

Construyendo la Administración electrónica con Software Libre, gnuINE una aplicación libre para la administración local

**Pedro A. Gracia
Zuzen de la Cueva**

Copyright (C) 2002 Pedro A. Gracia y Zuzen de la Cueva. Permitida la redistribución ilimitada de copias literales, la traducción del texto a otros idiomas y conversión a otros formatos, siempre y cuando se mantenga esta autorización y la nota de copyright.

gnuINE es una aplicación informática para la gestión del Padrón Municipal de Habitantes. Es una aplicación construida con Software Libre para la Administración Pública Local.

gnuINE es una aplicación modular que permite la gestión del Padrón a cualquier ayuntamiento sin pagos de licencias y con acceso al código fuente del mismo según la filosofía de desarrollo que impulsa el movimiento de software libre. La tecnología involucrada en el desarrollo es la más vanguardista y auna servidores de bases de datos SQL, interfaces gráficas diversas y gestión remota del sistema informático a un coste cercano a cero para los ayuntamientos.

Tabla de contenidos

1. Introducción	2
2. ¿Qué es Software Libre?	3
3. La reutilización sistemática de código	4
4. Fácil adaptabilidad por medio de la modificación de código existente	5
5. El modelo de desarrollo cooperativo de software	5
6. La libre distribución sin el pago de licencias, reduciendo la brecha digital	6
7. El fortalecimiento del capital humano técnico europeo	6
8. La reducción/eliminación de la fuga de capitales hacia fuera de la Unión	7
9. La "estandarización" de recursos de uso colectivo definibles como infraestructura básica de la Sociedad de la Información europea	7
10. La facilidad de internacionalización de las aplicaciones (i18n)	8
11. gnuINE un caso de estudio. El uso del Software libre para la Administración electrónica	9
11.1. Objetivo o finalidad del proyecto	9
11.2. Objetivo estratégico	9

11.3. La normativa aplicable al proyecto gnuINE	9
11.4. Resultados esperados	9
11.5. Descripción técnica y especificaciones funcionales	10
11.6. El estado actual	12
11.7. Metodología de desarrollo y plan de trabajo	12
11.8. gnuINE se libera bajo la GPL Licencia General Pública de la FSF	13
11.9. Tecnologías y recursos mas significativos incorporadas en el proyecto	13

1. Introducción

El software libre esta empezando a captar la atención y el interés de forma significativa de la Administración Pública en la Europa del siglo XXI. Se comienza a integrarlo en las líneas de acción de I+D, se promueven nuevos proyectos, se incorpora a los planes tecnológicos, se comienza a legislar en algunos Estados de la Unión Europea.

El software libre ya empieza a estar presente en todos los niveles de la Administración: europeo, estatal, regional y local. Este auge del software libre en la Administración Pública obedece a la percepción cada vez más nítida de su potencial como herramienta estratégica para el desarrollo de la sociedad de la información europea. Y especialmente para la construcción de la nueva Administración electrónica.

El potencial del SL como herramienta estratégica viene determinado por los siguientes factores fundamentales, entre otros:

1. la reutilización sistemática de código
2. fácil adaptabilidad por medio de la modificación de código existente
3. el modelo de desarrollo cooperativo de software
4. la libre distribución sin el pago de licencias, reduciendo la brecha digital
5. el fortalecimiento del capital humano técnico europeo
6. la reducción/eliminación de la fuga de capitales hacia fuera de la Unión
7. la "estandarización" de recursos de uso colectivo definibles como infraestructura básica de la Sociedad de la Información europea
8. la facilidad de internacionalización de las aplicaciones (i18n)

A lo largo de esta comunicación vamos a exponer sus implicaciones y potencialidades utilizando como referencia y caso de estudio la aplicación gnuINE que estamos desarrollando en estos momentos. gnuINE es una aplicación informática para la gestión del Padrón Municipal de Habitantes. Es una aplicación construida con Software Libre para la Administración Pública Local.

gnuINE es una aplicación modular que permite la gestión del Padrón a cualquier ayuntamiento sin pagos de licencias y con acceso al código fuente del mismo según la filosofía de desarrollo que impulsa el movimiento de software libre. La tecnología involucrada en el desarrollo es la mas vanguardista y auna servidores de bases de datos SQL, interfaces gráficas diversas y gestión remota del sistema informático a un coste cercano a cero para los ayuntamientos.

