>` y `<options>`.

```
1 <xs:element name="participant">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element ref="epar:vote" />
5     </xs:sequence>
6     <xs:attribute name="id" type="xs:string" use="required" />
7   </xs:complexType>
8 </xs:element>
9 <xs:element name="vote_type">
10  <xs:simpleType>
11    <xs:restriction base="xs:integer">
12      <xs:enumeration value="0" />
13      <xs:enumeration value="1" />
14      <xs:enumeration value="2" />
15    </xs:restriction>
16  </xs:simpleType>
17 </xs:element>
18 <xs:element name="vote">
19  <xs:complexType>
20    <xs:sequence>
21      <xs:element ref="epar:vote_type" />
22      <xs:element ref="epar:options" />
23    </xs:sequence>
24  </xs:complexType>
25 </xs:element>
```

Código 3.25: Estructura de los elementos “participant”, “vote_type” y “vote”.

El voto de cada participante es anónimo porque el elemento “participant” únicamente contiene el identificador del participante, que es una ristra de caracteres alfanuméricos generados aleatoriamente por el sistema, una vez que el usuario se ha dado de alta.

En el elemento `<vote_type>` queda almacenado el tipo de voto que ha realizado el usuario. Puede tomar tres valores: 0, si el voto emitido es nulo; 1, si el voto es válido y 2, si no es válido. En el elemento `<options>` quedan almacenadas las opciones sobre las que el participante ha emitido algún voto a favor.

En el esquema XML de este módulo el elemento `<option>` ha sufrido una modificación respecto al resto de módulos, añadiendo un nuevo elemento `<mark_intensity>`, cuya finalidad es guardar el número de puntos que un participante asigna a una determinada opción. Esto es así porque, dependiendo del sistema de votación que se haya seleccionado para el proceso participativo, las opciones pueden recibir 0, 1 o varios puntos. Por ejemplo, en caso de que el sistema de votación seleccionado sea por votación acumulativa, un mismo participante puede contar con mayor número de votos que opciones haya en el problema, por lo que alguna opción debe recibir más de un voto.

Otro elemento significativo del esquema XML de este módulo es `<solution>`, que permite guardar la solución obtenida, así como el sistema de conteo utilizado, una vez que ha terminado el proceso de votación.

```
1 <xs:element name="solution">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="num_votes" type="nonNegativeInteger" />
5       <xs:element ref="epar:options" />
6       <xs:element name="count_system" type="count_system" />
7     </xs:sequence>
8   </xs:complexType>
9 </xs:element>
```

Código 3.26: Estructura del elemento “solution”.

Está formado por tres elementos: `<options>`, donde está la lista de opciones que forman la solución final; `<num_votes>`, que almacena el número de votos recibidos por las opciones que forman parte de la solución; y el elemento `<count_system>`, que indica el sistema de conteo utilizado para obtener la solución.

El documento XML de esta fase puede generarse únicamente con el elemento “solution”, que contiene la información relevante y de mayor aceptación, pero se ha decidido que sea posible añadir todos los datos de la votación para que pueda ser utilizado como log de seguridad en caso de que se requiera algún tipo de consulta, o para ser almacenado como

posible copia de seguridad.

```
1 <problem id="1">
2   <name>Problem name</name>
3   <description>Problem description</description>
4   <voting id="1">
5     <percentage>80.0</percentage>
6     <solution>
7       <numVotes>50</numVotes>
8       <options numOptions="4">
9         <option id="1">
10          <name>Option 1</name>
11          <description>Description option 1</description>
12          <markIntensity>35</markIntensity>
13        </option>
14        <option id="2">
15          <name>Option 2</name>
16          <description>Description option 2</description>
17          <markIntensity>18</markIntensity>
18        </option>
19        <option id="3">
20          <name>Option 3</name>
21          <description>Description option 3</description>
22          <markIntensity>23</markIntensity>
23        </option>
24      </options>
25      <count_system>CUMULATIVE VOTING</count_system>
26    </solution>
27  </voting>
28 </problem>
```

Código 3.27: Ejemplo de XML generado únicamente con el elemento “solution”.

3.4.6. Esquema XML del módulo de arbitraje

El módulo de arbitraje, en caso de formar parte del proceso, debe ejecutarse una vez que haya terminado la fase de comunicación de preferencias, puesto que necesita disponer de las preferencias de los participantes para poder calcular una solución arbitrada. De este modo, el

esquema XML del módulo de arbitraje tiene una estructura similar a la fase de comunicación de preferencias, pero requiere añadir elementos que los diferencian. Así, podemos observar en el Cuadro 3.28 cómo se ha añadido el elemento <solutions>.

```
1 <xs:element name="problem">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="name" type="xs:string" />
5       <xs:element name="description" type="xs:string" />
6       <xs:element ref="epar:characteristics" />
7       <xs:element ref="epar:participants" maxOccurs="1" />
8       <xs:element ref="epar:solutions" maxOccurs="1" />
9     </xs:sequence>
10    <xs:attribute name="id" type="xs:string" use="required" />
11  </xs:complexType>
12 </xs:element>
```

Código 3.28: Estructura del elemento “problem” del módulo de arbitraje.

Este elemento <solutions> contiene la solución o soluciones óptimas y está formado por un atributo llamado “num_solutions” para indicar el número de soluciones óptimas encontradas por el sistema y el elemento <solution> con la solución propuesta por el árbitro, en este caso la aplicación. A su vez, el elemento <solution> se compone de dos elementos, <options> que muestra las opciones que forman la solución, y <method>, que especifica el método utilizado por el sistema para obtener la solución arbitrada.

```
1 <xs:element name="solution">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="method" type="epar:methods" />
5       <xs:element ref="epar:options" />
6     </xs:sequence>
7   </xs:complexType>
8 </xs:element>
9 <xs:element name="solutions">
10  <xs:complexType>
11  <xs:sequence>
12    <xs:element ref="epar:solution" maxOccurs="unbounded" />
```

```
13 </xs:sequence>
14 <xs:attribute name="num_solutions"
15     type="xs:nonNegativeInteger"/>
16 </xs:complexType>
17 </xs:element>
```

Código 3.29: Estructura de los elementos “solutions” y “solution”.

Los valores que puede tomar el elemento <method> se especifican en el “tipo simple” denominado “methods”. Estos valores, que son cadenas de caracteres, son predeterminados y sirven para indicar el método que se ha utilizado para encontrar la mejor solución.

```
1 <xs:simpleType name="methods">
2 <xs:restriction base="xs:string">
3 <xs:enumeration value="Nash" />
4 <xs:enumeration value="Kalai-Smorodinsky" />
5 <xs:enumeration value="BIM" />
6 </xs:restriction>
7 </xs:simpleType>
```

Código 3.30: Estructura del elemento “methods”.

Los algoritmos considerados para incluir en el módulo de arbitraje son “Nash”, véase Nash (1950), “Kalai-Smorodinsky”, véase Kalai y Smorodinsky (1975), y “BIM” (Balanced Increment Method), véase Ríos y Ríos Insua (2010), como se ve en el Código 3.30.

3.4.7. Esquema XML del controlador

El controlador es el módulo encargado de gestionar todo el proceso participativo. Por lo tanto, contendrá la información necesaria para saber qué módulo debe actuar en una determinada fecha, así como, su fecha de finalización.

El esquema XML está formado por el elemento <problem>, que permite conocer el problema al que pertenece cada fase y donde únicamente se almacena como atributo el identificador y una referencia a <controller>. Este último guarda como atributo un identificador

único que permite diferenciarlo del resto de controladores y, como elementos, una referencia a `<phase>` con cada una de las fases que forman el proceso participativo, la fecha de inicio (`<start>`) y de fin (`<finish>`) del mismo.

```

1 <xs:element name="controller">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element ref="phase" minOccurs="0" />
5     </xs:sequence>
6     <xs:attribute name="id" type="xs:string"/>
7   </xs:complexType>
8 </xs:element>
9 <xs:element name="problem">
10  <xs:complexType>
11    <xs:sequence>
12      <xs:element ref="controller" maxOccurs="unbounded"/>
13    </xs:sequence>
14    <xs:attribute name="id" type="xs:string" />
15  </xs:complexType>
16 </xs:element>

```

Código 3.31: Estructura de los elementos “problem” y “controller”.

El elemento `<phase>` toma distintos valores dependiendo de la fase que haya que almacenar mostrados en el siguiente esquema XML.

```

1 <xs:element name="phase">
2   <xs:complexType>
3     <xs:sequence>
4       <xs:element name="structuring" type="phaseData"
5         maxOccurs="unbounded" />
6       <xs:element name="debate" type="phaseData"
7         maxOccurs="unbounded" />
8       <xs:element name="preferences" type="phaseData"
9         maxOccurs="unbounded" />
10      <xs:element name="negotiation" maxOccurs="unbounded" />
11      <xs:element name="voting" maxOccurs="unbounded" />
12      <xs:element name="arbitration" maxOccurs="unbounded" />
13      <xs:element name="suggestion" type="phaseData"
14        maxOccurs="unbounded" />

```

```
15 <xs:element name="sampling" type="phaseData"
16     maxOccurs="unbounded" />
17 </xs:sequence>
18 </xs:complexType>
19 </xs:element>
```

Código 3.32: Estructura del elemento “phase”.

Como vemos, todas las fases son de tipo “phaseData”, que indica la fecha de inicio y fin, un identificador único que representa la fase y la dirección donde se aloja el servicio web que da solución a la fase. También se guardan los roles los usuarios que pueden tomar parte en dicha fase. El contenido de dicho tipo lo puede verse en el Código 3.33:

```
1 <xs:complexType name="phaseData" abstract="true">
2   <xs:sequence>
3     <xs:element name="start" type="xs:dateTime" minOccurs="1"
4       maxOccurs="1"/>
5     <xs:element name="finish" type="xs:dateTime" minOccurs="1"
6       maxOccurs="1"/>
7     <xs:element name="roles" type="roles"
8       maxOccurs="unbounded" />
9   </xs:sequence>
10  <xs:attribute name="url" type="xs:string" use="required" />
11  <xs:attribute name="id" type="xs:string"/>
12 </xs:complexType>
```

Código 3.33: Estructura del elemento “phase”.

Además, los elementos “negotiation”, “voting” y “arbitration” constan de un elemento adicional que representa el método de resolución en el caso de la negociación y el arbitraje, y el sistema de conteo para la votación. Los métodos implementados hasta el momento están representados en el Código 3.34.

```
1 <xs:simpleType name="methods_arb">
2   <xs:restriction base="xs:string">
3     <xs:enumeration value="Nash" />
4     <xs:enumeration value="Kalai-Smorodinsky" />
5     <xs:enumeration value="BIM" />
```

```
6 </xs:restriction>
7 </xs:simpleType>
8 <xs:simpleType name="methods_neg">
9 <xs:restriction base="xs:string">
10 <xs:enumeration value="POSTING" />
11 <xs:enumeration value="BIM" />
12 </xs:restriction>
13 </xs:simpleType>
14 <xs:simpleType name="count_system">
15 <xs:restriction base="xs:string">
16 <xs:enumeration value="Borda count" />
17 <xs:enumeration value="Cumulative voting" />
18 <xs:enumeration value="Simple majority" />
19 <xs:enumeration value="Qualified majority" />
20 </xs:restriction>
21 </xs:simpleType>
```

Código 3.34: Métodos de resolución y sistemas de conteo.

Por último, podemos ver en el Código 3.35 un ejemplo de documento XML generado en el controlador en el que podemos ver cómo el problema consta de tres fases. La primera fase, de estructuración, da comienzo el 15 de Junio de 2011 y finaliza el día 16 del mismo mes. A continuación, una fase de comunicación de preferencias, en la que los participantes dispondrán de dos días para transmitir sus prioridades al sistema. Para finalizar, una fase de votación en la que los usuarios disponen de un único voto y pueden asignarlo a un solo proyecto. El sistema de conteo utilizado es el de mayoría simple.

```
1 <problem id="1">
2 <controller id="1">
3 <phase>
4 <structuring id="1" url="http://appl/WS/strucService.asmx">
5 <start>2011/06/15 00:00</start>
6 <finish>2011/06/16 00:00</finish>
7 </structuring>
8 <preferences id="2" url="http://appl/WS/prefService.asmx">
9 <start>2011/06/16 00:00</start>
10 <finish>2011/06/18 00:00</finish>
11 </preferences>
12 <voting id="3" url="http://appl/WS/votingService.asmx">
```

```
13 <start>2011/06/18 00:00</start>
14 <finish>2011/06/19 00:00</finish>
15 <count_system>Simple majority</count_system>
16 <options_by_vote min="0" max="1" />
17 <points_by_option min="0" max="1" />
18 </voting>
19 </phase>
20 </controller>
21 </problem>
```

Código 3.35: Ejemplo de documento generado en el controlador.

3.5. Conclusiones

Hemos visto cómo en cualquier ámbito de trabajo sobre sistemas Web, la utilización de la tecnología XML es muy importante debido a las facilidades que aporta. De hecho, es el estándar de transmisión de información más utilizado en numerosos campos ya que permite crear un conjunto de marcas, pudiendo procesar gran cantidad de información de distinto tipo. Gracias a este conjunto de marcas, la información puede tratarse de manera estructurada y, de este modo, se gestiona de forma más eficiente.

Aprovechando el potencial de XML para el intercambio de información pretendemos que las reglas sintácticas desarrolladas en esta Tesis se conviertan en un referente dentro del contexto de la e-democracia.

Del mismo modo, sería interesante crear un repositorio donde se almacenen esquemas XML estándar para los distintos mecanismos de participación. Así, se consigue un ahorro considerable de tiempo y trabajo, cuando sea necesario utilizar una fase que es común a varios procesos. El repositorio contaría inicialmente con los procesos definidos anteriormente y, a medida que se fueran creando nuevos procesos, se añadirían al mismo, consiguiendo con el paso del tiempo un gran almacén de procesos participativos listos para ser utilizados y con el único esfuerzo de cargarlos en el sistema.

Capítulo 4

ePBPM: Un lenguaje gráfico de definición de procesos participativos

4.1. Introducción

Una vez presentada la arquitectura propuesta para dar soporte a una plataforma de diseño de procesos participativos y la estructura de los documentos XML construidos a partir de los diversos componentes de dicha arquitectura, parece conveniente crear un mecanismo sencillo de definición de tales procesos. Para ello, nos basamos en los sistemas de gestión de procesos de negocio (BPM) y presentaremos ePBPM un lenguaje gráfico creado para diseñar procesos de una forma sencilla e intuitiva.

El diseño de procesos por parte de expertos es un problema de gran complejidad porque es necesario conocer profundamente la definición de cada módulo del proceso, los roles de los participantes en el mismo, cuáles son los procesos más críticos, entre otros factores. Por ejemplo, en una empresa de paquetería es necesario conocer cuál es el ciclo de vida del paquete a enviar, desde que entra en la oficina hasta que se entrega al cliente final. Necesi-

tamos conocer cuáles son las etapas por las que pasa el paquete, como pueden ser el modo de clasificación (tamaño, peso, destino) o el rol de las personas que manejan el mismo. En el caso que nos ocupa, el diseño de instrumentos participativos, es muy importante conocer qué objetivos perseguimos, la duración aproximada del proceso, es decir, si contamos con poco o mucho tiempo para tomar la decisión, el ámbito del mismo, si las decisiones tomadas van a ser vinculantes o consultivas, qué personas pueden participar (representantes de asociaciones, mayores de edad, expertos en el problema, ...). Por otro lado, si además de representar el proceso, éste se quiere interpretar, el problema se agrava aún mas. Según el modelo tradicional de diseño de procesos, si se quieren aprovechar los sistemas informáticos para supervisar y apoyar la implantación de los procesos diseñados resulta necesario crear un software que lo haga específicamente. Por ello, cuando un experto en procesos quiere poner en marcha uno, ha de recurrir a un programador para que construya el programa adecuado. Además, un experto en procesos puede decidir en cualquier momento cambiar el flujo de ejecución para hacerlo más adecuado, lo que requeriría de nuevo la mediación del personal informático. Esta mediación puede conllevar errores y retrasos en la implantación de procesos, incluso si son muy simples.

