

Software libre y trabajo colaborativo

Intendencia de Montevideo

Alvaro Rettich

Intendencia de Montevideo
Av. 18 de Julio 1360
Montevideo, Uruguay
alvaro.rettich@imm.gub.uy

Sheila Rahi

Intendencia de Montevideo
Av. 18 de Julio 1360
Montevideo, Uruguay
sheila.rahi@imm.gub.uy

Resumen

Las Tecnologías de la Información y las Comunicaciones (TIC) representan una parte importante del presupuesto de cualquier empresa, tanto en infraestructura como en recursos humanos para los desarrollos de software y su administración. El sector público no escapa a esta realidad y tiene una particularidad: institucionalmente se encuentra fraccionado en múltiples unidades con presupuesto propio. Desde la óptica de un estado integrado, tal separación lleva a una utilización menos eficiente de los recursos informáticos. La hipótesis en la que se basa este trabajo es la posibilidad de una integración mayor en los desarrollos de soluciones de software si se comparten las soluciones existentes dentro del mismo Estado. En este contexto, se analiza la idea de trabajo colaborativo basado en software libre en el Estado, profundizando en el concepto de *trabajo colaborativo* y como es posible disminuir con esta aproximación los costos al mismo tiempo que se logra un mejor servicio al ciudadano. El análisis se realiza desde un enfoque práctico, basado en un trabajo de colaboración que se llevó adelante entre dos instituciones públicas del Uruguay: la Intendencia de Montevideo (IM) y el Instituto de Previsión Social (BPS).

1 Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC) son la plataforma sobre la que se apoyan casi todos los esquemas de negocio en la actualidad, representando una parte importante del presupuesto de cualquier empresa, tanto en lo que tiene que ver con infraestructura como en recursos humanos para los desarrollos de software y su administración. El sector público no escapa a esta realidad y tiene una particularidad: aunque para el ciudadano el Estado es uno solo, institucionalmente se encuentra fraccionado

en múltiples unidades con presupuesto propio. Desde la óptica de un estado integrado, tal separación lleva a una utilización menos eficiente de los recursos informáticos. La hipótesis en la que se basa este trabajo es que es posible una integración mayor en los desarrollos de software si se comparten las soluciones existentes dentro del mismo Estado. En este contexto, el presente artículo analiza la idea de trabajo colaborativo basado en software libre en el Estado. El componente “*desarrollo de software libre en el estado*” forma ya parte de la idea más amplia de Software Público; sin embargo nos parece interesante profundizar en el concepto de *trabajo colaborativo* y como es posible disminuir con esta aproximación los costos al mismo tiempo que se logra un mejor servicio al ciudadano. El análisis se realiza desde un enfoque práctico, basado en un trabajo de colaboración que se llevó adelante entre dos instituciones públicas del Uruguay: la Intendencia de Montevideo y el Instituto de Previsión Social.

El artículo se organiza de la siguiente forma, en la sección 2 se presenta el marco conceptual, destacando y motivando la necesidad del trabajo colaborativo en el estado y sus beneficios tanto para el área de Tecnología de la Información como para el ciudadano. En la sección 3 se muestra como el desarrollo de software libre es una herramienta clave para lograr estos beneficios; en la sección 4 se presenta el sistema de Agenda Electrónica como un caso de éxito de esta modalidad de desarrollo, mostrando desde la práctica como se lograron los objetivos planteados. Finalmente en la sección 5 se presentan las lecciones aprendidas y la sección 6 concluye el artículo.

2 Trabajo colaborativo en el Estado

¿Por qué es importante el trabajo colaborativo en el Estado? O, desde otra perspectiva, ¿Por qué le podría interesar a los responsables de una institución pública, cuyo negocio no es el desarrollo de software, participar en desarrollos de esta naturaleza? Razones como la autonomía de las instituciones, aspectos legales, de escala, de negocio, de presupuesto entre otros, llevan a las instituciones estatales a mantener su propio centro de cómputos y a buscar sus propias soluciones de software. Esta estructura lleva en muchos casos a “reinventar la rueda”, implementando o volviendo a comprar sistemas de software que muchas veces resuelven el mismo problema. El trabajo colaborativo puede ayudar a reducir este esfuerzo redundante, buscando simultáneamente mejorar el servicio a los ciudadanos (evitándole tener que aprender dos veces a hacer lo mismo) y reducir el gasto público.