2. ¿Qué es Software Libre?

Esos factores son posibles y existen como tales en función de las llamadas cuatro libertades que proporciona el Software Libre (SL) a los usuarios/desarrolladores, que han sido definidas por la FSF (Free Software Foundation <http://www.fsf.org/philosophy/free-sw.es.html>) y recogidas en la Licencia General Pública GPL. Dicen así:

El "Software Libre" es un asunto de libertad, no de precio. "Software Libre" se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

- La libertad de usar el programa, con cualquier propósito (libertad 0)
- La libertad de estudiar como funciona el programa, y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2).
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3). El acceso al código fuente es un requisito previo para esto.

Un programa es software libre si los usuarios tienen todas estas libertades. Así pues, deberías tener la libertad de distribuir copias, sea con o sin modificaciones, sea gratis o cobrando una cantidad por la distribución, a cualquiera y a cualquier lugar. El ser libre de hacer esto significa (entre otras cosas) que no tienes que pedir o pagar permisos.

También deberías tener la libertad de hacer modificaciones y utilizarlas de manera privada en tu trabajo u ocio, sin ni siquiera tener que anunciar que dichas modificaciones existen. Si publicas tus cambios, no tienes por que avisar a nadie en particular, ni de ninguna manera en particular.

La libertad para usar un programa significa la libertad para cualquier persona u organización de usarlo en cualquier tipo de sistema informático, para cualquier clase de trabajo, y sin tener obligación de comunicárselo al desarrollador o a alguna otra entidad específica.

La libertad de distribuir copias debe incluir tanto las formas binarias o ejecutables del programa como su código fuente, sean versiones modificadas o sin modificar (distribuir programas de modo ejecutable es necesario para que los sistemas operativos libres sean fáciles de instalar). Esta bien si no hay manera de producir un binario o ejecutable de un programa concreto (ya que algunos lenguajes no tienen esta capacidad), pero debes tener la libertad de distribuir estos formatos si encontraras o desarrollaras la manera de crearlos.

Para que las libertades de hacer modificaciones y de publicar versiones mejoradas tengan sentido, debes tener acceso al código fuente del programa. Por lo tanto, la posibilidad de acceder al código fuente es una condición necesaria para el software libre.

Para que estas libertades sean reales, deben ser irrevocables mientras no hagas nada incorrecto; si el desarrollador del software tiene el poder de revocar la licencia aunque no le hayas dado motivos, el software no es libre.

3. La reutilización sistemática de código

El acceso al código fuente de miles de aplicaciones y programas proporciona la posibilidad de reutilizar miles de líneas de código escritas por otros. De forma que es factible y habitual encontrar código escrito que cubra una necesidad para un programa dado. Esta es una característica que define y diferencia al SL.

Una de las principales fortalezas del software libre es su factor acumulativo. Esto reduce los tiempos de desarrollo y mejora la calidad final del código propio al incorporar código de calidad evitando la necesidad de volver a descubrir "el Mediterráneo" cada vez que se programa. En la medida en que se está utilizando código testeado que ha sido revisado y depurado.

La sistemática reutilización de código como práctica habitual entre la Comunidad de desarrolladores de SL es una de las razones que explica el vertiginoso avance producido en los 20 años de su existencia.

En el caso de gnuINE esto ha permitido reducir el número de desarrolladores necesarios para su construcción y ha proporcionado la solidez y robustez de código ensayado. gnuINE es posible porque utiliza, integra e interactúa con miles de líneas de código escritas por miles de desarrolladores de todo el mundo.