Este problema es resoluble dotando a los diseñadores de procesos de herramientas para poder diseñar los mismos. Para ello, es necesario encontrar modelos que nos permitan realizar de modo natural la automatización de los procesos para poder estandarizarlos. Los Sistemas de Gestión de Procesos de Negocio (*Business Process Management Systems*, BPMS) son plataformas software que permiten el modelado, despliegue y seguimiento de procesos de negocio o flujos de trabajo. Además, proporcionan un nuevo paradigma orientado a procesos para crear aplicaciones de gestión de los mismos. Un flujo de trabajo (del inglés *Workflow*) define acciones, y reacciones entre éstas, que estandarizan la planificación de un proceso. Estas acciones pueden ser realizadas por humanos que interpretan el flujo de trabajo descrito o, automáticamente, por sistemas informáticos, véase WfMCTG (1999).

Los BPMS proporcionan herramientas para la simulación, monitorización y ajuste de los procesos de negocio. Integran una serie de funcionalidades básicas para la correcta aplicación de la metodología BPM, que incluyen:

- Un modelador que permite expresar gráficamente procesos de negocio o flujos de trabajo de una forma comprensible a todos los usuarios de una organización.
- Un motor de procesos que permita ejecutar de forma automática los procesos modelados gráficamente.
- Un entorno de administración para el control y gestión del estado de los procesos y los usuarios.
- Un modelador de formularios para dar forma a las pantallas de introducción / visualización de datos que se mostrarán a los usuarios.
- Un entorno de simulación y análisis para detectar oportunidades de mejora de los procesos de negocio por medio de posibles escenarios (what-if).
- La integración de aplicaciones que permita el intercambio ágil de información entre los procesos y el resto de sistemas.

La Gestión de Procesos de Negocio ayuda a las organizaciones a aumentar la satisfacción de los clientes, la calidad de los productos o servicios y disminuye el tiempo de entrega, según un estudio empírico realizado por Kohlbacher (2009). Además, los Sistemas de Gestión de Procesos de Negocio conllevan una serie de actividades a realizar, véase Campos y de Almeida (2010), como podemos ver en la Figura 4.1:

- Modelado de la definición y creación de los procesos mediante una notación formal.
- Ejecución o simulación de los mismos.
- Análisis y monitorización de las ejecuciones.
- Optimización de los procesos, una vez analizados.

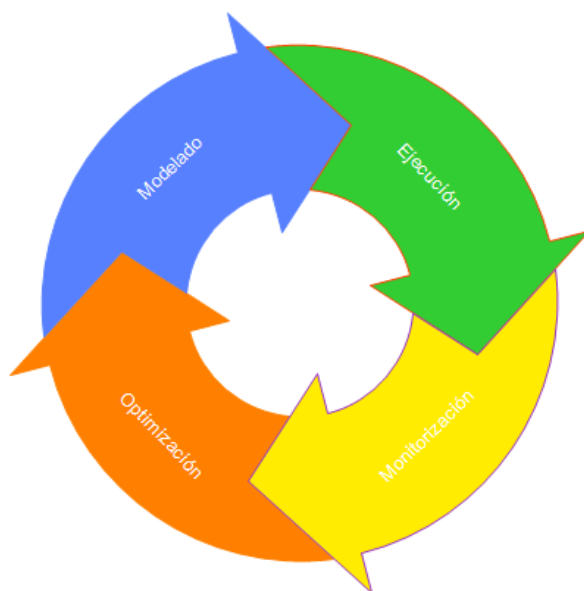


Figura 4.1: *Actividades de los Sistemas BPM.*

Nuestro objetivo en este capítulo será, por una parte, crear una herramienta intuitiva y amigable que permita diseñar de manera gráfica los procesos participativos y, por otra, utilizando el esquema XML del Capítulo 3, guardar dichos procesos en un formato XML que permita su intercambio entre diferentes aplicaciones. En la Sección 4.2 describiremos la solución BPM y algunas herramientas y estándares que nos permiten diseñar procesos de negocio. A continuación, en la Sección 4.3, realizaremos una comparativa de las notaciones presentadas en la Sección 4.2. En la Sección 4.4, presentaremos la extensión del estándar BPMN que se ha realizado para crear el lenguaje gráfico que represente un proceso participativo. Por último, presentamos algunas conclusiones.

4.2. Estándares y herramientas BPM

En torno a BPM se han definido numerosos estándares como son BPMN, BPEL, XPDL, WSCDL, jBPM, jPDL, entre muchos otros. Son pocos los que se han impuesto, mientras

que la gran mayoría, como BPML o WSFL, han quedado en desuso. A continuación, describiremos las características más relevantes de algunos de estos estándares.

4.2.1. BPMN

BPMN (*Business Process Modelling Notation*), véase BPMN Specification (2006), es un estándar de la *Business Process Management Initiative*, que utiliza una notación gráfica para diseñar flujos de procesos de negocio basada en redes de Petri (1962). Estas son extremadamente útiles para la especificación y simulación de procesos complejos que incluyan paralelismo y concurrencia. Este estándar tiene en consideración que la eficiencia, la optimización y la gestión de los procesos son la clave del éxito para las empresas y confieren una ventaja competitiva, véase Freund et al. (2011). Por otra parte, White (2009), definen el estándar BPMN como un lenguaje gráfico común que permite que las partes involucradas puedan comunicar los procesos de forma clara, completa y eficiente, facilitando la comunicación y comprensión de los procesos de negocio y las Tecnologías de la Información.

Los objetivos del estándar BPMN son proporcionar una notación fácilmente comprensible por todos los usuarios de un proceso, véase OMG (2006), y tender puentes entre el diseño del proceso de negocio y la implementación del mismo, White (2004), para que los lenguajes basados en XML tengan una notación gráfica para el diseño de los procesos.

BPMN define diagramas de proceso de negocio, llamados BPD (*Business Process Diagram*), basados en la técnica de diagramas de flujo, véase Fitter y Green (1979), para crear modelos gráficos, en los que el control fluye explícitamente de una a otra figura que se encuentran interconectadas, de las operaciones de procesos de negocio. En la Figura 4.2 puede verse un ejemplo de BPD para, dependiendo del número de votos en una votación, ocultar los resultados hasta que se supere cierto umbral:

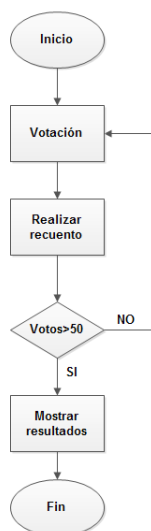




Figura 4.2: *Ejemplo de diagrama de flujo.*

Los elementos que componen los BPD pueden clasificarse en cinco categorías, véase BPMN (2011), aunque las cuatro más importantes son los objetos de flujo, los conectores, las calles y los artefactos.


Los objetos de flujo son un número reducido de elementos con el objetivo de que los modeladores de procesos no tengan que aprender y memorizar gran cantidad de iconos, véase White (2004). En el Cuadro 4.1 presentamos los tres más importantes para nuestros propósitos.

Tipo	Descripción	Imagen
Suceso (Event)	Algo que ocurre o puede ocurrir durante el transcurso de un proceso y afecta su flujo. Pueden ser de dos tipos: Inicio y Finalización	

Tipo	Descripción	Imagen
Actividad (Activity)	Representan el trabajo realizado dentro de una organización. Consumen recursos.	
Pasarela (Gateway)	Se utilizan para controlar la divergencia y convergencia del flujo, permitiendo la exclusión de tareas o que se ejecuten en paralelo.	


Cuadro 4.1: *Objetos de flujo.*

Los conectores se encargan de unir los objetos de flujo para crear la estructura del proceso. Se muestra en el Cuadro 4.2.

Tipo	Descripción	Imagen
Flujo de secuencia	Representan el control del flujo y la secuencia de las actividades y eventos.	

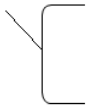
Cuadro 4.2: *Conectores.*

Las swimlanes, o calles, aportan un mecanismo para organizar actividades en categorías separadas visualmente con el fin de ilustrar diferentes capacidades funcionales o responsabilidades. En el Cuadro 4.3 puede verse uno de los tipos de objeto swimlanes que soporta BPD.

Tipo	Descripción	Imagen
Pool	Actúa como contenedor de un proceso. El nombre del pool puede ser el del proceso o el del participante.	

Cuadro 4.3: *Swimlanes*.

Los artefactos se utilizan para añadir información al proceso. En el Cuadro 4.4 se muestra uno de los artefactos estándar, aunque cada herramienta de modelado es libre de añadir tantos artefactos como sea necesario.

Tipo	Descripción	Imagen
Anotación (Annotation)	Son cuadros de texto utilizados para proporcionar información adicional sobre el proceso.	

Cuadro 4.4: *Artefactos*.

La notación BPM ha tenido una gran aceptación. Algunas de las herramientas más importantes que implementan la especificación de BPMN son ILOG: JViews, Intalio: n Designer, Software AG: Enterprise Business Process Manager y Yaoqiang.

Un ejemplo sencillo de proceso BPM, utilizando su notación, puede verse en la Figura 4.3. En una votación electrónica existen diversos actores y es necesario realizar varias actividades en un orden definido. Primero, se debe preparar la lista de alternativas que forman parte de la votación para que, a continuación, los participantes puedan emitir su voto acerca de su alternativa o alternativas preferidas. Una vez que ha finalizado el plazo de recepción de votos se preparan los resultados y, de forma paralela, se publican en la web y envían por correo electrónico a todos los participantes en la votación. Por último, los participantes reciben los resultados en su correo electrónico. Por ejemplo, no tendría sentido que los par-

participantes emitieran su voto antes de preparar la lista de alternativas o preparar los resultados antes de que los participantes depositen su voto.

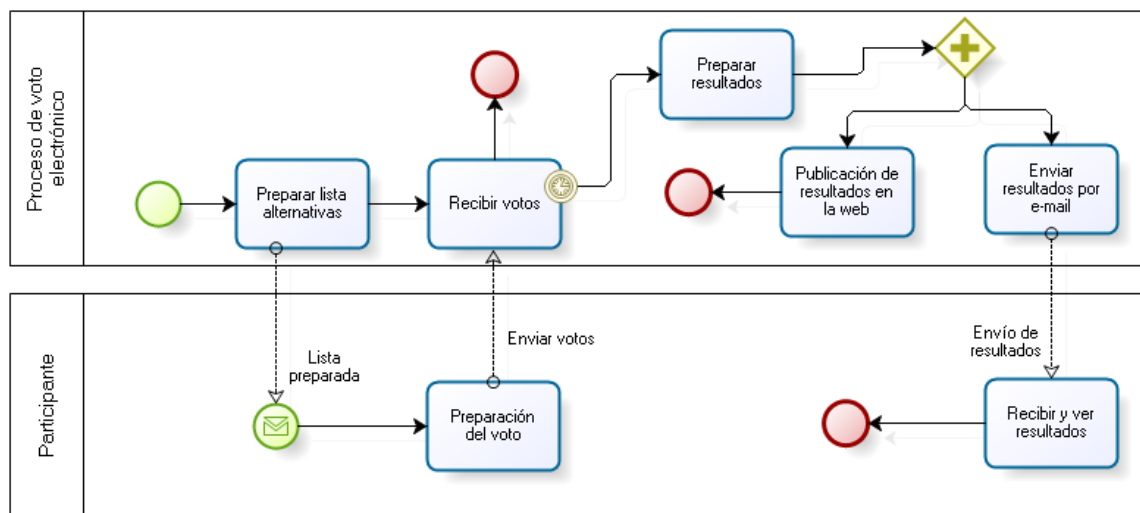


Figura 4.3: Ejemplo básico de proceso utilizando BPMN.

4.2.2. BPEL

BPEL (*Business Process Execution Language*), véase OASIS (2006), es un lenguaje de ejecución de procesos estandarizado por OASIS. Se centra en el proceso, con el inconveniente de no tener notación gráfica, aunque existen varias alternativas, como puede ser BPEL4People (2005), que suple tal carencia. Los lenguajes de ejecución de procesos permiten la combinación de servicios de dos tipos:

- **Orquestación:** Un conjunto de servicios es coordinado por otro que controla las interacciones, de modo que cada uno de los servicios coordinados no perciben que forman parte de un sistema.
- **Coreografía:** Define las colaboraciones entre distintos servicios. Cada servicio del sistema colabora con el resto y conoce con quién puede interactuar, cuándo y cómo

realizarlo.

BPEL es un lenguaje de orquestación que se encarga de controlar qué proceso ejecutar (servicio Web) y en qué momento. Es un lenguaje ejecutable con variables y operaciones. Las operaciones permiten enviar y recibir mensajes SOAP (Simple Object Access Protocol), un protocolo estándar utilizado para definir cómo dos objetos en diferentes procesos pueden comunicarse mediante intercambio de datos XML. Tiene un gran soporte para XML y transformaciones XML. Además, permite, de manera fácil, invocar múltiples servicios web al mismo tiempo y sincronizar los resultados. Como todo lenguaje basado en XML, consta de etiquetas, cada una con su semántica. Un documento BPEL tiene una etiqueta `<process>` que es el elemento raíz del documento. En el interior de dicha etiqueta se encuentra la descripción completa del proceso. Existen muchas otras etiquetas que modifican la forma en que la máquina ejecutora de BPEL interpreta el código. Algunas de ellas son `<partnerLinks>`, `<variables>` y `<sequence>`, véase White (2009).

Un ejemplo de documento BPEL puede verse en el Código 4.1. En él se muestra un proceso llamado *SendVote* definido en el atributo *name*. A continuación, *partnerLinks* se utiliza para definir los roles de los participantes del proceso donde, en este caso, existe un participante llamado *caller* que representa el papel de votante. La etiqueta *variables* se utiliza para almacenar los datos o estados que aparecen en el proceso. Para el caso que estamos tratando, tenemos dos posibles estados: enviar el voto (*request*) y recibir el mensaje de confirmación (*response*). Por último, en *sequence* se representan las actividades, el flujo del proceso de negocio donde, por un lado, tenemos la actividad de recibir los votos (*ReceiveVote*) enviados por los participantes con rol *caller* y, por otro, la actividad encargada de enviar el mensaje de confirmación al usuario (*SendConfirmation*) que envió el voto. En la etiqueta *assign* se genera el mensaje que se enviará de una actividad a otra.

```
1 <process name="SendVote">
2   <partnerLinks>
3     <partnerLink name="caller"
4       partnerLinkType="tns:Voter-Caller"
5       myRole="Voter" />
6   </partnerLinks>
7   <variables>
8     <variable name="request"
9       messageType="tns:voteListMessage" />
10    <variable name="response"
11      messageType="tns:sucessfulMessage" />
12  </variables>
13  <sequence name="MainSeq">
14    <receive name="ReceiveVote" operation="send"
15      partnerLink="caller" portType="tns:Voter"
16      variable="request" createInstance="yes" />
17    <assign name="ComposeMessage">
18      <copy>
19        <from expression="concat('Your vote has
20          been sent successfully!!.')" />
21        <to variable="response" part="voting" />
22      </copy>
23    </assign>
24    <reply name="SendConfirmation" operation="send"
25      partnerLink="caller" portType="tns:Voter"
26      variable="response" />
27  </sequence>
28 </process>
```

Código 4.1: Ejemplo de documento BPEL para confirmar voto enviado.