Una estrategia para lograr esta racionalización de recursos podría ser aplicar una solución de software cerrada a cada organismo que comparta la misma necesidad. Esta estrategia tiene varias desventajas: es difícil lograr que el enfoque único de un producto se adapte totalmente a la realidad de cada negocio, obligando muchas veces a que la realidad se tenga que adaptar al software en cuestión generándose además, una dependencia indefinida con el proveedor, a través de los mantenimientos asociados.

Dado que en muchos escenarios se cuenta con recursos humanos calificados en cada organismo, una alternativa viable (y, hasta donde sabemos, no demasiado explorada en la región) es la modalidad de trabajo colaborativo, donde un organismo que tuvo la necesidad de desarrollar una solución de software para satisfacer una necesidad propia, la deja disponible posibilitando su adopción por terceros y a la vez

brindando la posibilidad de colaborar en su enriquecimiento. De esta forma se puede compartir el desarrollo con el resto del Estado, logrando racionalizar recursos y obteniendo beneficios mutuos.

Mediante el trabajo colaborativo, podemos aprender de la experiencia de otros, ya sea sobre tecnología, sobre el negocio o ambos. La contraprestación a esta colaboración es clara: quienes colaboran inicialmente brindando su experiencia y conocimiento obtienen como ganancia un software con mejoras sin costo, pues las horas de desarrollo las asume quien esta aprendiendo. Para quien acepta utilizar un software desarrollado por otro, la ganancia esta en que contará con un sistema probado en producción con un acumulado de horas de desarrollo en las que no tuvo que invertir. Además de la ganancia material para ambos lados, existe otra ganancia quizás más valiosa a largo plazo: la transferencia de conocimiento.

Para que el software adoptado por un organismo pueda evolucionar con éxito sin la necesidad de una dependencia con quien lo desarrolló, es necesario por un lado contar con el código fuente para poder entender, y mantener el sistema y por otro transferir el conocimiento. De aquí que el uso de software libre en la colaboración juega un papel fundamental como agente habilitador para alcanzar estos beneficios.

3 Desarrollo de software libre en el Estado

El concepto de software público es relativamente nuevo pero parece estar madurando rápidamente en la región. Liderando la experiencia se encuentra Brasil¹, pero también Perú² y en general muchos países de Iberoamerica están siguiendo esta línea y así lo deja ver el “*Consenso de Buenos Aires*”³ firmado el 2 de Julio de 2010 durante la “*XII Conferencia Iberoamericana de Ministros de Administración Pública y Reforma del Estado*”.

En este marco, entendemos que liberar el software desarrollado por el estado es un factor clave para lograr los beneficios de transferencia de conocimiento y racionalización del gasto. Una vez presentados los beneficios, surgen preguntas al respecto de los riesgos de la adopción de software en esta modalidad. Cada vez que los responsables de TI se enfrentan con la situación de tener que incorporar un nuevo sistema, surge por lo menos una preocupación: *¿Cuán dependiente quedo de la tecnología subyacente y/o del proveedor del software y qué costo me implica dicha dependencia?*. Esta interrogante abre un debate sobre los pros y contras del uso de software libre versus software privativo, en el cual no queremos entrar, pues no es el foco de este artículo. Sin embargo, sí es pertinente responder a esta pregunta en el contexto del uso de software desarrollado por otro organismo del estado.

