ANEXO II:

TECNOLOGÍAS Y SEGURIDAD DEL SISTEMA

Introducción

El Sistema de Notificaciones Electrónicas es un software desarrollado con el fin de brindar una solución tecnológica al envío de cédulas.

Funcionamiento

Funcionalidades del sistema:

Creación de cédulas

El sistema permite la creación de cédulas para su posterior envío.

Para ello implementa distintas tecnologías de navegador comentadas más adelante en este anexo.

Firma digital de cédulas

El sistema permite firmar digitalmente una cédula anteriormente creada.

Para lograr esto el sistema se copla a la infraestructura de clave pública.

"En criptografía, una infraestructura de clave pública (o, en inglés, PKI, Public Key Infrastructure) es una combinación de hardware y software, políticas y procedimientos de seguridad que permiten la ejecución con garantías de operaciones criptográficas como el cifrado, la firma digital o el no repudio de transacciones electrónicas."

El sistema permite a un suscriptor firmar digitalmente cédulas con los certificados otorgados por la ONTI que es la AUTORIDAD CERTIFICANTE DE LA ADMINISTRACIÓN PÚBLICA.

Envío de la cédula (notificación)

El sistema permite a los funcionarios judiciales poseedores de un certificado, notificar a letrados de manera electrónica.

Para ello el sistema envía la notificación utilizando el protocolo SMTP, hacia la casilla otorgada al letrado correspondiente.

Visualización de la cédula

El sistema permite a los letrados consultar sus cédulas desde internet con el uso de cualquier navegador.

El sistema de Notificaciones Electrónicas fue desarrollado a través de la tecnología **Symfony 2**, utilizando como gestor de base de datos **MySQL** y como programa servidor **Apache2**. También utiliza tecnologías como: Doctrine 2, Jquery, JavaScript.

El conjunto de tecnologías que utiliza el sistema de Notificaciones Electrónicas se ejecuta en un servidor cuyo sistema operativo es Centos 6.

Además se utiliza un servidor de correo electrónico montado en un servidor ZIMBRA 8.x. Alojado en un servidor con sistema operativo Centos 7.

Las tecnologías utilizadas fueron escogidas pensando en la usabilidad y, además, teniendo en cuenta la importancia de los aspectos de seguridad. Cabe mencionar también que todo el software con el que se desarrolló el sistema es software libre, por lo tanto no requiere de pago de licencias, entre otras ventajas.

Se contempló como parte de la política de seguridad la posibilidad de realizar el respaldo de la base de datos de forma automática de manera diaria.

A continuación se detallan las ventajas de usar las mencionadas tecnologías.

Symfony 2

Un framework simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, un framework proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Por último, un framework facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

Symfony 2 es un completo framework MVC (Modelo Vista Controlador) diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web.

Dicho framework está desarrollado completamente con PHP 5 y es compatible con la mayoría de los gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se puede ejecutar tanto en plataformas *nix (Unix, Linux, etc.) como en plataformas Windows.

A continuación se muestran algunas de sus características:

- **Alto rendimiento**: Symfony2 ha sido desarrollado teniendo en cuenta el rendimiento como mayor prioridad, por lo que es uno de los frameworks más rápidos. Hasta 3 veces más rápido que Zend Framework 1.10 y consume la mitad de la memoria.

- **Extensible**:Symfony2 se construye a base de bundles. Un bundle es un paquete en el que se incluye todos los componentes de una aplicación o módulo: código fuente, vista, controlador, modelo, etc. Si bien los bundlesposeen una estructura de archivos predefinido, da la flexibilidad al desarrollador a organizar sus archivos a su conveniencia.

- **Flexible**: Gracias a que Symfony2 cuenta con un micro-kernel basado en un contenedor de inyección de dependencia y un manejador de eventos muy fácil de configurar.

- **Construido para desarrolladores**: Symfony2 proporciona las herramientas que en gran medida mejoran la productividad de los desarrolladores, como la famosa barra de depuración web, soporte nativo de entornos, páginas detalladas de errores y mucho más.

- **Construido en base a otros grandes frameworks**: Symfony2 tomó lo mejor de los conceptos de otros frameworks de desarrollo como Django, Spring y Ruby onRails. También aprovecha componentes de Zend Framework y de Doctrine.

- **Listo para usar**: Symfony2 cuenta con todas las características que el desarrollador de aplicaciones web necesita. También proporciona seguridad integrada y promueve el desarrollo web utilizando buenas prácticas.

- **Open-Source**: Puede ser modificado y utilizado de forma gratuita y libre.

Además, gracias a la flexibilidad de organizar los archivos en bundles, Synfony 2 da la opción de incorporar a nuestros proyectos, bundles desarrollados por terceros. Esto le permite al desarrollador utilizar, en forma combinada con Symfony 2, otras tecnologías como por ejemplo:

- ORM (Object Relational-Mapping, Mapeo Objeto-Relacional): Doctrine, Propel, etc.

- Motores de plantillas: Twig, PHP, XML, etc.