4.2.3. XPDL

XPDL es un lenguaje de definición de flujos de trabajo en procesos. Proporciona un formato de fichero XML que permite el intercambio de información entre distintas aplicaciones. En tal fichero se almacena información específica ejecutable como son los roles de los participantes, la descripción de las actividades o las llamadas a servicios Web. XPDL permite

almacenar e intercambiar el diagrama de un proceso y que un diagrama creado con una herramienta de diseño de procesos sea leído por otra, así como que la imagen resultante sea lo más parecida posible, véase Swenson (2006). Existen múltiples herramientas para trabajar con XPDL, véase Workflow Management Coalition (2011), como Amazonas Workflow, Bonita, IBM FileNet Business Process Manager, ORACLE 9i Warehouse Builder o TIBCO iProcess Suite.

XPDL está fuertemente ligado a BPMN tratando de representar los elementos principales de BPMN a través de la especificación de XPDL. A pesar de ello, existen algunos elementos gráficos que no pueden ser representados en XPDL. Igualmente, muchos atributos y elementos de XPDL no tienen su correspondiente elemento gráfico en BPMN. La principal desventaja de XPDL es que no está pensado para ser ejecutado, con lo que es necesario añadir información específica no estándar al lenguaje.

En el Código 4.2 podemos ver parte del documento XPDL generado a partir del proceso de negocio de la Figura 4.3.

```
1 <Package>
2 ...
3 <WorkflowProcesses>
4 <WorkflowProcess Id="d65fb468-b219-4bdf-a25b-47c30cac4412"
5   Name="Proceso de voto electrónico">
6 ...
7 <Activities>
8 <Activity Id="5d50c3ad-5a05-4433-bad2-d963a540601e"
9   Name="Inicio">
10 <Event>
11 <StartEvent Trigger="None" />
12 </Event>
13 </Activity>
14 <Activity Id="c0d21f06-ba8e-4c15-b800-c0f69d6f05a0"
15   Name="Preparar lista alternativas">
16 </Activity>
17 ...
18 </Activities>
19 <Transitions>
20 <Transition Id="8fae9704-604e-410a-bf8e-718a0ef28c27"
```


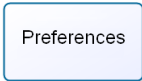


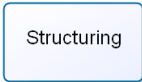
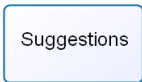


```
21     From="5d50c3ad-5a05-4433-bad2-d963a540601e"  
22     To="c0d21f06-ba8e-4c15-b800-c0f69d6f05a0">  
23     </Transition>  
24     ...  
25     </Transitions>  
26     </WorkflowProcess>  
27     </WorkflowProcesses>  
28     ...  
29 </Package>
```

Código 4.2: Ejemplo XPDL del proceso de la Figura 4.3.

4.3. eBPBMN: Diseño de procesos participativos

Como hemos visto en los capítulos anteriores, un proceso participativo consta de una serie de fases que combinadas y ejecutadas en determinado orden y, cumpliendo una serie de restricciones, véase Gómez et al. (2013), puede dar lugar a múltiples formas de participación de un grupo de personas en la toma de decisiones en un problema concreto. En cada una de las fases, no actúan los mismos participantes necesariamente, lo que hace que encontremos diferentes roles de participación. Por lo tanto, los procesos participativos pueden contemplarse como procesos de negocio con un ciclo de vida que consta de diferentes fases de ejecución y donde interactúan distintos participantes.

Nuestro objetivo aquí es crear un método para diseñar los procesos participativos de forma sencilla e intuitiva en la que el usuario únicamente tenga que arrastrar y soltar las etapas que van a formar parte del proceso a través de una interfaz gráfica. Además, sin necesidad de realizar ningún tipo de programación, debe poder indicar el orden en el que se ejecutarán y asignar los roles de los participantes a cada una de las fases. Para ello, basándonos en la notación BPM, se ha creado un conjunto de elementos asociados con el objeto de flujo “Actividades” del estándar BPMN que representan cada una de las tareas comunes identificadas. Así, tendremos quince actividades asociadas a las tareas explicadas en la Sección 2.2. Algunas de las más importantes se representan en el Cuadro 4.5:

Representación	Descripción
	Fase de debate en el proceso participativo en la que los participantes intercambian ideas.
	Fase de comunicación de preferencias en la que se utilizan métodos para que los participantes revelen sus prioridades.
	Fase de muestreo, en la que se elige una serie de participantes para representar al resto.
	Fase de votación en la que los participantes votarán por sus alternativas preferidas.
	Fase de estructuración de problemas en la que se organiza el problema.
	Fase de sugerencias, en la que los participantes envían sus propuestas para ofrecer soluciones al problema.
	Fase de negociación, en la que los participantes negocian para intentar llegar a un acuerdo.
	Fase de arbitraje en la que el árbitro toma la decisión final una vez presentadas las propuestas.

Cuadro 4.5: Elementos para el diseño de procesos participativos.

El resto de componentes necesarios para la creación de procesos serán los mismos que en la notación BPM, aunque sólo utilizaremos el flujo de secuencia de los conectores; la pasarela paralela, que permitirá ejecutar dos tareas al mismo tiempo; la pasarela exclusiva, que permite que si se ejecuta una tarea no se ejecute otra y; finalmente, los eventos de inicio y de fin. Así, no es necesario crear nuevos elementos, ya que los utilizados en BPMN contienen suficiente funcionalidad para nuestros propósitos.

Un ejemplo sencillo de proceso diseñado podría ser el representado en la Figura 4.4, donde el proceso comienza con una estructuración del problema, en la que se añaden los criterios, las alternativas y las posibles restricciones; continúa con dos tareas paralelas, un debate, donde se realiza un intercambio de información entre los participantes para presentar diferentes ideas y posibles soluciones al problema y una negociación para acordar cuáles son las alternativas disponibles en la última fase; y finalmente, concluye con una votación en la que los participantes eligen sus alternativas favoritas.

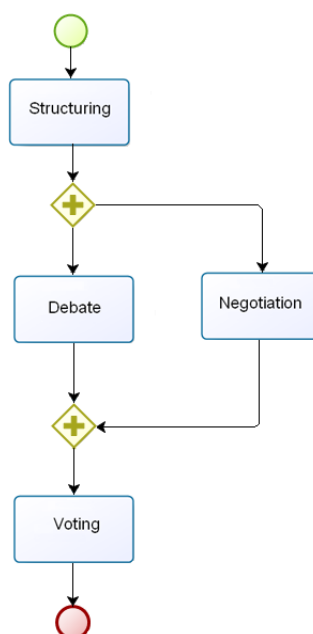


Figura 4.4: *Ejemplo de proceso diseñado.*

Una vez que el proceso ha sido esquematizado, es necesario indicar qué roles pueden participar en cada fase. Esto es posible a través de las propiedades de la actividad. En la Figura 4.5, se puede ver de qué forma se asignan los roles de los participantes en cada tarea. También resulta necesario indicar cuáles son las fechas de inicio y de fin de la fase que se añade al proceso. En el caso de la votación de la Figura 4.5, se puede ver que la votación comienza el día dos de Julio de 2012 y finaliza el día siete del mismo mes y año, siendo “Participantes” el rol de los usuarios que pueden participar en la votación. En este caso, los usuarios que desempeñen el papel de “Técnicos” o “Autoridades” no podrán intervenir en la tarea.

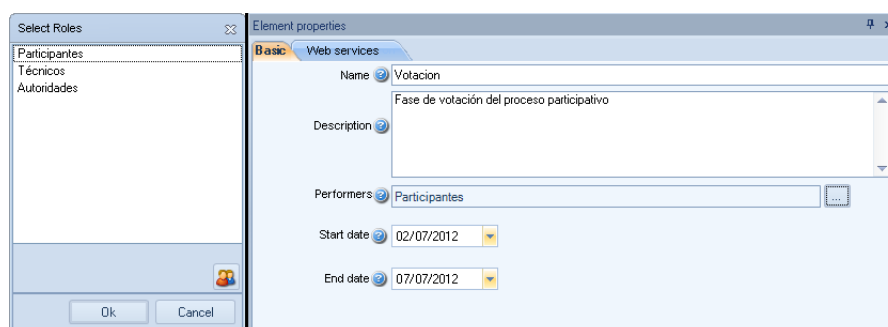


Figura 4.5: Asignación de roles a las tareas.

Del mismo modo, y, a través de las propiedades de la actividad, es posible indicar la ubicación del servicio web que ofrece solución a la tarea en cuestión. Para ello, a la hora de elegir un servicio web para una actividad determinada, la herramienta de diseño de procesos se conectará al servidor UDDI, véase Capítulo 2, donde están clasificados los servicios disponibles a la actividad correspondiente. Por ejemplo, para una tarea de “Votación” en la que tuviéramos tres posibles métodos de resolución, el diseñador de procesos mostraría el enlace web donde está alojado el servicio de cada uno de los métodos y el actor que diseña el proceso se encarga de elegir el método más adecuado para el mismo. En la Figura 4.6 se pueden ver los cinco servicios web disponibles a través de las propiedades de la tarea “Votación”.

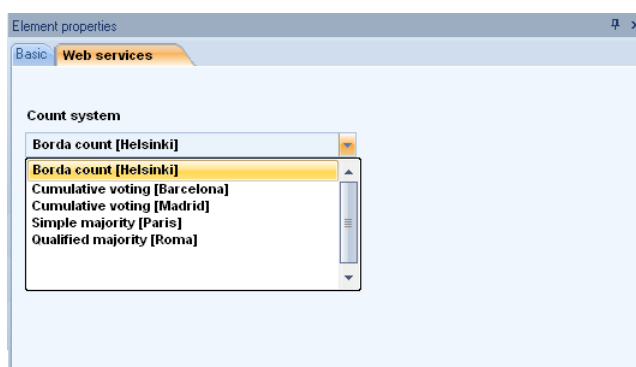


Figura 4.6: Elección de método de votación para resolver una tarea.

Una vez presentados los elementos de los que dispondremos, explicaremos de qué modo puede realizar un diseño un usuario con permisos para crear procesos. Como dijimos anteriormente, un proceso participativo puede considerarse como una combinación de tareas participativas. Al conjunto inicial de procesos lo denominaremos P y se define mediante:

$$P = \{m : m = (t_1, t_2, t_3, \dots, t_n), n \in N; t_i \in T\}$$

donde n es el número de tareas que se incluyen en el proceso m , t_i es la tarea que se realiza en la posición i -ésima y T es el conjunto de tareas participativas presentadas en el Capítulo 2. Por tanto, $T = \{\text{muestreo de participantes, uso de cuestionarios, elección de representantes, preparación de documentos, distribución de la información, intercambio de información, estructuración del problema, generación de alternativas, comunicación de preferencias, exploración individual del problema, debate, negociación, arbitraje, votación y explicación final}\}$.

Con la finalidad de facilitar y simplificar el diseño de procesos, el usuario tiene la posibilidad de que el sistema le ofrezca un diseño inicial para el proceso que quiere implementar. A través de un sencillo cuestionario, éste informa sobre los datos principales del proceso como son qué tareas deben formar parte del proceso, la duración máxima del mismo, tareas

inicial y final, el número de fases, el ámbito del proceso (comunitario, local, autonómico, estatal, . . .), la finalidad del proceso, es decir, qué datos queremos recuperar o proporcionar. Una vez que el cuestionario ha sido rellenado, se mostraría el proceso que mejor valoración obtiene, utilizando el algoritmo implementado por (Lavin et al., 2013), en el que hemos tenido en cuenta, experiencias anteriores para penalizar, en mayor o menor medida, una serie de factores como puede ser la duración de las tareas que produce una disminución en la participación, y restricciones que existen en el orden de aparición de las tareas. Las restricciones deben ser satisfechas para asegurar un correcto funcionamiento del proceso participativo. A continuación, se detallan algunas de las posibles restricciones:

- I) *Número máximo de tareas*. Este número depende del proceso que se quiera diseñar. Esta limitación evita efectos no deseados, como puede ser el cansancio de los ciudadanos que perjudica la participación.
- II) *Número mínimo de tareas*. Cualquier proceso participativo debe contar con un número mínimo de tareas. Simplificando mucho podemos conseguir procesos con únicamente dos tareas.
- III) *Repetición de tareas*. Con el fin de que el proceso no se alargue en el tiempo, hay tareas que deben ejecutarse una única vez, porque repetirlas puede ser demasiado costoso, retrasar el final del proceso o simplemente resulta poco útil repetirlas una vez que ya han sido ejecutadas.
- IV) *Posición en el proceso*. Debido a su funcionalidad, existen tareas que sólo pueden utilizarse en una determinada posición dentro del proceso. Por ejemplo, no se deben utilizar al inicio del proceso tareas que requieran la ejecución de otra tarea previamente.
- V) *Compatibilidad de tareas*. Existen tareas que, de ser ejecutadas, invalidan la presencia de otras. Por ejemplo, si existe una votación, no debe aparecer una tarea de arbitraje y viceversa.

VI) *Secuencia de tareas*. Algunas tareas deben ejecutarse previamente a otras, puesto que las primeras permiten la existencia de las segundas. Por ejemplo, el debate o el uso de cuestionarios pueden aparecer simultáneamente, pero siempre detrás de una estructuración del problema.

Explicadas las posibles restricciones que debe cumplir cualquier proceso participativo, pasamos a citar los objetivos que debe cumplir el mismo en una situación dada. Generalmente los objetivos del proceso serán definidos por el dueño del problema pero en la mayoría de los casos lo que se busca es mejorar el flujo de información entre ciudadanos y autoridades (objetivo 1), aumentar la transparencia de las decisiones (objetivo 2), implicar a los ciudadanos en la ejecución del proceso (objetivo 3), representar al conjunto de los participantes incluyendo a minorías o asociaciones (objetivo 4) y, por último, minimizar la duración del mismo evitando cansancio entre los participantes (objetivo 5). Con los objetivos ya definidos, se construye una función de valor basándose en un modelo de función aditiva, véase Keeney y Raiffa (1993), Von Winterfeldt y Edwards (1986) y Clemen y Reilly (2001), sin gran pérdida de generalidad.

Así, dado un proceso $m = (t_1, t_2, t_3, \dots, t_n)$, se le asignan valores a los objetivos y se evalúa mediante la función de valor:

$$v(m) = \sum_{i=1}^n \rho_i \left[\sum_{j=1}^4 w_j v_j(t_i) \right] + w_5 v_5(d(m))$$

donde w_j son los pesos de los criterios con $w_j \geq 0$, $\sum w_i = 1$, $v_j(t_i)$ es la evaluación que se hace de la tarea t_i en el objetivo j , $v_5(d(m))$ es la duración normalizada del proceso, y $\rho_i = \rho^{k-1}$, si t_i es la k -ésima vez que aparece en el proceso tal tipo de tarea. Por lo tanto, lo que se busca es maximizar el valor de $v(m)$ para obtener el instrumento mejor valorado con las tareas seleccionadas.

$$\text{máx } v(m) \text{ s.a. } m \in R$$

donde R, representa los posibles procesos que se pueden construir una vez aplicadas las restricciones.

De este modo, nuestro objetivo es facilitar el diseño de procesos participativos y conseguir que cualquier persona, sin necesidad de ser experta en el diseño de este tipo de procesos, pueda configurarlo para crear el que mejor se adapte a sus necesidades. En las Figuras 4.7 y 4.8, pueden verse dos ejemplos más visuales de cómo representar los procesos diseñados para que, cualquier persona, independientemente de sus conocimientos sobre procesos participativos, pueda comprender cuáles son las etapas del proceso y qué usuarios podrán participar en cada una de ellas. Cuando todas las modificaciones han sido realizadas y, antes de que el usuario guarde el proceso para su posterior modificación, el sistema comprueba, utilizando el algoritmo citado anteriormente, que las tareas cumplan todas las restricciones para, en caso contrario, informar al usuario de qué tareas debe modificar antes de poder guardar el proceso.