Un gerente de TI podría cuestionar con razón la adopción de este tipo de software, por considerar posible la generación de una dependencia no deseada con otro organismo. Preguntas como: *¿qué soporte tengo? ¿lo puedo utilizar en producción? ¿qué sucede si discontinúan el sistema? surgirán sin lugar a dudas.* Como se mencionó, los dos factores claves para que estas situaciones no sucedan o si suceden no

1 Software público en Brasil: <http://www.softwarepublico.gov.br>

2 Software público en Perú: <http://www.softwarepublico.gob.pe>

3 Consenso de Buenos Aires: <http://www.softwarepublico.gov.br/spb/download/file/consenso.pdf>

afecten a la institución y su autonomía, son el uso de software libre y lograr una transferencia real de conocimiento durante la colaboración.

Al usar software libre, se tiene garantizado el código fuente, por lo cual se tiene total libertad de adaptarlo a las necesidades de la empresa y nunca quedará cautiva de su creador y sus exigencias. Esta es la primer hipótesis del razonamiento. Sin bien, la afirmación anterior es teóricamente válida, es conocido el costo que implica mantener código hecho por terceros si no se tiene la menor idea de su composición. Aquí entra la segunda hipótesis del razonamiento, la *transferencia real de conocimiento*. De esta forma es posible asegurarse que la adopción de un nuevo software pasará por el proceso de comprensión necesario para su mantenimiento y evolución. Sin embargo, el uso de software libre es sólo un habilitador, no hará de por sí que esta transferencia suceda. El encargado de hacer *real* dicha transferencia es el trabajo colaborativo, pues será en esta instancia donde las partes intercambiarán su conocimiento en pos de lograr optimizar los recursos y al mismo tiempo no perder su autonomía.

Para ilustrar los conceptos presentados, en lo que resta de este artículo se presenta el sistema de Agenda Electrónica y cómo fue llevada a cabo la colaboración entre dos organismos del estado, buscando mostrar los hechos en los que se basa el presente análisis para entender este tipo de experiencia como algo positivo para el estado y la sociedad toda.

4 Un caso de éxito

La iniciativa de liberar el código fuente de un sistema de Agenda ya desarrollado en la Intendencia de Montevideo surge como una decisión política por parte de la dirección de la División Tecnología de la Información. Su objetivo fue promover la transferencia de conocimiento hacia la comunidad y fomentar el uso de software libre en el estado y su sociedad. Esta decisión abrió la puerta a una experiencia de trabajo colaborativo que culminó con una serie de logros para las instituciones participantes. En lo que sigue de esta sección se explica brevemente de que trata el sistema de agenda electrónica, para luego presentar una reseña histórica de lo acontecido y así poder entender desde la práctica los conceptos presentados anteriormente sobre los beneficios del trabajo colaborativo basado en software libre.

4.1 Agenda electrónica

La Agenda Electrónica (SAE) es un sistema integral para la gestión de reserva de horas para trámites, permitiendo la reserva de horas en línea a través de la web, contando además con servicios para ser consumidos por otros tipos de clientes como IVR o SMS. Dentro de sus funcionalidades se destaca el nivel de configuración en tiempo de ejecución de todos los elementos vinculados al caso de uso "*Realizar una reserva*" como son textos, ingreso de datos y validaciones. También se destaca el módulo que permite la gestión de la atención a los ciudadanos el día para el cual tienen fijada una reserva. Este módulo cuenta con una pantalla para cada puesto de atención y un llamador para advertir al cliente que está siendo llamado y a que puesto debe dirigirse. Cabe mencionar que se trata de una aplicación 100% web desarrollada en Java y

basada en el estándar JEE.

Un aspecto crucial a la hora de su aceptación por terceros, es su arquitectura: el sistema funciona en realidad como un servidor de agendas. Cada oficina de la institución cuenta con sus propias agendas para los trámites que así lo requieran. Cada agenda que se configura en el sistema vive aislada, en su propio espacio de trabajo y solo los usuarios con roles para dicha agenda podrán gestionarla. La capacidad de servir múltiples instancias de agenda sumada a la capacidad de total configuración en tiempo de ejecución, sumada a la interfaz de configuración orientada a un usuario final, hacen que crear, configurar y poner en funcionamiento una nueva agenda sea cuestión de minutos.