En definitiva la reutilización sistemática de código es una de las consecuencias del acceso al código fuente y supone un elemento de racionalidad en el uso de recursos escasos. Mejora la calidad final del código propio, reduce el tiempo de desarrollo, elimina la redundancia en el esfuerzo de producción. Evita la reinención sistemática y continúa de la rueda, mejora la rueda preinventada.

4. Fácil adaptabilidad por medio de la modificación de código existente

Otra consecuencia del acceso al código fuente y a la posibilidad de su modificación sin restricciones, es su adaptabilidad y capacidad de "pegarse al terreno".

Una ventaja innegable es el hecho de poder ajustar un programa o aplicación a las especiales y particulares necesidades de un usuario dado. Usando gnuINE, por ejemplo, cuando se libere la primera versión de producción de la aplicación, un responsable de informática de un ayuntamiento podrá decidir que le interesa que gnuINE se comporte de una forma distinta a la que lo hace.

Simplemente tendrá que modificar el código de forma que se adapte a las necesidades especiales de su ayuntamiento. Adaptabilidad y flexibilidad son características diferenciadoras del SL. Que además posibilitan el enriquecimiento continuo por medio de la incorporación de nuevo código escrito para cubrir necesidades específicas de los usuarios. Se activa entonces otra de las fortalezas de SL, la retroalimentación continua y la participación activa de los usuarios y desarrolladores en la evolución de un programa dado. El llamado modelo de desarrollo cooperativo de software.

5. El modelo de desarrollo cooperativo de software

La clave del éxito y de la expansión del SL reside en el modelo de desarrollo cooperativo utilizado desde su génesis. Hay quien dice que este modelo ha significado la primera acción humana colectiva y consciente de ámbito planetario de la historia. Dos elementos lo han hecho posible: la red mundial Internet y el afán de miles de desarrolladores de software por compartir sus conocimientos y experiencias.

El modelo de desarrollo que ha cristalizado en la Comunidad de Software Libre se basa en la articulación de metodologías de trabajo que potencian la participación mancomunada de programadores dispersos por todo el planeta para la elaboración colectiva de programas informáticos libres. Esa metodología ha permitido la gigantesca tarea, realizada en los últimos 20 años, de crear desde "cero" un Sistema Operativo Libre completo que substituye a cualquier Sistema Operativo de propietario.

Ha sido ese modelo el que ha permitido que miles de programadores apliquen sus conocimientos y esfuerzos para la resolución de los problemas que plantea el diseño, la depuración y la corrección del código fuente de esos programas para alcanzar los excepcionales niveles de calidad y estabilidad que hoy tienen. Modelo que asegura su evolución continua con un factor exponencial por acumulación.

Eric S. Raymond lo ha expresado magistralmente en su ya clásico ensayo "La Catedral y el Bazar" de la siguiente forma: "A lo mejor al final la cultura del software libre triunfará no porque la cooperación sea moralmente correcta o porque el atesoramiento del software sea moralmente incorrecto (suponiendo que uno crea lo anterior, cosa que no hacemos ni Linus ni yo), sino simplemente porque el mundo comercial no puede ganar la carrera contra las comunidades de desarrollo libre que ponen varios ordenes de magnitud más horas-hombre en un problema." ensayo cuya lectura recomendamos, se puede obtener en:

<http://www.eitig.com/linux/gnu/catedral.html>

Esa es la clave, la capacidad de acumular gigantescas cantidades de esfuerzo a un problema. El modelo bazar para el desarrollo de software.

6. La libre distribución sin el pago de licencias, reduciendo la brecha digital

Distribuir libremente, sin restricciones y sin tener que pagar licencias, sitúa el coste de uso del software en su lugar. Nadie puede negar hoy que el problema al que realmente nos enfrentamos es la llamada "brecha digital". La brecha tecnológica que supone la capacidad de uso de la población frente a las posibilidades y potencialidades que hoy están a su alcance.