PETICIONES CIUDADANAS

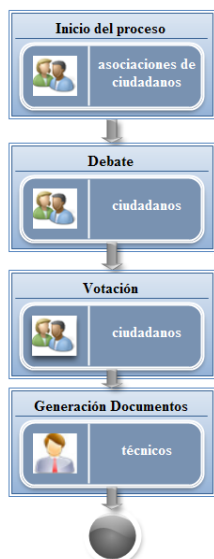


Figura 4.7: *Peticiones ciudadanas*

FOROS DE BARRIO


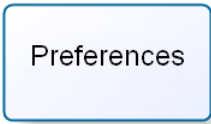



Figura 4.8: *Foros de barrio*

4.4. Conexión con XML

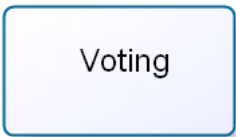
Tras realizar el diseño gráfico del proceso, es necesario disponer de un mecanismo de intercambio de información entre las diferentes aplicaciones. Para ello, basándonos en el estándar XPDL, nos hemos propuesto relacionar dicho diseño con el esquema XML del módulo controlador de la arquitectura presentada en el Capítulo 2 de esta Tesis. Podemos encontrar una correspondencia directa entre los elementos gráficos utilizados para construir el proceso y los elementos del esquema XML. Además, una vez definido el proceso, éste puede exportarse a un documento XML utilizando el esquema XML del controlador mostrado en el Capítulo 3.

Cada una de las actividades que pueden formar parte del proceso participativo tiene su elemento específico en el esquema XML. En el Cuadro 4.6, podemos ver cuál es exactamente la correspondencia de algunas de las actividades más importantes:

Representación gráfica	Representación XML
	<pre><controller> <phase> <debate> <start></start> <finish></finish> </debate> </phase> </controller></pre>
	<pre><controller> <phase> <preferences> <start></start> <finish></finish> </preferences> </phase> </controller></pre>
	<pre><controller> <phase> <voting> <start></start> <finish></finish> </voting> </phase> </controller></pre>

Cuadro 4.6: Correspondencia de algunas actividades con el esquema XML.

Además, como hemos visto, algunas de estas actividades pueden tener diferentes métodos de resolución, por lo que aparecen nuevas etiquetas para conocer cuál es dicho método. En el Cuadro 4.7 podemos ver una votación cuyo método de resolución es mayoría simple.

Representación gráfica	Representación XML
	<pre><voting> <start></start> <finish></finish> <count_system> simple majority </count_system> <options_by_vote /> <points_by_option /> </voting></pre>

Cuadro 4.7: Correspondencia votación (mayoría simple) con el esquema XML.

Un ejemplo de documento XML creado a partir del proceso participativo de la Figura 4.4 sería el mostrado en el Código 4.3:

```

1 <problem>
2   <controller>
3     <phase>
4       <structuring url="http://app1/WS/strucService.asmx">
5         <start>2012/07/02 00:00</start>
6         <finish>2012/07/04 00:00</finish>
7         <roles>
8           <rol>Técnico</rol>
9         </roles>
10        </structuring>
11       <debate url="http://app2/WS/debateService.asmx">
12         <start>2012/07/05 00:00</start>
13         <finish>2012/07/07 00:00</finish>
14         <roles>
15           <rol>Participante</rol>
16         </roles>
17        </debate>
18       <negotiation url="http://app3/WS/negService.asmx">
19         <start>2012/07/05 00:00</start>
20         <finish>2012/07/07 00:00</finish>
21         <roles>
22           <rol>Participante</rol>
```

```
23     </roles>
24 </negotiation>
25 <voting url="http://app4/WS/votService.asmx">
26     <start>2012/07/08 00:00</start>
27     <finish>2012/07/10 00:00</finish>
28     <roles>
29         <rol>Participante</rol>
30     </roles>
31     <count_system>Simple majority</count_system>
32     <options_by_vote min="0" max="1" />
33     <points_by_option min="0" max="1" />
34 </voting>
35 </phase>
36 </controller>
37 </problem>
```

Código 4.3: Ejemplo de documento XML a partir del proceso de la Figura 4.4.

4.5. Conclusiones

En este capítulo, hemos visto que los sistemas BPM son herramientas que mejoran la eficiencia a través de la gestión de los procesos de negocio, lo que hace, que se encuentren en un claro auge de expansión. Constan de cuatro etapas principales: Modelado, Ejecución, Monitorización y Optimización. Además, permiten asignar usuarios a las diferentes tareas del proceso. Por lo tanto, podemos considerar un proceso participativo como un proceso de negocio en que se puede implementar utilizando BPM, ya que consta de conjunto de actividades en las que participan diferentes actores que se ejecutan en un orden determinado, para dar solución a un problema concreto.

Existen múltiples estándares y herramientas para facilitar el diseño de procesos de negocio. Por tanto, tomando como referencia alguno de ellos, proponemos una notación gráfica para que el diseño de procesos participativos se convierta en una tarea sencilla para un usuario sin experiencia. Nos hemos basado en el estándar BPMN para crear una notación gráfica que facilite el diseño de procesos participativos, de modo que, el usuario encargado

de diseñarlos, pueda arrastrar y soltar las tareas en el editor, indicar cuál es el orden de ejecución, los actores que participan e indicar la localización del servicio web que da solución a cada tarea, así como las fechas de inicio y fin de las mismas.

Sin embargo, BPMN es una herramienta que facilita el trabajo a los diseñadores y desarrolladores de procesos pero no es posible su ejecución en ningún sistema informático. Para ello, se utiliza el estándar XPDL, que utiliza lenguaje XML y permite almacenar e intercambiar diagramas de procesos para que puedan ser ejecutados en un motor de flujos de trabajo. En nuestro caso, dicho motor sería el módulo controlador puesto que se encarga de comprobar el estado del proceso participativo.

Por lo tanto, se han cumplido las cuatro actividades principales de un sistema BPM (Modelado, Ejecución, Monitorización y Optimización). Podemos diseñar procesos participativos utilizando una representación gráfica. Exportar el diseño para ejecutarlo en un motor de flujos de trabajo como puede ser el módulo controlador. Se permite la monitorización tales ejecuciones para comprobar su correcto funcionamiento a través de la aplicación web y, el proceso diseñado por el usuario se optimiza utilizando el algoritmo explicado en Gómez et al. (2013).

Capítulo 5

Fundamentos para un módulo de explicación basado en textos en una arquitectura de e-participación

5.1. Introducción

Un reto pendiente en el desarrollo de sistemas de e-participación es facilitar la explicación de la metodología y los resultados a los ciudadanos, independientemente de su formación académica, profesional o personal. Cuando se crea un proceso participativo es tan importante tener en cuenta cuál es la finalidad del mismo, como conocer qué personas van a participar en el mismo. Esto último, no sólo afecta a cómo se deben elegir los criterios de evaluación, sino también, a la generación de informes con los resultados obtenidos. Además, el resultado de un proceso de decisión ha de ser comunicado a todas las partes involucradas de una forma transparente y comprensible. De lo contrario, lo más probable es que el resultado no sea aceptado por los participantes (Geldermann, 2010).

La explicación de resultados se ha considerado un punto crucial para que expertos y usuarios puedan entender cómo y por qué un sistema produjo un resultado y, por lo tanto, puedan evaluar si el resultado es aceptable o no (McKeown, 1983). Podemos decir que la explicación consiste en exponer algo de modo que sea entendible por el receptor (Lacave y Díez, 2002). Por ello, consideramos que resulta imprescindible un módulo de explicación que permita transmitir de forma comprensible a cualquier individuo las decisiones tomadas. Este módulo, se encontraría integrado en la arquitectura presentada en el Capítulo 2 en la que se explica con más detalle su funcionalidad. En este capítulo nos centramos en la extracción de explicaciones basada en los textos que, por ejemplo, se transmiten en los foros del sistema.

El capítulo se organiza de la siguiente manera. En la sección 2, hacemos una revisión sobre minería de textos y sistemas de explicación. En la sección 3, presentaremos algunos de los métodos utilizados y los objetivos planteados con dicho análisis. La sección 4 introduce la metodología empleada para llevar a cabo nuestra evaluación de diferentes técnicas de clasificación de mensajes y minería de opiniones. En la sección 5 se presentan los resultados de un experimento real, junto con una discusión sobre las diferentes implicaciones que pueden inferirse a partir de las métricas de rendimiento. Por último, en la sección 6, resumimos las conclusiones obtenidas.

5.2. Revisión

En la sociedad actual, debido al gran auge que en los últimos años está experimentando lo que recibió el nombre de Web 2.0, véase O'Reilly (2007), que incluye las redes sociales, micro-blogs, blogs personales o redes profesionales, se ha producido un incremento exponencial de la información subjetiva disponible en Internet. Así, cualquier persona o institución puede recuperar información sustancial a partir del análisis de los datos disponibles en forma de mensajes y comentarios recibidos mediante los canales de redes sociales, blogs o wikis, véase Mergel et al. (2009). Estos datos subjetivos tienen un gran potencial y pueden

explotarse para, por ejemplo, conocer la opinión sobre personalidades públicas, elegir la propaganda idónea según las preferencias u opiniones de la gente o, encontrar el producto mejor valorado por los usuarios, véase Liu (2007). Para ello, ha surgido el análisis de sentimientos, también conocido como minería de opiniones, que es una tarea del procesamiento del lenguaje natural que identifica opiniones relacionadas con un objeto y trata de analizar las opiniones, sentimientos, evaluaciones o emociones en relación con cualquier tema de interés mencionado en los datos recopilados. Los temas pueden variar desde la opinión sobre una marca o producto, en caso de querer conocer la satisfacción de los clientes, a temas más generales como noticias de última hora, cuestiones políticas o asuntos económicos, véase Liu (2012).

La minería de textos es una aplicación del procesamiento de textos que pretende facilitar la identificación y extracción de nuevo conocimiento a partir de colecciones de documentos (Baeza-Yates y Ribeiro-Neto, 1999), trabajando, por lo tanto, con información no estructurada. Inicialmente, la minería de textos debería facilitar el análisis de estos documentos que, a priori, resultarían inmanejables debido a su tamaño, de modo que se puedan obtener relaciones entre los documentos y extraer conclusiones sobre los mismos. En el caso de documentos escritos en diferentes idiomas, resulta interesante la utilización de diferentes recursos lingüísticos (glosarios, ontologías, ...) para representar los documentos mediante rasgos independientes del idioma (Steinberger et al., 2005).

Los problemas básicos que pueden abordarse con técnicas de minería de texto son la recuperación de información relevante, la categorización de documentos y la agrupación de documentos relacionados con un tema. Para conseguir esto, la minería textual adopta una serie de técnicas procedentes de la recuperación de información y de la lingüística computacional como son:

- El pre-procesamiento de documentos, que se compone de la extracción de términos, la eliminación de palabras vacías y la normalización de los términos mediante la técnica de “stemming”, que consiste en extraer la raíz de las palabras.

- La representación de los documentos mediante un modelo vectorial, con el que un documento se representa mediante un vector, siendo cada término del documento un componente del mismo.
- La categorización automática, que se utiliza para clasificar documentos en alguna de las categorías preestablecidas.

Por tanto, la minería de textos tiene como objetivo intermedio procesar y presentar la información disponible en un formato que facilite su comprensión y análisis. En este punto es donde se entra en relación con los sistemas de explicación, en particular en sistemas de participación, como nos interesa en esta Tesis.

Originalmente, los sistemas de explicación surgen en el contexto de los sistemas expertos y la inteligencia artificial, incorporando técnicas de lenguaje natural. Los sistemas de explicación deben ser capaces de describir claramente el dominio del problema y los resultados obtenidos. Se ha demostrado que las explicaciones adecuadas contribuyen positivamente a mejorar los resultados y la satisfacción de los usuarios (Gregor y Benbasat, 1999). Además, influyen en las percepciones del usuario como la confianza, la confidencialidad y la satisfacción, e incrementan sus niveles de aceptación y aprendizaje (Dhaliwal y Benbasat, 1996). Cuando se contruye un sistema o proceso, hay que tener cuidado con predecir las necesidades de los futuros usuarios y elaborar el material que satisfaga las necesidades de los mismos, puesto que si tales suposiciones no están probadas, pueden no constituir una base fiable para el diseño (Forsythe y Buchanan, 1991). Una falta de atención a la hora de diseñar el proceso sin tener en cuenta las necesidades de los usuarios, está destinada a limitar la funcionalidad del sistema, ya que es probable que no todos los usuarios conozcan los supuestos que se dan por sentados.

Un buen ejemplo son los módulos de explicación para justificar los resultados de los análisis multicriterio que tratan de comunicar los resultados obtenidos mediante un sistema de toma de decisiones multicriterio de modo comprensible. Usualmente, generan dos informes en lenguaje natural, uno para evaluar los resultados y comparar alternativas, y otro,

que explica la sensibilidad al cambio, en por ejemplo, los pesos asignados a los criterios (Geldermann, 2010).

Los sistemas de explicación son especialmente importantes en democracia electrónica, donde muchas personas con diferentes percepciones o expectativas están involucradas en los procesos de toma de decisiones complejas (Ainsworth, 1999)

5.3. Análisis de los mensajes

Durante el estudio de los diferentes instrumentos de participación utilizados globalmente, vimos cómo una de las tareas comunes en muchos de ellos era el debate a través de un foro en el que los participantes intercambiaban información y comunicaban sus preferencias. También describimos el método de negociación POSTING, con el que los usuarios pueden escribir mensajes que apoyen las propuestas enviadas. Por ello, en la arquitectura presentada en el Capítulo 2 se incluye un módulo para implementar un foro que permita añadir dicha tarea de debate a un proceso participativo. Toda esta información puede ser fácilmente recuperada de los documentos XML creados a partir de los esquemas presentados en el Capítulo 3.

Nuestro objetivo en este capítulo es, mediante el análisis y clasificación de los mensajes escritos por los participantes, conocer cuáles son las tendencias de opinión de los mismos, las propuestas con mayor aceptación, las posibles soluciones al problema planteado que satisfagan a la mayoría de los participantes, qué reacciones generaría el hecho de tomar una decisión concreta, etc, con vistas a proporcionar una explicación semiautomática.

En esta sección explicaremos brevemente cómo transformar mensajes textuales en vectores y dos enfoques de aprendizaje adaptados al problema de análisis de sentimientos en el contenido de textos como son k -NN, k vecino más próximo (del inglés k -Nearest Neighbour) y SVM, máquinas de vector soporte (del inglés Support Vector Machines).

5.3.1. Representación vectorial de los mensajes

Los mensajes y términos de, por ejemplo, un foro, pueden representarse conjuntamente a través de la denominada *matriz de términos-documentos* (Baeza-Yates y Ribeiro-Neto, 2011), a la que nos referimos como matriz de términos-mensajes con el fin de que sea más representativa del ejemplo que mostraremos. Consideremos una muestra de n mensajes $C = \{x_1, \dots, x_n\}$ y una muestra de m términos $T = \{t_1, \dots, t_m\}$. En esta matriz, cada columna representa un mensaje y cada fila corresponde a un término. Entonces, la celda (i, j) corresponde a la frecuencia ij de aparición del término t_i en el mensaje x_j . En el Cuadro 5.1 podemos ver un ejemplo de una matriz de términos-mensajes con n mensajes y m términos.