Un ejemplo vale más que mil palabras, en <http://www.montevideo.gub.uy> bajo la sección “Servicios online” ingresando en el enlace “Reserva de horas para trámites”⁴ se encuentran los trámites y servicios actualmente disponibles bajo esta modalidad de reserva en línea. A modo de ejemplo, en la Figura 1 puede verse el primer paso de la reserva para una de estas agendas.

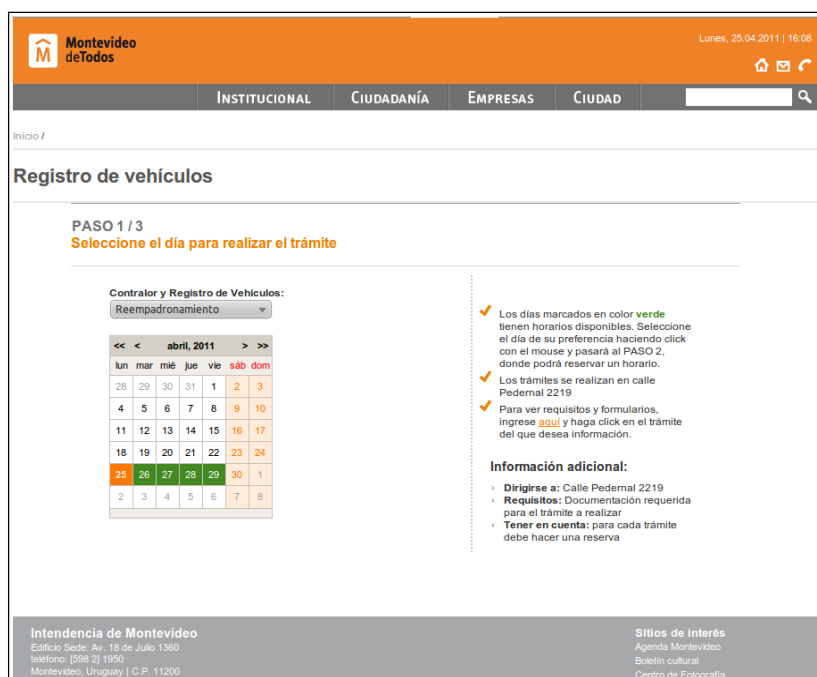


Figura 1: Primer paso de una reserva para el registro de vehículos

4 <http://www.montevideo.gub.uy/institucional/tramites-y-servicios/reserva-de-hora-para-tramites>

Como se a mencionado, el mismo sistema se encuentra actualmente en producción en otra institución pública, en <http://www.bps.gub.uy/> bajo la sección “Servicios en línea” ingresando al enlace “Agenda Web”⁵ se encuentra la Agenda Electrónica alojada en otro centro de cómputos y configurada para los trámites de dicha institución. En la Figura 2 se puede ver el primer paso para la reserva de uno de sus trámites. ¿Se observa la similitud?... y ¿las diferencias?