La reducción y eliminación de esa "brecha digital" es una necesidad para el desarrollo de la Sociedad de la Información y el uso intensivo de la Administración electrónica. No sirve de nada poner a disposición de la población recursos que no puede usar, ni mantener funcionando. El coste artificial introducido por el pago de licencias debe ser eliminado, el SL permite centrar los esfuerzos en la formación, en el mantenimiento y en el uso.

Una distribución libre del software permite activar y poner en juego el factor meritocrático. Prueba, compara y elige. La calidad, usabilidad, pertinencia y adecuación a las necesidades frente a ilusión mediática generada por el marketing y la publicidad.

La disponibilidad del código fuente y la libre distribución posibilitan la selección "natural" en función de la calidad y solidez del software. Y reducen significativamente los costes a soportar por los usuarios. Facilitando la focalización del coste hacia la formación, mantenimiento y uso, imprescindibles para reducir la brecha digital.

Que gnuINE sea distribuido libremente facilita que sus posibles usuarios lo prueben, utilicen, mejoren y modifiquen. Permite que se impliquen en el proceso de evolución directamente. Si algo no se ajusta a tus necesidades, cámbialo y redistribuye los cambios para que otro ayuntamiento se beneficie de tu experiencia y esfuerzo.

El no pagar por licencias "igual"; es lo mismo que un ayuntamiento tenga uno o cien ordenadores donde utilizar gnuINE. Su coste en licencias es el mismo: ninguno.

7. El fortalecimiento del capital humano técnico europeo

El uso de SL para la construcción de la Administración electrónica posibilita el fortalecimiento y desarrollo de una comunidad de técnicos local. Proporciona la ventaja comparativa de la cercanía y proximidad de los programadores a la realidad cotidiana que debe resolver eficazmente el software que soporte esa Administración electrónica.

Una de las carencias sistemáticamente señaladas desde la Comisión Europea es la de técnicos cualificados. Se estiman en más de setecientos mil nuevos técnicos, los necesarios para cubrir las necesidades mínimas actuales en el marco de la Unión. Esto supone que es prioritario la promoción, apoyo y el estímulo para la formación y desarrollo de una comunidad técnica fuerte, capacitada y eficiente.

El hecho de la disponibilidad del código fuente que permite un conocimiento exhaustivo del funcionamiento de un programa y que favorece sensiblemente la capacidad de formación, el mantenimiento y el desarrollo del mismo por cualquier técnico. La inversión en SL supone automáticamente la inversión en capital humano autóctono y asegura además la no dependencia de agentes externos. Lo que evidentemente fortalece la capacidad competitiva de la industria del software local.

8. La reducción/eliminación de la fuga de capitales hacia fuera de la Unión

La inversión de capitales en el SL reduce y casi elimina la fuga de los mismos fuera de la Unión Europea. Esto supone que al no tener que pagar por licencias de software se pueden destinar esos recursos al desarrollo, la formación y el mantenimiento. Lo que redundará en retroalimentar la industria local de software.

La gigantesca cantidad de capital que hoy gasta la Administración por el pago de licencias podría ser invertido en propiciar un salto cualitativo en el desarrollo de la Administración electrónica. Con costes de desarrollo mucho menores, creando una infraestructura técnica mucho más sólida y flexible.

gnuINE tiene un coste para su construcción de menos de 60.000 euros, lo que supone en la práctica (teniendo en cuenta el número de usuarios potenciales, los ayuntamientos) menos de seis euros por ayuntamiento. Que trasladado al número de habitantes se convierte en cercano a cero.

Lo realmente importante es la necesidad de un colectivo humano capaz de proporcionar mantenimiento y soporte que indefectiblemente tiene que ser local. No se puede depender de agentes externos para el funcionamiento de sistemas, que hoy, son vitales. La consecución de una red de empresas y profesionales cualificados que proporcionen servicios de valor añadido de asistencia y mantenimiento es un prerrequisito para la construcción de la Administración electrónica.