Diccionario	Mensaje 1	Mensaje 2	...	Mensaje n
mejor	f_{11}	f_{12}	...	f_{1n}
grande	f_{21}	f_{22}	...	f_{2n}
malo	f_{31}	f_{32}	...	f_{3n}
bueno	f_{41}	f_{42}	...	f_{4n}
...
término- m	f_{m1}	f_{m2}	...	f_{mn}

Cuadro 5.1: Ejemplo de matriz de términos-mensajes.

La lista de términos T forma lo que se denomina el *diccionario* de términos. Como resultado, podemos considerar cada columna como una representación m -dimensional de los correspondientes mensajes en \mathbb{R}^m , permitiéndonos utilizar diferentes métodos de aprendizaje cuyo funcionamiento está basado en la representación vectorial de mensajes.

5.3.2. k -NN

Dado un documento x , su distancia $d(x, x_i)$ a los mensajes en C induce un orden de mensajes representado como $x_{(1)}, \dots, x_{(k)}, \dots, x_{(n)}$, donde $x_{(1)}$ es el documento más cercano a x en la muestra C y $x_{(n)}$ el más lejano. De este modo, la distancia del mensaje x a su k -ésimo vecino más próximo en C se define como $d_k(x, C) = d(x, x_{(k)})$, donde d puede ser cualquier distancia válida entre mensajes. Determinar el valor adecuado de k es un problema abierto. Nosotros seguiremos el enfoque de Silverman (1986), en el que k es proporcional al entero truncado del valor $n^{\frac{4}{m+4}}$. En el caso de información textual, la elección preferida es la distancia inducida por la similaridad del coseno, véase Baeza-Yates y Ribeiro-Neto (2011), que viene dada, para los mensajes x_i y x_j , por:

$$\text{sim}(x_i, x_j) = \frac{\vec{x}_i \bullet \vec{x}_j}{|\vec{x}_i| \times |\vec{x}_j|} = \frac{\sum_{t=1}^m f_{t,i} \times f_{t,j}}{\sqrt{\sum_{t=1}^m f_{t,i}^2} \times \sqrt{\sum_{t=1}^m f_{t,j}^2}}$$

A partir de ella, la distancia entre dos mensajes se calcula como:

$$d(x_i, x_j) = 1 - \text{sim}(x_i, x_j)$$

Supongamos que C_1, \dots, C_l son un conjunto predefinido de categorías que contienen cada una de ellas una muestra de mensajes asignados a cada categoría por un experto. Para clasificar un nuevo mensaje x en una de tales categorías, todo lo que hay que hacer es asignar el mensaje a la categoría más cercana utilizando el método k -NN, donde, hemos seleccionado un valor de k igual a 1 y x se asigna a C_j si C_j es la categoría tal que $d(x, C_j) = \min\{d(x, C_1), \dots, d(x, C_l)\}$.

5.3.3. Máquinas de Vector Soporte

En los últimos años, las SVM se han aplicado a distintos problemas de análisis de datos con resultados satisfactorios. En esta sección, mostramos una descripción de la versión de SVM que realiza la clasificación en dos categorías, definidas como $\{+1, -1\}$, junto con sus características principales. Las SVMs forman parte de los llamados métodos de regularización descritos, por ejemplo, en Moguerza y Muñoz (2006). Estas técnicas permiten la construcción de funciones de clasificación resolviendo el problema de optimización, véase (Tikhonov y Arsenin, 1977):

$$\min_{f \in H_K} \frac{1}{n} \sum_{i=1}^n l(y_i, f(x_i)) + M \|f\|_K^2,$$

donde (x_i, y_i) , $i = 1, \dots, n$, son un conjunto de mensajes con $x_i \in \mathbb{R}^n$, $y_i \in \{-1, +1\}$; H_K es un espacio de Hilbert con Núcleo Reproductivo (RKHS, del inglés Reproducing Kernel Hilbert Space) generado por un núcleo K ; $\|f\|_K$ es la norma de f en H_K ; $l(y_i, f(x_i))$ es una función de pérdida, y $M > 0$ es una constante de penalización. Para los casos de clasificación con SVM, esta función de pérdida se define normalmente mediante

$$L(y_i, f(x_i)) = \max(1 - y_i \times f(x_i), 0)$$

El objetivo de dicha función de pérdida es encontrar una función suave $f^* \in H_K$ que resuelva el problema de optimización descrito anteriormente. Esta función tendrá la forma

$$f^*(x) = \sum_{i=1}^n \alpha_i K(x, x_i) + b^*,$$

donde α_i y b^* son constantes, $K(x, y) = \Phi(x)^T \Phi(y)$ es la función de núcleo que genera H_K y $\Phi : \mathbb{R}^m \rightarrow \mathbb{R}^p$ es la definición de K . De este modo, geoméricamente Φ transforma los datos de lo que se denomina “espacio de entrada” (\mathbb{R}^m) en el “espacio de salida” (\mathbb{R}^p).

$M > 0$ penaliza la no suavidad de las posibles soluciones del problema.

5.4. Metodología

En esta sección presentaremos los detalles de la metodología e implementación que hemos utilizado para extraer opiniones y . Nuestro objetivo es:

- Describir un proceso multifase para la categorización de los mensajes y la extracción de opiniones, combinado con métodos supervisados y no supervisados de aprendizaje.
- Comparar el rendimiento de diferentes procedimientos para el análisis de sentimientos del contenido textual extraído de los comentarios del foro.
- Obtener conclusiones y recomendaciones para esta comparación, que puede utilizarse para mejorar la metodología e implementación del análisis de sentimientos en otros casos similares.

Para analizar los mensajes extraídos, es importante utilizar los métodos y diccionarios apropiados para llevar a cabo el análisis de sentimientos.

5.4.1. Evaluación de los mensajes

Nuestro propósito es comparar diferentes técnicas para la clasificación del contenido y el análisis de sentimientos de los mensajes expuestos en un foro, utilizando una combinación de métodos de aprendizaje supervisados y no supervisados, véase Witten et al. (2011). Para lograr esto, proponemos una clasificación multifase y un análisis de opiniones como el presentado en la Figura 5.1. En primer lugar, utilizamos métodos de aprendizaje supervisado para discriminar los mensajes categorizándolos en las temáticas $1, \dots, N$. Seguidamente,

y utilizando de nuevo los métodos de aprendizaje supervisado, clasificamos los mensajes de cada una de las temáticas como comentario positivo, negativo o neutro. Por último, aplicamos el algoritmo de k -medias (aprendizaje no supervisado) para extraer las sub-temáticas que caracterizan las diferentes tendencias de opinión expresadas en los mensajes de cada temática. El esquema mostrado en la Figura 5.1 es adaptativo, pues permite la actualización de tendencias de opinión con la llegada de nuevos mensajes que llegan al sistema.

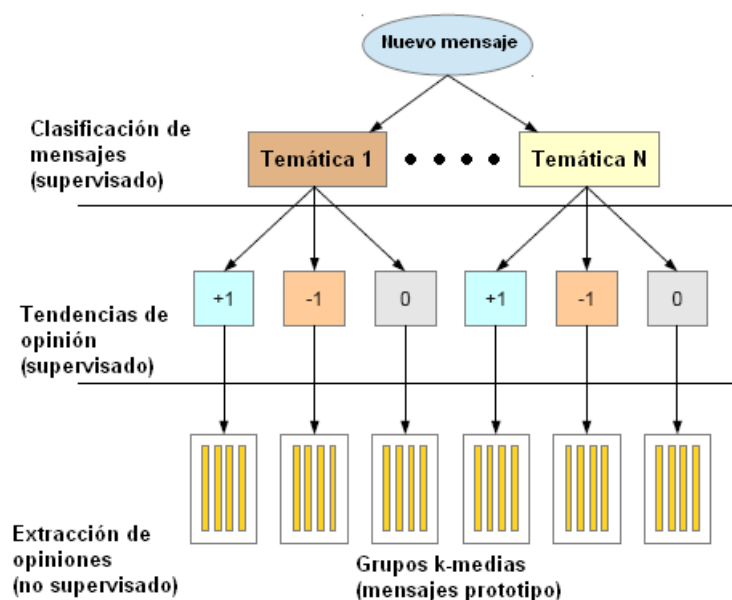


Figura 5.1: Método de clasificación multifase para categorización de mensajes y análisis de sentimientos.

La Figura 5.2 muestra una vista general del procedimiento llevado a cabo en cada fase, lo que implica un método de aprendizaje supervisado. Todos los mensajes se clasifican manualmente de acuerdo al objetivo correspondiente a cada nivel del proceso multifase mostrado en la Figura 5.1. En cada nivel, se lleva a cabo un proceso de votación simple para clasificar los mensajes. En el primer nivel, clasificación de contenido, los mensajes se clasifican agrupándolos en las temáticas 1, ..., N. En caso de que un mensaje mencione varias temáticas, se divide para agrupar cada texto en su temática correspondiente.

En el segundo nivel (tendencias de opinión) podríamos contar con tantas categorías como consideremos necesario. Para el caso que trataremos hemos clasificado los mensajes de cada temática en tres categorías diferentes:

- *Categoría +1* (C_{+1}): Mensajes cuyo contenido expresa una opinión positiva.
- *Categoría 0* (C_0): Mensajes sin un claro signo de opinión, ni positivo ni negativo.
- *Categoría -1* (C_{-1}): Mensaje cuyo contenido expresa una opinión negativa.

Estas clasificaciones, realizadas manualmente, se utilizarán como entrada a los métodos de aprendizaje supervisado aplicados en este estudio. En particular, se toma el 70 % de los mensajes para crear el *conjunto de entrenamiento*, utilizando el 30 % restante como *conjunto de prueba* para evaluar el rendimiento de los métodos (de aprendizaje supervisado). La misma operación se repite para la categorización del contenido. En el primer nivel, se identifican los mensajes que hablan de cada temática y, en el segundo, se identifican las tendencias de opinión.

Una vez hecho esto, los mensajes deben prepararse para el análisis. Para ello, cada uno de ellos se trata utilizando la librería de minería de textos *tm*, disponible en R (2012), que permite eliminar palabras vacías (palabras comunes que normalmente no añaden información útil para el análisis), números, signos de puntuación y espacios en blanco. Después de esto, el texto de todos los mensajes se convierte a minúsculas y se utiliza la función *termDocumentMatrix* de *tm* para generar la matriz de términos-mensajes

Debemos indicar que sólo los mensajes utilizados para entrenamiento contribuyen a la construcción del diccionario de términos cuando un nuevo mensaje del conjunto de prueba se analiza con el clasificador, se ignora cualquier término que no aparece en los mensajes de entrenamiento a la hora de calcular la distancia entre mensajes. Es importante indicar qué sinónimos, abreviaciones o formas alternativas para un término dado no han sido tenidas en cuenta en el trabajo realizado.

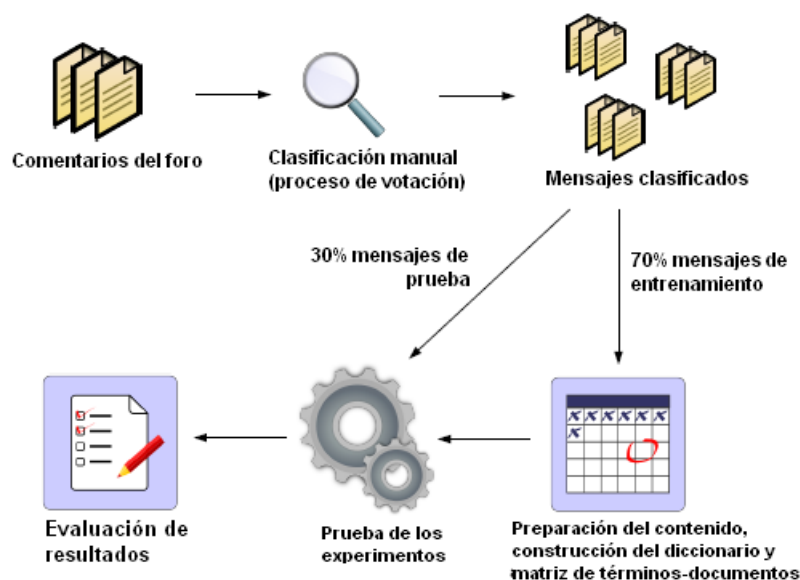


Figura 5.2: Descripción de la metodología.

5.4.2. Clasificación de textos y análisis de sentimientos

En los dos primeros niveles del procedimiento multifase descrito anteriormente se han comparado los resultados de dos métodos de aprendizaje supervisado que pueden aplicarse a la clasificación de textos y al análisis de sentimientos en los mensajes:

- *k-NN*: Este método viene proporcionado directamente por la librería *tm*. Se basa en el contenido de todo el conjunto de mensajes que se pretenden analizar. Siguiendo este procedimiento, explicado en la Sección 5.3.2, se obtiene la distancia de todos los mensajes de prueba a los mensajes de entrenamiento de cierta categoría C . En el primer nivel, se consideran dos categorías (que para nuestra discusión denominamos Candidato A o Candidato B), mientras que en el segundo nivel se comparan los mensajes de prueba con el conjunto de mensajes de entrenamiento en tres categorías diferentes (C_{+1} , C_0 y C_{-1}).

De este modo, se calculan las distancias de la *matriz de mensajes-mensajes* para ca-

da categoría C , como puede verse en la Tabla 5.2. A continuación, se ordenan estas distancias para cada categoría, de modo que se pueda calcular el percentil p que maximiza el rendimiento del clasificador. Así, tomando este percentil como referencia de la distancia a cada clase, se asigna finalmente el mensaje de prueba a la categoría cuya distancia es menor. Si el mensaje de entrenamiento se encuentra a la misma distancia de dos categorías, no se asigna a ninguna y se almacena para ser evaluado manualmente. Una descripción más extensa puede verse en Muñoz y Moguerza (2005).

- *Clasificación utilizando SVM*: En este caso, se ha seguido el método SVM explicado en la sección 5.3.3 para clasificar los mensajes de los Candidatos A o B (primer nivel), e identificar dichos mensajes siguiendo diferentes tendencias de opinión, pertenecientes a las categorías C_{+1} , C_0 o C_{-1} (segundo nivel). En este caso, hemos utilizado un núcleo lineal para esta tarea, habiéndose obtenido resultados similares utilizando el núcleo radial. Este método está disponible en varios paquetes del entorno estadístico R, del que hemos seleccionado la librería *e1071*.

Diccionario	Prueba 1	Prueba 2	...	Prueba t
Entrenamiento 1	d_{11}	d_{12}	...	d_{1t}
Entrenamiento 2	d_{21}	d_{22}	...	d_{2t}
Entrenamiento 3	d_{31}	d_{32}	...	d_{3t}
...
Entrenamiento a	d_{a1}	d_{a2}	...	d_{at}

Cuadro 5.2: Matriz mensaje-mensaje generada por el método k -NN para una categoría C . d_{at} representa la distancia entre el mensaje de entrenamiento a y el mensaje de prueba t .

Finalmente, procedemos a la extracción de opiniones. En este nivel, nuestro objetivo es obtener las palabras relevantes que caracterizan los comentarios asignados a las diferentes tendencias de opinión (positivo, negativo o neutro) catalogadas para el Candidato A o el

Candidato B. Con este objetivo, parece razonable utilizar métodos de aprendizaje no supervisados, pues no tenemos categorías o temas de interés predefinidos que podamos utilizar. Para este propósito, uno de los algoritmos de agrupamiento comúnmente utilizado es el método de k -medias, véase Mardia et al. (1980).

En nuestro caso, aplicamos el método de k -medias (con $k = 4$) para identificar un número variable de grupos de mensajes en cada categoría (C_{+1} , C_0 y C_{-1}), dependiendo del número de mensajes asignados a dicha categoría. Seguidamente, se inspeccionan los mensajes prototipo que actúan como centroides de los grupos para obtener las palabras que caracterizan el contenido de los mismos. Así, obtenemos las palabras más relevantes definidas en los comentarios enviados por los participantes sobre cada candidato para elaborar las tendencias de opinión.