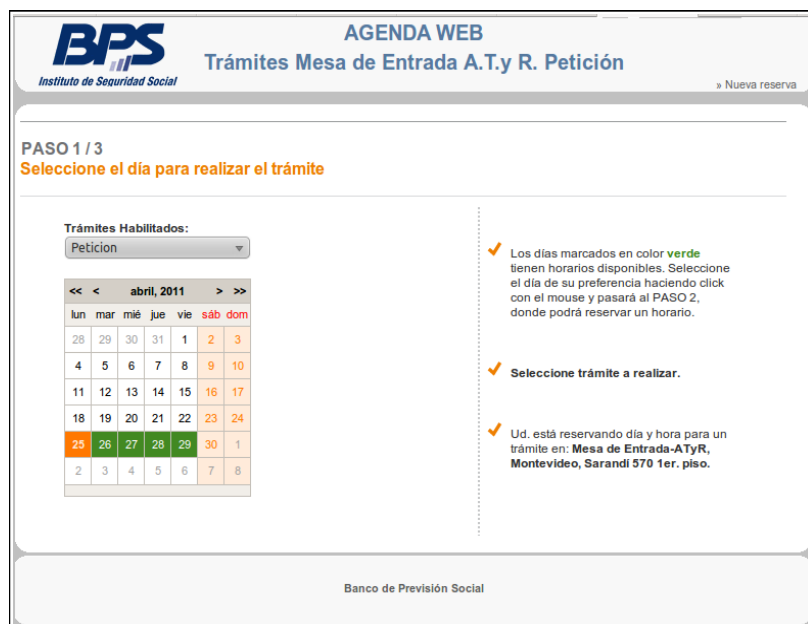


Figura 2: Primer paso de una reserva para trámites del BPS

4.2 Liberación del código fuente

La liberación del código fuente se lleva a cabo cuando el software ya estaba en producción en su primer versión desde hacía algunos meses. La primer liberación consistió en empaquetar todos los proyectos que componían el sistema, adjuntar un pequeño manual y dejarlo disponible en el sitio web de la Intendencia de Montevideo para su descarga. El único cambio que se le hizo al código fuente fue la incorporación de la licencia GPLv3⁶.

La segunda liberación sucede unos meses después cuando, al empezar a recibir las primeras preguntas sobre como instalar, utilizar y por sobre todo al querer publicar correcciones del sistema, se decide abrir un proyecto en SourceForge⁷ donde poder publicar los fuentes, los binarios, documentación y contar con un repositorio público para el versionado del sistema. Aquí se abre la puerta para una colaboración real, es decir un ida y vuelta, un desarrollo colaborativo sobre una versión común del sistema. Por más información acerca del sistema puede acceder al sitio del proyecto en SourceForge:

<http://sourceforge.net/projects/imm-sae>

5 <http://www.bps.gub.uy/ServiciosEnLinea/AgendaWeb.aspx>

6 Para conocer más acerca de la licencia GPLv3 ver <http://www.gnu.org/licenses/gpl-3.0.html>

7 SourceForge: <http://sourceforge.net>

4.3 Colaboración entre instituciones

Todo comienza cuando el BPS⁸, en la búsqueda de un producto para la gestión de reservas, encuentra la publicación de este software en Internet. Iniciándose con una serie de consultas vía correo electrónico de sus técnicos para con los nuestros, los intercambios van creciendo hasta transformarse en una colaboración oficial entre ambas instituciones.

Por tratarse solamente de dos actores, la modalidad de colaboración consistió en reuniones de trabajo periódicas donde los técnicos del BPS planteaban sus dudas y nuevos requerimientos, dado que ambas partes conocían el código fuente y por lo tanto, conocían como estaba construido el software; en poco tiempo estas reuniones pasaron de la modalidad “pedido de ayuda” a “desarrollo en conjunto”.

Además de las reuniones de trabajo y el correo electrónico como medios de comunicación, la clave fue trabajar bajo la consigna “*versión única*”. Esta consigna, acordada desde el comienzo, implica que cada institución desiste de crear una versión propia del software: todo cambio deberá ser parte de una solución única, y cada personalización necesaria deberá ser configurable, incluyendo una opción por defecto que pueda publicarse. Esto fue estampado desde el comienzo de la colaboración como un principio de arquitectura del sistema.

Trabajando de esta forma, en poco tiempo se lograron mejoras genéricas que ambos tomaron y quedaron disponibles para el resto de la comunidad y también se implementaron funcionalidades dependientes del negocio de cada institución. Un ejemplo simple, pero claro, consiste en determinar si un día es feriado para la institución, de forma de no permitir reservas para ese día. Claramente esta lógica es dependiente de cada institución. La versión disponible en SourceForge viene con una implementación *naive* por defecto donde solo los sábados y domingos son considerados no laborables. En la práctica cada organismo tiene una implementación propia de esta lógica, la que incorpora al sistema cambiando el valor de una propiedad en un archivo de configuración.