9. La "estandarización" de recursos de uso colectivo definibles como infraestructura básica de la Sociedad de la Información europea

El SL propicia la "estandarización" de recursos de uso colectivo, en la medida en que se tiene acceso al código fuente, se estimula la adopción de estándares de facto. Que son susceptibles de convertirse en estándares oficiales.

El quid de la cuestión reside en la definición de qué es infraestructura mínima, ¿cuál es la Base Tecnológica Común (BTC)? Hoy en día con el vertiginoso avance que estamos viviendo cada vez engloba más cosas esa definición de mínimos. Nadie duda que el acceso ágil y eficiente a la Red y a sus servicios es una "necesidad" básica, por ejemplo. Hoy están limitados a las grandes organizaciones o empresas, pero los servidores conectados permanentemente a la Red de PYMES y ciudadanos particulares son un escenario cada vez más cercano.

La Comunidad de software libre es una vanguardia en este terreno y está demostrando, hoy que se puede hacer con "relativa" facilidad. Y conforme avanza el SL cada día se va haciendo un poco más sencillo para simples usuarios. La reutilización y reciclaje de miles de máquinas "viejas" para funcionar como servidores locales es una realidad que ya está trascendiendo a la Comunidad de Software Libre.

El proyecto Linux counter <http://counter.li.org> proporciona una referencia del uso y destino que los usuarios informan sobre sus máquinas aportando información sobre las características del software que está corriendo en ellas y las funciones que realizan

La construcción de la Administración electrónica necesita elevar el nivel de estandarización de forma que sea más fácil y sencillo la interconexión e interacción de los recursos que se implementen. Que las aplicaciones y programas de uso en la Administración estén contruidos con SL, de los que se dispone del código fuente, facilita sensiblemente esa estandarización, puesto que todo el mundo tiene la posibilidad de estudiar y adecuar sus programas para que interactúen y funcionen eficazmente.

Una de las ventajas de ese proceso de estandarización consiste en la posibilidad de elevar el nivel de interlocución entre aplicaciones y servidores. Es previsible (a parte de deseable) que en un futuro cercano las PYMES y ciudadanos alcancen el nivel de posesión y uso de servidores locales que les provean de conexión permanente a la red, que hagan funciones de cortafuegos y les proporcionen servicios básicos de la red (correo, web, news, etc).

Esto permitirá a la Administración electrónica implementar servicios y recursos que interroguen, interactúen y se interrelacionen con los servidores de PYMES y ciudadanos aumentando exponencialmente la capacidad de explotación intensiva de recursos informáticos. Reduciendo costes de explotación, acelerando procesos, agilizando en definitiva la relación A2B y A2C. El papel del software libre en ese entorno es evidentemente preponderante y nos atrevemos a decir que imprescindible.

10. La facilidad de internacionalización de las aplicaciones (i18n)

Otra de las fortalezas del SL es su capacidad para los procesos de localización e internacionalización del software. Hoy en día es extremadamente sencillo acometer cualquier proceso de traducción y localización de un software dado.

Esto se consigue separando el código fuente de los mensajes de la aplicación y usando herramientas estándar tipo `gettext` para su manipulación, con lo cual tenemos la ventaja añadida de que con un coste reducido (limitado a la traducción) podemos presentar una aplicación dada a los usuarios en su idioma materno con agilidad y rapidez.

11. gnuINE un caso de estudio. El uso del Software libre para la Administración electrónica

11.1. Objetivo o finalidad del proyecto

El objetivo principal de este proyecto es que todos los ayuntamientos de España dispongan de un programa informático para la gestión del Padrón Municipal de Habitantes desarrollado en el modelo Software Libre bajo licencia GPL.