5.5. Presentación y resultados del experimento

En esta sección, se presentan los resultados del experimento comparando diferentes métodos para la clasificación y el análisis de sentimientos del procedimiento multifase explicado anteriormente. En este caso, se han simulado 10 repeticiones para cada uno de los dos primeros niveles (clasificación de los mensajes y tendencias de opinión).

5.5.1. Datos del problema

Los mensajes utilizados para realizar el estudio de la metodología presentada pertenecen al foro de debate de un experimento propuesto con la arquitectura presentada en el Capítulo 2. En dicho experimento, se realizaron diversas cuestiones para conocer las opiniones de los participantes sobre dos posibles representantes (Candidato A y Candidato B) a unas futuras elecciones en una asociación de empresarios. En el Cuadro 5.3 se muestran algunas estadísticas generales sobre el número de mensajes analizados y el número medio de palabras de estos mensajes.

Estadísticos descriptivos	Valor
Número de mensajes analizados	483
Número medio de palabras por mensaje	39.83
Número medio de palabras por mensaje (filtrando palabras vacías)	16.14
Número total de mensajes del Candidato A	252
Número total de mensajes del Candidato B	231

Cuadro 5.3: Estadísticas generales del caso analizado.

Del mismo modo, en el Cuadro 5.4 podemos ver algunos ejemplos de mensajes clasificados en cada una de las diferentes categorías consideradas en el análisis.

Candidato	Categoría de opinión	Mensaje de muestra
A	C_{+1}	<i>Candidato A</i> te apoyamos, cuenta con mi voto para las próximas elecciones.
A	C_0	Me gustaría saber la opinión del <i>Candidato A</i> sobre la subida de impuestos a los empresarios de la zona.
A	C_{-1}	Pienso que el <i>Candidato A</i> es una mala persona, él solamente quiere mandar.
B	C_{+1}	De hecho, pienso que el <i>Candidato B</i> ofrece una buena imagen para el resto de empresarios.
B	C_0	El <i>Candidato B</i> ya nos representó en años anteriores.
B	C_{-1}	¿Es cierto que algunas personas no ven que el <i>Candidato B</i> esta dañando al resto de empresarios?

Cuadro 5.4: Ejemplo de mensajes asignados a cada categoría considerada en el caso de uso.

5.5.2. Métricas de rendimiento para clasificación

Para evaluar el rendimiento de los diferentes métodos de clasificación de textos y de análisis de sentimientos, se han utilizado las métricas *precision*, *recall* y *accuracy* para la recuperación de información, véase Baeza-Yates y Ribeiro-Neto (2011) y Olson y De-len (2008). Para ello, definimos *verdaderos positivos* (t_p), como el conjunto de mensajes correctamente clasificados; *verdaderos negativos* (t_n), como el conjunto de mensajes que correctamente no se han catalogado en cierta clase; *falsos positivos* (f_p) a los mensajes que han sido incorrectamente clasificados en una categoría y *falsos negativos* (f_n) a aquellos mensajes catalogados incorrectamente en una categoría que no era la suya. Como consecuencia, se definen las métricas:

- *Precision* para la clase C_i : Es la fracción de mensajes asignados a la clase C_i que pertenecen a esa clase.

$$Precision = \frac{t_p}{t_p + f_p}$$

- *Recall* para la clase C_i : Es la fracción de mensajes que deberían pertenecer a la clase C_i que se clasifican correctamente.

$$Recall = \frac{t_p}{t_p + f_n}$$

- *Accuracy*: Es la proporción de mensajes correctamente catalogados sobre todos los mensajes disponibles.

$$Accuracy = \frac{t_p + t_n}{t_p + t_n + f_p + f_n}$$

5.5.3. Clasificación de mensajes

En el primer nivel, nuestro objetivo era comparar el rendimiento de los algoritmos k -NN y SVM para la clasificación de mensajes, diferenciando entre comentarios dirigidos al Candidato A y al Candidato B. La Figura 5.3 resume los resultados para este caso. Como se podría esperar de antemano, los resultados para el método SVM son mejores que los del método k -NN, aunque por pequeño margen. El hecho de tener unos temas bien definidos y precisos para la clasificación de mensajes facilita la tarea a ambos métodos de aprendizaje, ya que se encuentra una clara distinción entre los mensajes dirigidos al Candidato A o al Candidato B.

Otro resultado destacable es la baja variabilidad en cualquiera de las tres medidas de precisión en ambos métodos. Esto indica que la estimación de las métricas de rendimiento es bastante acertada y existe escasa influencia por la selección de diferentes muestras de mensajes en cada experimento.

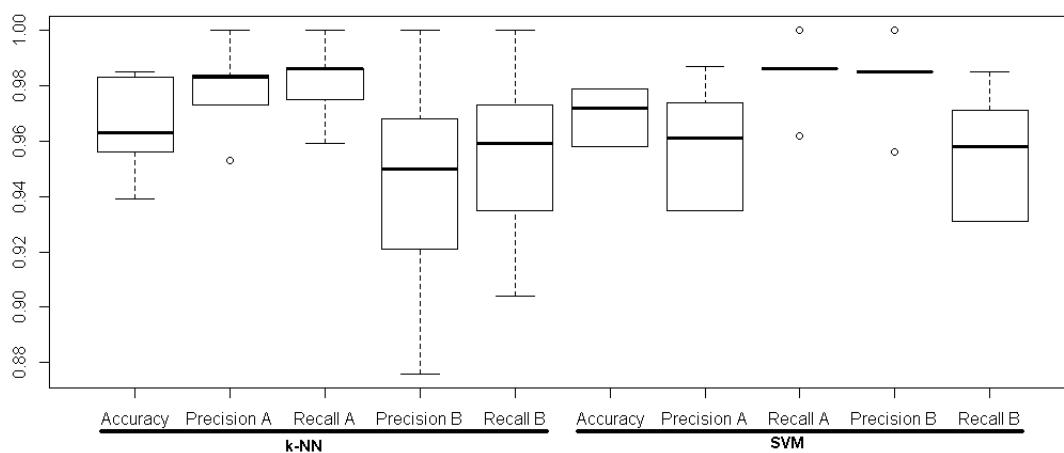


Figura 5.3: Rendimiento de algoritmos de clasificación de mensajes en el primer nivel.

5.5.4. Identificar tendencias de opinión

En el segundo nivel se compara el rendimiento de los métodos de aprendizaje supervisado para la clasificación de mensajes en cada una de las tres posibles categorías de opinión: C_{+1} , C_0 o C_{-1} . En la Figura 5.4 se muestran los resultados obtenidos en los experimentos para los mensajes clasificados como del Candidato A. Para el Candidato B se obtuvieron resultados similares. A pesar de tener tres categorías en esta etapa, se resumen los resultados mediante la comparación de dos categorías, C_{+1} y C_{-1} , puesto que son nuestro foco de atención para comparar tendencias de opinión positivas o negativas.

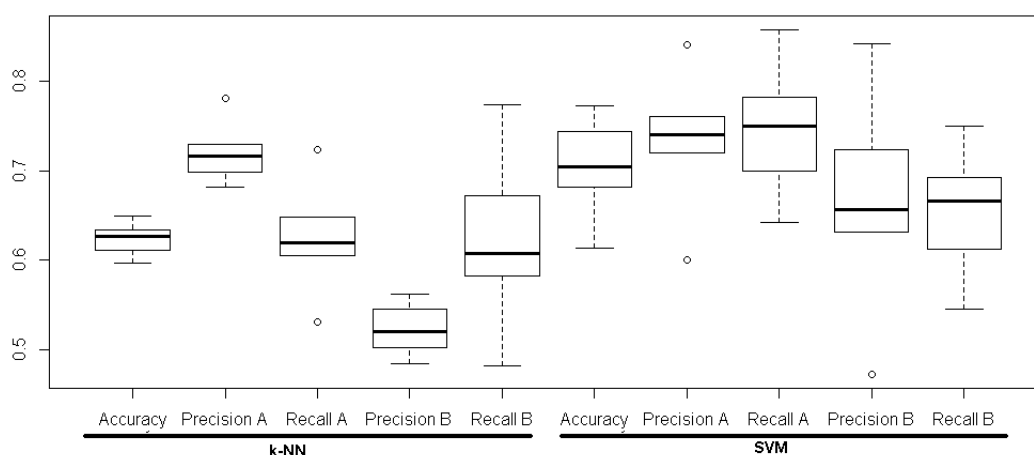


Figura 5.4: Métricas de rendimiento para el segundo nivel del método multifase (sin palabras vacías), para los mensajes asignados al Candidato A.

Como se observa, los resultados son bastante buenos, pero no tanto como en el primer nivel. La identificación de tendencias de opinión es más difícil que la categorización de texto, puesto que la distinción de los mensajes relacionados con alguna de las tres categorías consideradas puede depender sólo de unas pocas palabras. Además, la variabilidad en las estimaciones continúa siendo bastante buena, como indican los pequeños valores obtenidos en la desviación típica.

Por otra parte, los mensajes escritos en el foro son muy cortos cuando las palabras vacías u otra información textual superflua es eliminada. Debido a ello, la frecuencia de aparición de las palabras importantes para detectar tendencias de opinión es mucho más baja que en problemas típicos de categorización de textos donde los documentos son normalmente largos y los términos que caracterizan cada tema están más definidos. Por esta razón, es importante saber si podríamos mejorar nuestros resultados manteniendo las palabras vacías que son normalmente filtradas en los métodos tradicionales de clasificación de textos. Así, decidimos repetir el experimento con las palabras vacías en un intento de mantener términos que pudieran servir para discriminar diferentes tendencias de opinión y que pudieran mejorar nuestro experimento anterior.

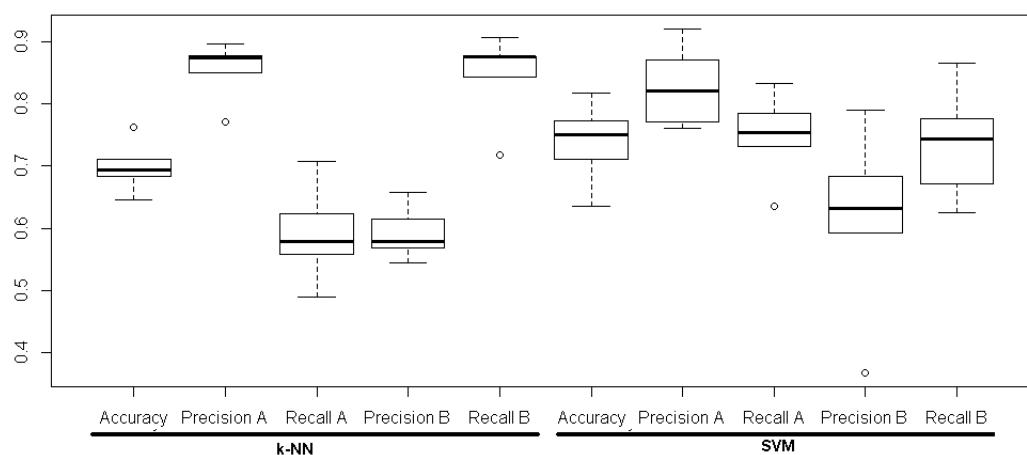


Figura 5.5: Métricas de rendimiento para el segundo nivel del método multifase (con palabras vacías) para los mensajes asignados al Candidato A.

En la Figura 5.5 se muestran los nuevos resultados manteniendo palabras vacías. En general, se observa que mejoran en ambas categorías aproximadamente un 10-15 %, excepto para el recall de la categoría C_{+1} y la precisión de la categoría C_{-1} . Esto sugiere que el hecho de eliminar palabras vacías influye en la detección de tendencias de opinión, especialmente en los casos en los que se trabaja con mensajes cortos y baja frecuencia de aparición de cada término en los documentos. Por ejemplo, se filtran los términos que cla-

ramente contienen significado positivo o negativo como *bueno*, *malo* o *mejor* en el caso del primer experimento. Esto muestra cómo eliminando palabras vacías que no tienen ninguna influencia en la categorización clásica de grandes documentos puede tener un impacto negativo en la detección de opiniones, ya que desaparecen términos que pueden ayudar a encontrar las mismas.

5.5.5. Extracción de opiniones y palabras relevantes

Para concluir el análisis, se aplica el método k -medias para identificar los grupos de mensajes clasificados en las categorías C_{+1} , C_0 y C_{-1} para los Candidatos A y B. Para simplificar, sólo se muestran los resultados de la versión SVM del esquema multifase, puesto que los resultados del método k -NN son similares. El Cuadro 5.5 muestra la lista de términos que caracterizan los diferentes grupos identificados para cada tendencia de opinión. En este caso, se han mantenido las palabras vacías para minimizar la probabilidad de eliminar términos destacables que pudieran definir cada tendencia. En el caso del Candidato A, el número de mensajes positivos era menor que en las otras dos clases y sólo se han identificado dos grupos. Para el Candidato B, no se ha identificado claramente ningún grupo para los mensajes positivos, y sólo se han identificado dos grupos para los mensajes de la categoría C_0 . En el resto de casos, se han identificado 4 grupos diferentes de mensajes.

Examinando las palabras de cada grupo, resulta sencillo encontrar rápidamente algunos términos relevantes que caracterizan los diferentes temas de opinión. Entre ellos, existen referencias a temas económicos y de enseñanza, procesos burocráticos y enlaces a un proceso electoral. En este sentido, se observa cómo podemos aplicar esta técnica para obtener más información acerca de los conceptos e ideas asociadas a cada corriente de opinión. La investigación en profundidad de estos resultados puede llevar a una lista más elaborada de términos importantes relacionados con cada categoría. De hecho, esta lista puede utilizarse como entrada de datos para mejorar la toma de decisiones en política o encontrar los intereses que más preocupan a los usuarios que participan en el debate. Además, esta lista puede utilizarse para generar los informes que expliquen las decisiones tomadas para aumentar el

grado de satisfacción de los usuarios.

Candidato/tendencia	Grupo 1	Grupo 2	Grupo 3	Grupo 4
A/C_{+1}	felicidades, excelente, documento, elegido	arbol, bueno, socios, correcto	N/A	N/A
A/C_0	promesas, facultad, reunión, declaración	ataque, directores, arbitrario, confortable	interinos, calidad, sentencia, adherirse	elecciones, paz, amigos, brillante
A/C_{-1}	enseñanza, artículo, católico, creyente	universidad, carta, culto, disciplinado	advertir, barreras, corrupción, difamatorio	dinero, actividades, conservador, gerente
B/C_{+1}	N/A	N/A	N/A	N/A
B/C_0	juez, amigos, comisión, mandato	asociado, divertido, piensa, por favor	N/A	N/A
B/C_{-1}	sentencia, elecciones, departamento, justicia	admitido, contrario, denunciado, prohibir	reunión, consejo, compasión, responsable	explicación, fundación, grave, acoso

Cuadro 5.5: Términos que caracterizan los grupos identificados por el algoritmo k -medias, manteniendo palabras vacías. *N/A* representa *No disponible*.

5.5.6. Comparación de experimentos

Las investigaciones previas en análisis de sentimientos y minería de opiniones se han centrado usualmente en la comparación de métodos para mejorar el rendimiento en la detección de corrientes de opinión o en la clasificación de mensajes. Sin embargo, las dependencias entre la categorización de temas y el análisis de sentimientos rara vez han sido exploradas.