4.4 Algunos números

El proyecto Sistema de Agenda Electrónica inició y terminó en la Intendencia de Montevideo en un lapso aproximado de un año, mientras que al personal técnico del BPS le tomó unos pocos meses poner en producción el sistema adaptado a su realidad.

Para la Intendencia de Montevideo, las horas consumidas en la colaboración se corresponden a las reuniones de trabajo semanales y algunas horas dedicadas al diseño de las partes del sistema que necesitaban ser personalizadas y aún no lo eran. Estas pocas horas de colaboración redituó, en lo inmediato, en reporte de errores y algunas mejoras en la funcionalidad como algún tipo de optimización. Para el BPS la ganancia fue clara, obtuvieron un sistema sin costo, pues el costo insumido es comparable al costo de instalación y aprendizaje que cualquier sistema privativo siempre requiere, con la ventaja de que el

8 BPS: Instituto de Seguridad Social

personal se queda con el conocimiento del sistema y lo más importante, manteniendo independencia del proveedor.

4.5 En qué estamos

A la fecha el sistema se encuentra en producción en la IM y el BPS. El Banco de Seguros del Estado ya está trabajando en su adaptación para utilizarlo en varios de sus trámites, mientras que otros organismos del estado se encuentran evaluando su incorporación.

5 Lecciones aprendidas

A lo largo de este artículo se han expuesto los beneficios y razones por las cuales se entiende positivo y necesario el apoyo a este tipo de iniciativa, sin embargo también se han aprendido algunas lecciones que nos parece importante compartir en pos de mejorar la experiencia de colaboración y facilitar la adopción de software público. Antes que nada, para embarcarse en este tipo de iniciativas con chance de éxito se debe estar seguro de poder cumplir una serie de requisitos previos los que se pueden agrupar en factores humanos y factores técnicos.

Dentro de los factores humanos, por una parte debe de existir apoyo político de la gerencia de ambos organismos; si bien esto parece obvio, muchas veces tras la idea de software libre hay una sensación de informalidad y algo de fanatismo. Por el contrario, para llevar adelante una colaboración efectiva, debe estar claro desde el inicio, las responsabilidades y recursos con los que se debe contar desde ambas partes.

El otro factor humano refiere a que los técnicos participantes se sientan comprometidos con la causa. Dado que este tipo de colaboración se basa en el aprendizaje, y todos sabemos lo difícil que es aprender del código de otro, es necesario generar conciencia de equipo.

Yendo a los factores técnicos, en términos generales trabajar en colaboración con software libre no es simplemente "*Rigth click, add to zip and publish*" del código fuente. Este código y su diseño será el lenguaje de comunicación entre las partes, por lo tanto si su diseño no está bien cuidado, desde la arquitectura hasta el nombrado de los artefactos, será muy difícil que un tercero llegue a apropiarse del código fuente. Por supuesto, este tipo de requerimiento es deseable en cualquier sistema para extender su ciclo de vida, pero en este contexto adquiere particular importancia.

Otro factor técnico refiere a *pensar en genérico*: de nada sirve un sistema que resuelva un problema común a otras instituciones pero que haya sido pensado exclusivamente para una institución. Para situarnos, en la jerga de la Intendencia de Montevideo tiene sentido la idea de servicio o unidad para referirse a una oficina puntual que se encarga de determinados servicios a la comuna; para el BPS estos nombres pueden no tener sentido. Evitar este tipo de diseño es clave para que el software sea adaptable a otras realidades: *al problema lo que es del problema*.