11.2. Objetivo estratégico

Que las corporaciones locales dispongan de una herramienta para la gestión del Padrón de Habitantes que puedan personalizar y usar al menor coste posible.

Y que permita que por medio de la participación colectiva de los usuarios se vaya modificando y mejorando el código para obtener la más ajustada y versátil herramienta posible.

Como resultado final se espera lograr una comunidad de usuarios y desarrolladores que mantengan actualizado y mejoren continuamente la aplicación. Siguiendo el modelo de desarrollo cooperativo de software donde la participación e implicación hacen posible acortar y mejorar los procesos de depuración y expansión de un programa dado.

11.3. La normativa aplicable al proyecto gnuINE

GnuINE se basa principalmente en la Ley 4/1996 de 10 de enero, por la que se modifica la Ley 7/1985, de 2 de abril, Reguladora de las Bases del Régimen Local, en relación con el Padrón municipal.

También sigue el Real Decreto 2612/1996 de 20 de diciembre, por el que se modifica el Reglamento de Población y Demarcación Territorial de las Entidades Locales aprobado por el Real Decreto 1690/1986, de 11 de julio.

Las instrucciones técnicas para la gestión y revisión del Padrón Municipal la dicta la Resolución de 9 de abril de 1997 de la Subsecretaría, por la que se dispone la publicación de la Resolución de 1 de abril, de la Presidenta del Instituto Nacional de Estadística y del Director General de Cooperación Territorial.

Se está teniendo en cuenta también la Orden de 11 de julio de 1997 sobre comunicaciones electrónicas entre las Administraciones públicas referentes a la información de los Padrones municipales.

11.4. Resultados esperados

- Un programa de gestión del Padrón de Habitantes, multiplataforma (GNU/Linux y MS Windows 9x) que los ayuntamientos puedan "bajarse" directamente de Internet para su instalación y uso sin ningún tipo de costes añadidos ni pagos de licencias.
- Manual de usuario de la aplicación. La documentación facilitada al usuario final marcará sin duda el éxito del proyecto. Este manual constituye una parte fundamental de la "usabilidad" del programa. Las fuentes del manual formarán parte del proyecto para que pueda ser fácilmente mantenido.
- Notas de implementación de la aplicación. Para facilitar la entrada de desarrolladores, la depuración del código del programa y el mantenimiento del mismo, este proyecto contará con la documentación del análisis y diseño del mismo. También se irá reflejando los avances y cambios en un *changelog*.
- Validación en ayuntamientos. Pruebas piloto que permitan validar la aplicación y pulirla para alcanzar el estado de explotación operativa.

11.5. Descripción técnica y especificaciones funcionales

Este programa está concebido para su utilización en GNU/Linux y en MS Windows 9x, y está siendo desarrollado en Python.

La aplicación es cliente y servidor, pero el servidor de bases de datos puede encontrarse en el mismo ordenador del usuario de la aplicación o en un servidor independiente. El servidor de bases de datos SQL utilizado para el desarrollo inicial es el PostgreSQL.

La aplicación tiene actualmente dos interfaces: la gráfica y la de línea de comandos. El código fuente esta publicado en *Sourceforge*. Para mas detalles, ver la pagina en <http://www.sourceforge.net/projects/padhab>.

La forma en la que está estructurada la aplicación permite la publicación de la información en web y su interconexión con otros programas y aplicaciones a través de drivers ODBC o similares.

Además es fácilmente integrable con programas para PDAs para ayudar a los agentes censales, consultas por web del tipo intranet, etc.

En cuanto a las especificaciones funcionales podemos decir que la aplicación responderá a las siguientes necesidades:

- Adaptado a la normativa padronal vigente.
- Permitirá: altas, bajas, rectificaciones, consultas,...