Hemos mostrado un método multifase que puede adoptarse en aquellos casos en los que

resulte necesaria la clasificación textual de documentos, de acuerdo a los temas relevantes de opinión y considerando el análisis de sentimientos para la clasificación. Este procedimiento aporta importantes ventajas:

- Modularidad en la clasificación de textos. En lugar de realizar al mismo tiempo la categorización de textos y la detección de tendencias de opinión, se pueden implementar diferentes procedimientos en cada nivel de acuerdo a las necesidades específicas. Siguiendo esta aproximación, se pueden aprovechar los métodos precisos para la categorización de contenidos, obteniendo un rendimiento superior en esta tarea y, a continuación, centrarnos en el análisis de sentimientos para cada categoría identificada en el primer nivel.
- Requisitos de acierto para el análisis de sentimientos. Puesto que se ha obtenido un buen rendimiento en la clasificación de contenido en el primer nivel, la detección de tendencias de opinión puede ser satisfactoria con unos buenos resultados de rendimiento.
- Combinación de métodos supervisados y no supervisados. La combinación adecuada de ambas técnicas proporciona resultados más satisfactorios que la sola aplicación de métodos supervisados de aprendizaje. Con este enfoque, no sólo somos capaces de identificar diferentes corrientes de opinión, sino que conseguimos caracterizar grupos de mensajes relacionados por la tendencia de acuerdo a su contenido. Esto claramente proporciona más información que un mero análisis cuantitativo de la cantidad de mensajes asignados a cada categoría.

Del mismo modo, podemos ofrecer una serie de recomendaciones a partir de los resultados y la experiencia práctica obtenida a través de estos experimentos:

- Clasificar mensajes cortos es un reto: La clasificación de mensajes cortos no es comparable a otros tipos de documentos como libros, artículos, noticias, etc. Esto es es-

pecialmente relevante para la información textual extraída de páginas web o servicios de micro-blogging.

- Alto rendimiento de las SVM: En general, el uso de métodos de clasificación SVM ofrece mejores resultados de rendimiento que los métodos k -NN, tanto en categorización de textos como en análisis de sentimientos. Sin embargo, esta diferencia no es muy notable. En cualquier caso, el escaso tiempo necesario para entrenar una SVM comparado con k -NN y su rapidez de computación, especialmente con grandes conjuntos de documentos, son ventajas importantes.
- Influencia de las palabras vacías: Los métodos tradicionales para la categorización de textos recomiendan la eliminación de palabras vacías como paso fundamental para mejorar el rendimiento de estas técnicas. Sin embargo, como hemos visto en estos experimentos, algunos términos identificados como palabras vacías por las librerías de análisis textual pueden contener información válida desde el punto de vista del análisis de sentimientos. Por lo tanto, las palabras vacías no deben eliminarse sin tener en cuenta qué términos podrían mejorar la tasa de acierto en la detección de tendencias de opinión.

5.6. Conclusiones

En este capítulo se ha presentado una metodología multifase de análisis de sentimientos y extracción de opiniones combinada con algoritmos y técnicas de aprendizaje supervisado y no supervisado. El objetivo final es la detección de tendencias de opinión positivas y/o negativas de los mensajes obtenidos del foro de discusión presentado en nuestra arquitectura. El esquema ha sido probado exitosamente con los mensajes de un foro, aportando información valiosa al dueño del problema, facilitando la explicación de las decisiones tomadas. El diseño de la herramienta descrita en este capítulo es bastante flexible permitiendo la detección de cambios en las tendencias de opinión en un corto espacio de tiempo, de modo que, puede ser adaptada a actualizaciones en tiempo real de las tendencias cuando llegan al

sistema nuevos mensajes.

Las tendencias de opinión se representan mediante vectores prototipo de palabras que pueden ser interpretados por los responsables del problema. El esquema mostrado puede adaptarse a cualquier foro específico utilizando, por ejemplo, métodos alternativos tales como el análisis de sentimientos latente o mediante la construcción de diccionarios específicos para dicho análisis, además de utilizar términos derivados de los documentos bajo estudio.

La herramienta propuesta puede utilizarse para la generación de informes una vez que ha concluido el proceso que permita explicar a los participantes las conclusiones obtenidas. Por lo tanto, estaría enlazada con el módulo de explicación contenido en la arquitectura presentada en capítulos anteriores. También se puede usar directamente, por ejemplo, para la detección de tendencias de opinión populares en campañas electorales o para conocer el grado de satisfacción con nuevas leyes o políticas públicas. Del mismo modo, también puede adoptarse en otras aplicaciones tales como la categorización automática y detección de mensajes relacionados con campañas de publicidad, productos y servicios comerciales, así como las críticas u opiniones a páginas web.

Capítulo 6

Conclusiones y cuestiones abiertas

En este capítulo se presentan las conclusiones fundamentales de este trabajo. Además, se plantean líneas futuras de investigación sobre algunos aspectos que quedan abiertos o sobre temas que, tratados con mayor profundidad, pudieran proporcionar una mejor solución a las metodologías propuestas.

6.1. Conclusiones

A continuación, se recogen a modo de reflexión final, las conclusiones obtenidas en los distintos capítulos de esta tesis doctoral. Su objetivo último ha sido crear una serie de herramientas para diseñar y ejecutar los instrumentos participativos ya existentes, así como, nuevos que pudieran surgir. Con la finalidad de dotar a la ciudadanía de los mecanismos necesarios para tomar decisiones tratamos, tal vez, de aumentar los índices de participación y mejorar su satisfacción.

En el capítulo 1, se describe el concepto de participación ciudadana. Esta ha estado

presente a lo largo de la historia aunque, en la actualidad, se encuentra en una posición muy relevante en la sociedad, representada por la democracia, aportando importantes beneficios. Para ello, debe adoptar una serie de principios que generen valor, y no desconfianza, entre los ciudadanos.

En los últimos años, se ha visto deteriorada la relación entre ciudadanos y políticos que ha originado la aparición de diferentes instrumentos de participación. Dichos instrumentos permiten que los ciudadanos tomen parte en las decisiones de problemas que les afectan consiguiendo aumentar la legitimidad, aceptación y transparencia de tales decisiones. Además, están desarrollándose nuevos instrumentos, gracias a la expansión en el uso de las TIC que tienen gran aceptación entre los ciudadanos debido principalmente por su rapidez.

A continuación, se describen algunos de los mecanismos de participación más extendidos, detectando que la mayoría de ellos son el resultado de la combinación en distinto orden de diversas tareas participativas comunes. De este modo, pensamos en la creación de un algoritmo genérico que permita diseñar nuevos procesos, coordinando las tareas y adaptando su orden de aparición a las necesidades de cada caso. Así, surge la idea de desarrollar una arquitectura basada en servicios web, que estuviera dividida en módulos o componentes, donde cada uno de ellos, se encarga de implementar una o varias tareas participativas, de modo que la construcción de un nuevo proceso sea una labor sencilla y rápida. Al dividirse en módulos permite conseguir un bajo acoplamiento entre los mismos y una mayor escalabilidad, de modo que se puedan añadir nuevos elementos sin necesidad de realizar cambios importantes. Al sistema sólo pueden acceder usuarios registrados mediante la identificación con nombre de usuario y contraseña, a través de una interfaz web.

Seguidamente, revisamos las ventajas y facilidades que aporta la tecnología XML en cualquier ámbito de trabajo con sistemas Web. Se muestran los métodos más extendidos para crear las reglas sintácticas de un documento XML y cuáles fueron los motivos para decidir el uso de esquemas XML. Se presentan entonces las partes en las que se divide el esquema XML que permite el intercambio de información entre los componentes de

nuestra arquitectura. Todos los módulos comparten la información general del problema como nombre, descripción y características pero, también tienen sus particularidades, que son expuestas en la sección correspondiente a cada módulo.

En el Capítulo 4 se presenta un procedimiento para diseñar procesos participativos de forma sencilla. Para ello, nos basamos en los sistemas BPM, que son herramientas para mejorar la eficiencia de los procesos. Puesto que un proceso participativo puede considerarse como una combinación de tareas en las que participan diferentes actores, se exponen las reglas que consideramos necesarias para que cualquier persona pueda diseñar un proceso participativo de una forma sencilla. Para ello, el usuario únicamente deberá arrastrar objetos e indicar el perfil de los usuarios que pueden actuar en cada etapa, la duración de la misma y la localización del servicio web que da soporte a esa tarea. Al mismo tiempo, consideramos necesario que el sistema ofrezca el proceso óptimo, que cumpla las restricciones, a partir de las preferencias de diseño del usuario. Para que luego pueda ejecutarse ese proceso, se debe permitir exportar en formato XML el diseño creado, validándolo contra el esquema XML del capítulo anterior.

Por último, se ha desarrollado una metodología que permite extraer tendencias de opinión y análisis de sentimientos a partir de los mensajes escritos por los usuarios en foros, utilizando alguna de las herramientas disponibles en la arquitectura descrita. Del mismo modo, esta metodología se utiliza para generar los informes definitivos, una vez que el proceso ha concluido, para presentar de forma legible las decisiones tomadas a todos los participantes. Se trata de una metodología multifase que combina algoritmos y técnicas de aprendizaje supervisado y no supervisado. El diseño de la herramienta es flexible, puesto que permite detectar cambios en las tendencias de opinión en tiempo real cuando llegan al sistema nuevos mensajes.

6.2. Mejoras futuras

El objetivo final de esta Tesis ha sido el desarrollo de una plataforma que permita a los ciudadanos tomar parte en las decisiones que más les afectan.

Aunque cada vez menos, buena parte de la sociedad actual es reticente a, por ejemplo, ejercer su derecho a voto a través de Internet porque no confía en que los votos puedan ser manipulados. Por contra, no suele existir ningún inconveniente en enviar nuestro voto por correo ordinario o permitir que el camarero de un restaurante cobre con la tarjeta de crédito sin nuestra presencia, cuando son acciones mucho más inseguras que acceder a una web gubernamental. Además, muchos de los agujeros de seguridad en los sistemas se deben a fallos de los usuarios que comunican sus credenciales a través de correos electrónicos falsos, como ejemplos claros de spam o phishing, o utilizan contraseñas débiles fácilmente descifrables. Como vemos, uno de los aspectos más importante en el mundo de la tecnología es la seguridad, que permite que los usuarios que participan en un sistema, confíen en que sus datos no serán revelados a terceras personas ni se vulnerarán aspectos de privacidad importantes como pueden ser datos personales, temas de opinión o votaciones. En el caso de nuestra arquitectura este sería un punto importante que debe tratarse. La seguridad que se ha implementado es sencillamente un protocolo de encriptación de datos en las peticiones web, pero habría que securizar otros aspectos como son el acceso a la base de datos o añadir nuevas técnicas de acceso a la aplicación. En este sentido, una buena aproximación, sería basarla en el uso del DNI electrónico, que provee a los ciudadanos una identidad digital homogénea y robusta, sentando las bases necesarias para el intercambio seguro de información.

Respecto al sistema implementado, podría mejorarse la interfaz de la aplicación, cumpliendo las normas de accesibilidad e implementando mecanismos que faciliten a los usuarios tareas con cierta complejidad, como puede ser la comunicación de preferencias, en la que sería interesante desarrollar algún método que sea fácilmente comprensible por los usuarios que participen en los procesos.

En el caso de los documentos XML, se podría utilizar alguno de los estándares más extendidos como XSLT (transformaciones XSL) o XPATH, para transformar los documentos generados en cada módulo de la arquitectura, en páginas web que presenten los datos de forma más visual en un navegador web. De este modo, se podrían visualizar todos los datos del problema antes de cargarlo en el sistema.

Todos los datos almacenados en los XML deberían cifrarse utilizando, por ejemplo, el estándar “XML Encryption” que especifica un lenguaje para asegurar la confidencialidad de los documentos XML cifrando parcial o totalmente dicho documento. Al mismo tiempo, se podría utilizar XML Signature que asegura integridad a los XML proporcionando autenticidad a los mismos.

Del mismo modo, sería interesante crear un repositorio donde se almacenen XML de procesos estándar para cada una de las fases de los distintos mecanismos de participación. Así, se consigue un ahorro considerable de tiempo y trabajo cuando sea necesario utilizar una fase común a varios procesos. El repositorio contaría inicialmente con los procesos definidos anteriormente y, a medida que se fueran creando nuevos, se añadirían al mismo, consiguiendo, con el paso del tiempo, un gran almacén de procesos participativos listos para ser utilizados y con el único esfuerzo de cargarlos en el sistema.

Este repositorio sería extensible al caso de los procesos creados con la herramienta propuesta de diseño de procesos, ya que, se podría contar con varios diseños iniciales de experiencias anteriores almacenados en el sistema, de modo que, el usuario pudiera seleccionar alguno de los procesos guardados. Además, se podría implementar algún tipo de búsqueda con palabras clave, para agilizar la tarea de encontrar el proceso que más se adapte a las necesidades del usuario.

Por otra parte, para completar el ciclo de actividades de los sistemas BPM, podría implementarse alguna herramienta que permita analizar los procesos para optimizar su rendimiento, tratando de mejorar la toma de decisiones, aumentar la participación entre los usuarios, reducir los gastos asociados al proceso o corregir posibles errores de diseño en el

mismo.

Dado que es bastante común en la minería de textos y el análisis de sentimientos tener matrices dispersas, para la representación vectorial de los mensajes podría utilizarse diferentes técnicas para disminuir el tamaño del diccionario. Una de ellas podría ser la técnica de “stemming” que utiliza la raíz de una palabra para englobar todos los términos que tienen el mismo significado. También, se podría eliminar del diccionario de términos palabras atípicas, bien por su baja frecuencia de aparición, como por ejemplo palabras que aparecen en un solo documento, o bien, porque aparecen muchas veces pero sólo en un documento y, por lo tanto, no van a aportar mucha información. De este modo, se consigue un diccionario de términos más reducido, por lo que la longitud de los vectores que representan los mensajes disminuye y, por consiguiente, se produce una mejora sustancial respecto al tiempo de computación.

Se podrían utilizar otras métricas de frecuencia de palabras como tf-idf. De este modo, no se tendría únicamente en cuenta la frecuencia de aparición de los términos en los documentos, sino también cuántas veces aparece dicho término en el resto de documentos. Así, puede conocerse la relevancia de los términos y, en caso de que no sea relevante, eliminarlo del diccionario.

Al mismo tiempo, se podría implementar el sistema para que soporte múltiples idiomas utilizando, por ejemplo, traductores online, puesto que, realmente no nos interesa que la traducción sea correcta, sino, las palabras que aparecen en los mensajes. En este caso, habría que obtener los conjuntos de palabras vacías de diversos idiomas para eliminar palabras que no contienen información relevante.

También, podría mejorarse el sistema de explicación mediante la creación de diferentes plantillas que faciliten la redacción de los informes finales dependiendo del proceso. El informe constaría de una estructura predefinida donde habría espacios para añadir las decisiones adoptadas y las razones que han motivado la toma de dichas decisiones.

Además, gracias a la flexibilidad del método multifase propuesto, que puede configu-

rarse, personalizarse y extenderse según las necesidades específicas, podría aplicarse a otros campos como la categorización y detección de opiniones de los mensajes recibidos en las críticas de tiendas online, sistemas automatizados para la moderación en sitios web de participación como blogs, foros, espacios de noticias, etc., en los cuales se marcan los mensajes negativos para la revisión de los administradores de la web antes de su publicación, como ayuda para la categorización y etiquetado de contenido (incluyendo información sobre connotaciones positivas o negativas) de mensajes enviados a sitios web de preguntas-respuestas como Quora, Stack Exchange o Yahoo! Answers o para la clasificación y extracción de opiniones previa del contenido obtenido en redes sociales relacionadas con las campañas de publicidad y promociones, procesos electorales, plataformas de democracia participativa y administraciones públicas e instituciones.

Todos estos problemas abiertos serán objeto de trabajo futuro.