El sistema debe estar, además, desacoplado de la realidad de negocio puntual del organismo. Es clara la necesidad de que el sistema debe comunicarse con otros sistemas corporativos, pero lo debe hacer siguiendo patrones de diseño y estándares de comunicación que permitan que dicha lógica pueda ser intercambiada según la realidad de cada institución. Si bien este parece ser un requerimiento innecesario para un software de uso interno, en realidad es una práctica sana y recomendable en ingeniería de software, el contar con artefactos con *alta cohesión y bajo acoplamiento*. En el caso de la agenda electrónica este requerimiento no funcional se tomó en cuenta desde el inicio sin pensar en que sería liberada, sino más bien pensando en el mantenimiento y ciclo de vida del sistema, pero fue una de las claves para que su adopción por terceros fuera posible.

Todos estos factores se dieron en el proyecto de colaboración IM-BPS y es por eso que los resumimos como lecciones aprendidas. Desde la experiencia vivida, consideramos que estos factores son condición necesaria para una colaboración exitosa. En particular, es necesario tener en cuenta los factores técnicos desde el inicio de un proyecto, pues si más tarde se decide liberarlo y estos no fueron tenidos en cuenta, difícil será lograr una colaboración.

6 Conclusión

A nuestro entender el aporte principal de este artículo es transmitir desde un enfoque práctico como el trabajo colaborativo y el software libre pueden combinarse para lograr beneficios en el estado. Tanto la racionalización del gasto público en lo referente a Tecnologías de la Información como el aporte de una visión integradora mejorando el servicio al ciudadano, son beneficios cuantitativos que refuerzan el concepto de Software Público.

Así mismo, el análisis de esta experiencia deja como resultado, además, una serie de lecciones aprendidas que pueden ser de utilidad para quienes decidan transitar este camino en el futuro. Estas lecciones pueden verse como requerimientos necesarios para lograr los beneficios presentados. Primero tenemos el factor humano, debe existir apoyo político de la gerencia de los organismos participantes, así como también, los técnicos participantes deben sentirse comprometidos con la causa, debido a que es difícil aprender del código de otro, es necesario generar conciencia de equipo. Por otra parte tenemos los factores técnicos, dado que el código fuente es el lenguaje de comunicación entre las partes, su diseño debe estar bien cuidado, desde la arquitectura hasta el nombrado de los artefactos, de lo contrario será muy difícil que un tercero llegue a apropiarse de este código.

Un concepto fundamental del diseño de la aplicación refiere a *“pensar en genérico”*, de nada sirve un sistema que resuelva un problema común a otras instituciones pero que haya sido pensado exclusivamente desde el modelo de negocio de una institución en particular. Así mismo, la necesidad de comunicarse con el sistema corporativo debe satisfacerse siguiendo patrones de diseño y estándares de comunicación que apunten a tener artefactos con *alta cohesión y bajo acoplamiento*, permitiendo un sistema desacoplado de la realidad del negocio, dando la posibilidad de sustituir la lógica particular a la realidad de cada institución. En el caso de la Agenda Electrónica estos factores se tomaron en cuenta desde el inicio pensando en el

mantenimiento y ciclo de vida del sistema, pero fue una de las claves para que su adopción por terceros fuera posible.

Con respecto a la etapa de colaboración el factor técnico crucial fue mantener durante toda la colaboración la premisa de “*versión única*”, lo que sin lugar a duda, sumado a los otros factores fue el catalizador de lo que consideramos un caso de éxito.

También nos interesa dejar algunas reflexiones. Todo esto no hubiera sido posible sin el compromiso de las partes bajo un mismo lema “*trabajar juntos*” y por supuesto el hecho de contar con las libertades necesarias sobre el código fuente. Otro resultado no tan visible es la transferencia de conocimiento que puede darse entre las instituciones públicas y hacia la sociedad, el software libre es un recurso educativo.

Para terminar, solo queda decir que es posible esta modalidad de trabajo, tiene sus retos, pero también sus beneficios tanto para el estado como para la sociedad, y bien encausada sin duda los beneficios son superiores a los costos en los que se pudiera incurrir.