- Mantenimiento en línea de los movimientos, historia de la persona.
- Explotaciones retrospectivas del padrón.
- Deberá contemplar todos procedimientos de intercambio con el I.N.E. ya bien sean de entrada o salida.
- Notificación automática de la baja al Ayuntamiento de procedencia con las tablas de direcciones de todos los Ayuntamientos.
- Permitirá, a partir de su información, extraer información para el censo electoral del municipio.
- Permitirá la confección automática de certificados (convivencia, residencia, viaje, empadronamiento, texto libre, etc.) en el momento de ser solicitados.
- A partir de la información contenida en el Padrón podrán realizarse estadísticas generales, para aportar información a otras administraciones (INE, ISTAC, EUSTAC...). Como mínimo :
 - Pirámides de población, simulaciones de evolución de población
 - Por año de llegada
 - Por lugar de procedencia, por lugar de nacimiento
 - Por nivel de instrucción, por estado civil, por miembros, por familia, etc.
 - Densidades de población según diferentes selecciones territoriales
- Listados:
 - Bajas, nacimientos, defunciones, matrimonios, etc.
 - Resumen numérico de población anual y mensual
 - Extranjeros comunitarios residentes en la ciudad
 - Extranjeros de otras procedencias y nacionalidades
- Resumen de actividades:
 - Certificados emitidos, por tipos, por días/meses, etc.

- Elecciones (censo electoral):
 - Generación y revisión
 - Carga del Censo Electoral que proporciona el INE para su consulta.
 - Gestión de cola de errores resultantes del intercambio de datos del censo electoral INE-AYUNTAMIENTO.
 - Mantenimiento del fichero de votantes, mesas electorales, composición de mesas, etc.
 - Posibilidad de efectuar por sorteo la composición de las mesas electorales
 - Se deberá adaptar el sorteo de los miembros de las mesas a las normas que para el mismo aprueba el Pleno de este Ayuntamiento, las designaciones de los miembros de las mesas no se realizan a través del proceso de insaculación, sino eligiéndose a través de un proceso informático.
 - Gestión de las renunciaciones de cargos a las mesas
 - Gestión de los componentes designados por mesas
 - Identificadores de cargos y comunicaciones con ellos y con la Oficina del censo Electoral
 - Obtención de los resultados electorales por mesas y total del municipio según la ley de Hondt.

11.6. El estado actual

GnuINE no está operativo para su uso en la actualidad ya que se encuentra en fase de desarrollo pero hay una versión beta funcionando para probar las bases de datos y la interfaz gráfica (GUI) desde hace tres años en un ayuntamiento.

El proyecto ha sido presentado a financiación del programa PROFIT 2002 del Ministerio de Ciencia y Tecnología MCYT. Y se prevé el lanzamiento de la primera versión estable a comienzos de la primavera de 2003.

11.7. Metodología de desarrollo y plan de trabajo

Se va a realizar el trabajo en un periodo de diez meses y se han considerado diez hitos importantes que coincidirán con el final de mes. Los diferentes módulos se irán validando en los ayuntamientos a medida que se vayan realizando.

F. Inicio	F. Fin	Descripción del modulo a desarrollar
1/3/2002	31/3/2002	Módulo de mantenimiento de tablas maestras.

1/4/2002	30/4/2002	Módulo de gestión de hojas padronales.
1/5/2002	31/5/2002	Módulo de gestión de habitantes.
1/6/2002	30/6/2002	Módulo de impresión de hojas padronales.
1/7/2002	31/7/2002	Módulo de impresión del callejero y tablas maestras.
1/8/2002	31/8/2002	Módulo de listados.
1/9/2002	30/9/2002	Módulo de exportación de datos, creación de disquetes y envío de información al INE.
1/10/2002	31/10/2002	Módulo de estadísticas e impresión de las mismas.
1/11/2002	30/11/2002	Módulo de certificados
1/12/2002	31/12/2002	Módulo de censo y gestión de mesas electorales.