Bibliografía

- Ainsworth, S. E. 1999. Designing effective multi-representational learning environments. *ESRC Centre for Research in Development, Instruction and Training, University of Nottingham*.
- Alfaro, C., Gómez, J., Ríos, J. 2010. From participatory budgets to e-participatory budgets en Ríos Insua, D. y French, S. (eds.) *e-Democracy. A Group Decision and Negotiation Perspective*. Springer. 5: 283-300.
- Arenilla, M. 2007. *Los modelos, proyectos y políticas de participación en grandes y medias ciudades*. Librería-Editorial Dykinson.
- Arenilla, M. 2008. Planificación y presupuestos participativos en la Ciudad de Madrid. *Revista Quorum*, nº. 20, p.109-126.
- Arenilla, M. 2010. Concepts in Democratic Theory en Ríos Insua, D. y French, S. (eds.) *e-Democracy. A Group Decision and Negotiation Perspective*. Springer. 5: 15-30.
- Arnstein, S. 1971. Ladder of participation in the USA. *Journal of the Royal Town Planning Institute*. p.176-182.
- Barrat i Esteve, J., Reniu i Vilamala, J.M. 2004. Democracia electrónica y participación ciudadana, *Informe sociológico y jurídico de la Consulta Ciudadana "MadridParticipa"*, Madrid: Ayuntamiento de Madrid.
- Baeza-Yates, R., Ribeiro-Neto, B. 1999. *Modern Information Retrieval*. Addison Wesley.

- Baeza-Yates, R., Ribeiro-Neto, B. 2011. *Modern Information Retrieval: The Concepts and Technology behind Search*. Addison-Wesley Professional. URL <http://www.mir2ed.com>
- Benyoucef, M., Keller, R. K., Lamoureaux, S., Robert, J., Trussart, V. 2000. *Towards a Generic E-Negotiation Platform, Proceedings, Re-Technologies for Information Systems*, 6th International Conference, Austrian Computer Society, Zurich, Switzerland, pp. 95-109.
- Benyoucef, M., Verrons, M-H. 2008. Configurable e-negotiation systems for large scale and transparent decision making. *Group Decision and Negotiation*, Vol. 17, No. 3, Springer, pp. 211-224.
- Bessette, Joseph M. 1980. Deliberative Democracy: The Majority Principle in Republican Government, en R. Goldwin and W. Schambra (eds.), *How Democratic is the Constitution*, Washington: American Enterprise Institute. p.102-116.
- Bisdorff, R., Meyer, P., Veneziano, Th. 2009. *XMCD: a standard XML encoding of MC-DA data*. Invited presentation at the 23rd European Conference on Operational Research Bonn (DE), July 5-8.
- Blanco, I. 2002. *Presupuesto Participativo y Democracia Local: una comparación entre las experiencias brasileñas y españolas*. VII Congreso Internacional del CLAD sobre la Reforma del Estado y de la Administración Pública, Lisboa, Portugal, 8-11 Oct. 2002
- Blondiaux, L., Sintomer, Y. 2002. L'impératif délibératif, *Politix*, n° 57, p. 17-35.
- Borge, R. 2005. La participación electrónica: estado de la cuestión y aproximación a su clasificación». IDP. Revista de Internet, Derecho y Política [artículo en línea]. N.º 1. UOC. <<http://www.uoc.edu/idp/1/dt/esp/borge.pdf>> ISSN 1699-8154.
- FUNDACIÓ BOFILL 2004. *La participació ciutadana a través de les noves tecnologies. Estratègies per a la utilització de Consensus*. Finestra Oberta. N° 42.
- Campos, A., de Almeida, A. 2010. Using Multiple Criteria Decision Analysis for Supporting Decisions of Business Process Management. Industrial Engineering and Engineering Management (IEEM). IEEE International Conference on Issue, pp. 52-56, Macao.

-
- Carlisle, D., Ion, P., Miner, R. 2009. Mathematical Markup Language (MathML) Version 3.0. Disponible online vía <http://www.w3.org/TR/MathML/>.
- Choi, I., Jeong, H., Song, M., Ryu, Y. 2005. IPM-EPDL: an XML-based executable process definition language. *Computers in Industry*, 56, pp. 85–104.
- Cohen, J. 1998. “Democracia y libertad”, en ELSTER, J. (comp.), *La democracia deliberativa*, pp. 245-281. Barcelona, Gedisa.
- Clemen, R.T., Reilly, T. 2001. *Making hard decisions with decision tools*. Duxbury Press.
- Dhaliwal, J. S., Benbasat, I. 1996. The use and effects of knowledge-based system explanations: Theoretical foundations and a framework for empirical evaluation. *Information Systems Research*, 7, 342-362.
- Dollery, B.E., Byrnes, J.D., Crase, L. 2008. A note on structural reform in Australian local government, *Australian Journal of Political Science*, 43(2), 333-339.
- El-Gayar, O., Tandekar, K. 2007. An XML-based schema definition for model sharing and reuse in a distributed environment. *Decision Support Systems*, 43(3), 791–808.
- Elster, J. 1998. Introduction, en J. Elster (ed.), *Deliberative Democracy Cambridge*, Cambridge University Press.
- Fitter, M., Green, T.R.G. 1979. “When do diagrams make good computer languages”, *International Journal of Man-Machine Studies*, 11, 235-261.
- Forsythe, D. E., Buchanan, B. G. 1991. Broadening our approach to evaluating medical information systems. In P. D. Clayton (Ed.), 15 th Symposium on Computer Applications in Medical Care (pp. 8-12). New York: McGraw-Hill.
- Fourer, R., Lopes, L., Martin, K. 2005. LPFML: A W3C XML schema for linear and integer programming. *INFORMS Journal on Computing*, 17(2), 139-158.
- Fourer, R., Ma, J., Martin, K. 2010. Optimization Services: A Framework for Distributed Optimization. *Operation Research*, 58(6), 1624-1636.

- French, S., Insua, D. R., Ruggeri, F. 2007. e-Participation and decision analysis. *Decision Analysis*, 4(4), 211-226.
- Freund, J., Rucker, B. y Hitpass, B., BPMN 2.0 Manual de Referencia y Guía Práctica. CreateSpace. 2011. Disponible en <http://books.google.es/>
- Geldermann, J. 2010. Explanation systems en Ríos Insua, D. y French, S. (eds.) *e-Democracy. A Group Decision and Negotiation Perspective*. Springer. 5: 241-259.
- Gómez, J., Ríos Insua, D., Lavín J. M., Alfaro, C. 2013. On deciding how to decide: Designing participatory budget processes, *submitted for publication*.
- Gregor, S., Benbasat, I. 1999. Explanations from intelligent systems: theoretical foundations and implications for practice. *MIS Quarterly*, 23, 497-530.
- Habermas, J. 1991. *The structural transformation of the public sphere: An inquiry into a category of bourgeois society*. The MIT Press.
- Habermas, J. 1999. *La inclusión del otro*, Barcelona, Paidós.
- Habermas, J. (2005). Facticidad y validez, trad. M. JIMÉNEZ, Madrid, Trotta.
- Iglesias, A., García, V., Sanz, M., Mañanes, G. 2004. *Guía de instrumentos de participación ciudadana*. Dirección General de Participación Ciudadana. Área de Gobierno de Economía y Participación ciudadana del Ayuntamiento de Madrid.
- Kalai, E., Smorodinsky, M. 1975. Other Solutions to Nash's Bargaining Problem *Econometrica: Journal of the Econometric Society*, 43, 513-518.
- Keeney, R. L., Raiffa, H. 1993. *Decisions with Multiple Objectives: Preferences and Value Trade-offs*. Cambridge University Press.
- Kolhbacher, M. 2009. *The Effects of Process Orientation on Customer Satisfaction, Product Quality and Time-Based Performance*. 29th International Conference of the Strategic Management Society, Washington DC.

-
- Lacave, C., Díez, F. 2002. A review of explanation methods for Bayesian networks. *The Knowledge Engineering Review*, 17(2), 107–127.
- Lautenbach, B. 2004. Introduction to XML Encryption and XML Signature. *Journal Information Security Tech. Report*, 9(3), p.6-18.
- Lavín, J.M., Ríos, D., Gómez, J., y Alfaro, C. 2013. An online methodology for designing e-Participatory processes. Technical Report. Universidad Rey Juand Carlos. In proceedings.
- Liu, B. 2007. *Web data mining: exploring hyperlinks, contents, and usage data*. Springer Verlag.
- Liu, B. 2012. Sentiment Analysis and Opinion Mining. *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers.
- Madise, Ü. y Maaten, E. 2010. Internet Voting in Estonia en Ríos Insua, D. y French, S. (eds.) *e-Democracy. A Group Decision and Negotiation Perspective*. Springer. 5: 301-321.
- Mardia, K., Kent, J.T., Bibby, J. 1980. *Multivariate Analysis*. Academic Press.
- Martín Granados, I., Pineda Nebot, C. 2007. *La participación presupuestaria local. Estudio comparado de Albacete y Segovia*. VIII Congreso Nacional de Ciencia Política, Valencia.
- McKeown, K. 1983. User-Oriented Explanation for Expert Systems. *Columbia University Computer Science Technical Reports*. New York.
- Mergel, I.A., Schweik, C.M., Fountain, J.E. 2009. The transformational effect of web 2.0. *Technologies on Government*. URL <http://dx.doi.org/10.2139/ssrn.1412796>
- Moguerza, J., Muñoz, A. 2006. Support vector machines with applications. *Statistical Science* 21(3), 322-336.
- Muñoz, A., Moguerza, J. M. 2005. Building smooth neighbourhood kernels via functional data analysis. *Artificial Neural Networks: Formal Models and Their Applications–ICANN 2005*, 631-636.

Plataforma MyUniversity disponible en <https://e-myuniversity.eu/>

Nash, J. F. 1950. The Bargaining Problem, *Econometrica*, 18, 155–162.

Nurmi, H. 2010. Voting Theory en Ríos Insua, D. y French, S. (eds.) *e-Democracy. A Group Decision and Negotiation Perspective*. Springer. 5: 101-123.

Olson, D.L., Delen, D. 2008. *Advanced Data Mining Techniques*, 1st edn. Springer Publishing Company, Incorporated.

O’Reilly, T. 2007. What is web 2.0: Design patterns and business models for the next generation of software. *Communications & Strategies* (1), 18-37. URL <http://ssrn.com/abstract=1008839>

Pérez, J. (2007). Notaciones y Lenguajes de Procesos: Una Visión Global. *Ph. D. in Computer Engineering, University of Sevilla, Sevilla, Spain*.

Pérez Castell, M. 2004. La experiencia de los Presupuestos Participativos en Albacete. *Temas para el Debate*, 113, 54-56.

Petri, C.A. 1962. Kommunikation mit Automaten. Bonn: *Institut für Instrumentelle Mathematik*, Schriften des IIM Nr. 2.

Prieto Marín, P. 2006. Sistemas avanzados para la participación electrónica municipal: ejes conceptuales para su diseño. *Textos de la Cibersociedad*, No. 9, 4.

R Core Team. 2012. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>. ISBN 3-900051-07-0

Raiffa, H., Richardson, J., Metcalfe, D. 2002. *Negotiation Analysis: The Science and Art of Collaborative Decision Making*. Harvard University Press, Cambridge, MA.

Ríos, J., Ríos Insua, D. 2008. A methodology for participatory budget formation, *J. of the Operational Research Society*, 59, 203-212.

-
- Ríos, J., Ríos Insua, D. 2010. Balanced increment and concession methods for negotiation support, *RACSAM*, 104, 41-56.
- Rodríguez, C. 2012. *La mayor y mejor campaña de Social Media de todos los tiempos*. Disponible online en <http://bloglanders.com/2012/04/11/la-mayor-y-mejor-campana-de-social-media-de-todos-los-tiempos/>
- Rowe, G., Frewer, L. 2000. Public Participation Methods: A Framework for Evaluation. *Science, Technology and Human Values* 25(1): 3-29.
- Rowe, G., Frewer, L. 2005. A Typology of Public Engagement Mechanisms. *Science, Technology and Human Values* 30(2): 251-290.
- Sartori, G. 2003. *¿Qué es la Democracia?*. Taurus.
- Silverman, B. 1986. Density Estimation. Chapman and Hall/CRC Monographs on Statistics and Applied Probability Series. Chapman & Hall.
- Sintomer, I. 2007. La participación ciudadana como tendencia política europea. *Los modelos, proyectos y políticas de participación en grandes y medias ciudades*, 17-34.
- Steffek, J., Kissling, C., Nanz, P. 2008. (eds.), Civil Society Participation in European and Global Governance. *A Cure for the Democratic Deficit?*
- Steinberger, R., Pouliquen, B., Ignat, C. 2005. "Navigating multilingual news collections using automatically extracted information". *Journal of Computing and Information Technology*, Vol. 13, No. 4, pp. 257-264.
- Tan., A.-H. 1999. *Text mining: The state of the art and the challenges*. In Proc of the Pacific Asia Conf on Knowledge Discovery and Data Mining PAKDD'99 workshop on Knowledge Discovery from Advanced Databases, pages 65-70, 1.
- Tikhonov, A., Arsenin, V. 1977. *Solutions of ill-posed problems*. Scripta series in mathematics. Winston. URL <http://books.google.es/books?id=ECrvAAAAMAAJ>

Verba, S., Schlozman, K.L., Brady, H.E. 1995. *Voice and Equality. Civic Voluntarism in American Politics*. Harvard University Press.

Von Winterfeldt, D. y Edwards W. 1986. *Decision Analysis and Behavioral Research*. Cambridge: Cambridge University Press

WfMC. 2011. Workflow Management Coalition. Xpdl tools. <http://www.wfmc.org/standards/xpdl.htm>.

WfMC. Workflow Management Coalition Terminology Glossary. WFMCTC- 1011, Document Status Issue 3.0, 1999.

A. White, S. 2004. *Introduction to BPMN*. Technical report, IBM Corporation.

White, S. 2005. Using BPMN to Model a BPEL Process. *BPTrends*, 3(3):1–18, March.

White, S. A., Miers, P. D. 2009. Guía de referencia y modelado BPMN. *Comprendiendo y utilizando BPMN*.

Witten, I., Frank, E., Hall, M. 2011. *Data Mining: Practical Machine Learning Tools and Techniques: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.

Bibliografía Web

BPEL Specification, <http://docs.oasisopen.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf>, 2006.

BPMN. 2006. www.bpmn.org. Documents.

BPEL4People Specification. http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-bpel4people/BPEL4People_white_paper.pdf, (2005)

OMG. Object Management Group. Business process management initiative. <http://www.bpmi.org/>, (2006)

Business Balls, Business Process Modelling, www.businessballs.com

BPMN 2.0, <http://www.omg.org/spec/BPMN/2.0/> 2011.

GovTalk. <http://cabinetoffice.gov.uk/>

Presupuestos participativos en la ciudad de Getafe, <http://www.getafe.es/>

JBoss Red Hat Division. 2004. Jboss jbpm white paper: <http://jboss.com/pdf/jbpm-whitepaper.pdf>

ITALO. <https://oficinavirtual.mityc.es/>

Participación Bogota, <http://www.participacionbogota.gov.co/>

Singh, D. 2011. 5 Key Benefits of Business Process Modeling. http://www.accuprocess.com/documentation/WP-5_Benefits_of_Business_Process_Modeling.pdf

SOAP, <http://www.w3.org/TR/soap>

Swenson. K. 2006. *The bpmn-xpdl-bpel value chain.* <http://kswenson.wordpress.com/2006/05/26/bpmn-xpdl-and-bpel/>.

UDDI, <http://www.uddi.xml.org>

VVK, <http://www.vvk.ee/voting-methods-in-estonia/engindex/statistics>

WSDL, <http://www.w3.org/TR/wsdl>

XML, <http://www.w3.org/TR/xml>