11.8. gnuINE se libera bajo la GPL Licencia General Pública de la FSF

gnuINE es software libre los terminos de la licencia se pueden consultar en la Web de la FSF <http://www.fsf.org>

11.9. Tecnologías y recursos mas significativos incorporadas en el proyecto

Como se ha señalado con anterioridad sin la reutilización de código preexistente y la interacción con otras aplicaciones de SL no sería posible realizar este proyecto, ni en cuanto a costes de programación ni en cuanto a tiempos de desarrollo. Se puede hacer por que es Software libre y utiliza Software libre para su construcción.

- GNU/Linux: El autor primario del *kernel* Linux es Linus Torvalds y este kernel ha sido incorporado al sistema GNU para producir un sistema operativo completo. A partir de la versión original, ha sido mejorado por incontables personas de todo el mundo. Se trata de un clon del sistema operativo Unix, escrito desde cero. Ni USL, ni la Universidad de California, Berkeley, tuvieron participación en la escritura de GNU/Linux. Una de las cosas más interesantes de kernel Linux es que el desarrollo ocurre simultaneamente alrededor del mundo. Linux ha sido registrado bajo los términos de la Licencia Pública General GNU (GNU General Public License) o GPL. Esta licencia, escrita por la Free Software Foundation (FSF), está diseñada para evitar que alguna persona restrinja la distribución de software. También significa que debe estar disponible el código fuente.

- Python: Python es un lenguaje de programación fácil de aprender y potente. Tiene eficaces estructuras de datos de alto nivel y una solución de programación orientada a objetos simple pero eficaz. La elegante sintaxis de Python, su gestión de tipos dinámica y su naturaleza interpretada hacen de él el lenguaje ideal para guiones (scripts) y desarrollo rápido de aplicaciones, en muchas áreas y en la mayoría de las plataformas. El interprete de Python y la extensa biblioteca estándar están disponibles libremente, en forma de fuentes o ejecutables, para las plataformas más importantes, en la sede web de Python, <http://www.python.org>, y se pueden distribuir libremente.
- PostgreSQL: Postgres, desarrollada originalmente en el Departamento de Ciencias de la Computación de la Universidad de California en Berkeley, fue pionera en muchos de los conceptos de bases de datos relacionales orientadas a objetos que ahora empiezan a estar disponibles en algunas bases de datos comerciales. Ofrece soporte al lenguaje SQL92/SQL3, integridad de transacciones, y extensibilidad de tipos de datos. PostgreSQL es un descendiente de dominio público y código abierto del código original de Berkeley. La sede web de PostgreSQL es <http://www.postgresql.org>
- GTK: GTK (GIMP Toolkit) es una biblioteca para crear interfaces gráficas de usuario. Su licencia es la LGPL, así que mediante GTK podrá desarrollar programas con licencias abiertas, gratuitas, libres, y hasta licencias comerciales no libres, sin mayores problemas. Se llama el GIMP toolkit porque fue escrito para el desarrollo del General Image Manipulation Program (GIMP), pero ahora GTK se utiliza en un gran número de proyectos de programación, incluyendo el proyecto GNU Network Object Model Environment (GNOME). GTK está construido encima de GDK (GIMP Drawing Kit) que básicamente es un recubrimiento de las funciones de bajo nivel que deben haber para acceder al sistema de ventanas sobre el que se programe (Xlib en el caso de X windows).
- ReportLab: Es un generador de documentos en PDF, fundamentado sobre estándares abiertos manteniendo una fácil interoperabilidad con otros paquetes de software y sistemas. Los documentos generados con ReportLab pueden automáticamente incorporar datos de muchas fuentes, gráficos, imágenes o diseños de cualquier programa de diseño digital profesional. ReportLab usa el formato multiplataforma y abierto PDF, y puede incluir elementos de virtualmente cualquier aplicación que soporte impresión. Esto contrasta con formatos como el de Microsoft Word: cerrado, propietario y que solo interopera con una muy limitada lista de aplicaciones.