- Sistema de control de versiones: Git, Subversión, etc.

Doctrine 2

El ORM (Object Relational Mapping, Mapeo Objeto-Relacional) es una técnica de programación que permite, a través de un motor de mapeo, manipular una base de datos relacional como si fuera una base de datos orientada a objetos. Dicha conversión, posibilita la utilización de técnicas de la orientación a objetos para manipular la base de datos, principalmente el polimorfismo y la herencia.

Doctrine es un ORM para PHP que provee una persistencia transparente para objetos PHP. Su tarea principal es la de convertir las filas de una base de datos relacional en un objeto PHP y viceversa.

Una de sus características más importante es la de brindar un lenguaje propio de consultas denominado DQL (Doctrine QueryLenguage, Lenguaje de Consultas Doctrine) el cual aumenta el alcance y flexibilidad de las consultas SQL.

Algunas de las características más importantes de Doctrine son:

- **Generación automática del modelo**: Cuando se trabaja con ORM, necesitas crear el conjunto de clases que representa el modelo de la aplicación, luego estas clases serán vinculadas al esquema de la base de datos de forma automática con un motor ORM. Aunque son cosas diferentes, cuando diseñas un modelo relacional y un modelo de clases, suelen ser muy parecidos. Doctrine se aprovecha de esta similitud y nos permite generar de forma automática el modelo de clases basándose en el modelo relacional de tablas. Es decir, si tenemos una tabla llamada usuarios, se autogenerará una clase llamada Usuarios cuyas propiedades son las columnas de dicha tabla.

- **Generación manual del modelo de diferentes formas**: Doctrine puede generar de forma automática el modelo, pero también deja la posibilidad (como es lógico) que puedas definir tu mismo el mapeo de tablas y sus relaciones. Esto se puede hacer con código PHP, con YAML, que es un formato de serialización de datos legible por humanos muy usado para este fin, o con anotaciones.

- **Doctrine_Record y Doctrine_Table**: Prácticamente todo nuestro modelo heredará de estas dos clases. Doctrine_Record representa una entidad con sus propiedades (columnas) y nos facilita métodos para insertar, actualizar o eliminar registros entre otros.

- **Buscadores mágicos (Magic finders)**: En Doctrine, puedes buscar registros basándote en cualquier campo de una tabla. Si existen los campos llamados name y email, podemos hacer findByName () y findByEmail (). También es importante decir que existe el método findAll (), que obtiene todos los registros de la tabla.

- **Relaciones entre entidades:** En Doctrine, una vez que se ha definido el modelo (o se ha creado de forma automática) con las tablas y sus relaciones, resulta fácil acceder y moverse por entidades relacionadas.

- **Lenguaje DQL:** DQL es un lenguaje creado para ayudar al programador a extraer objetos de la base de datos. Entre las ventajas de usar este lenguaje se encuentran:

- Está diseñado para extraer objetos, no filas, que es lo que interesa.

- Entiende las relaciones, por lo que no es necesario escribir los joins a mano.

- Portable con diferentes bases de datos.

Twig

Una plantilla es un archivo de texto. Esta puede generar cualquier formato basado en texto (HTML, XML, CSV, LaTeX, etc.). Una plantilla contiene variables o expresiones, las cuales se reemplazan por valores cuando se evalúa la plantilla, y etiquetas que controlan la lógica de la plantilla.

Twig, es un potente motor de plantillas desarrollado para PHP que permite realizar un diseño web más estructurado y fácil de mantener. Twigesta basado en otros lenguajes de plantillas de texto como Django, Smarty y Jinja.

Algunas de sus características más importantes son:

- **Rápido**: Twig compila las plantillas hasta código PHP regular optimizado. El costo general en comparación con código PHP regular se ha reducido al mínimo.

- **Seguro:** Twig tiene un modo de recinto de seguridad para evaluar el código de plantilla que no es confiable. Esto te permite utilizar Twig como un lenguaje de plantillas para aplicaciones donde los usuarios pueden modificar el diseño de la plantilla.

- **Flexible**: Twig es alimentado por flexibles analizadores léxico y sintáctico. Esto permite al desarrollador definir sus propias etiquetas y filtros personalizados, y crear su propio DSL.

- **Herencia:**La herencia de plantillas te permite crear un "esqueleto" de plantilla base que contenga todos los elementos comunes de tu sitio y definir los bloques que las plantillas descendientes pueden sustituir.

Seguridad lógica

- Cuenta con un amplio soporte para la seguridad lógica del sitio, que permite protegernos de los ataques más comunes hoy en día existentes, como ser XSS (Cross-site scripting), CSRF (Cross Site Request Forgery), SQL Injection, etc.

Apache

Apache, es un programa servidor web, creado en el año 1995 por un grupo de desarrolladores sin fines de lucro, denominados Apaches.

Hasta Diciembre del 2011, más del 65% de los servidores web del mundo utilizan Apache, según una encuesta realizada por la compañía Netcraft.

Algunas de las características que llevan a elegir a Apache como servidor web son:

- **Es un sistema multiplataformas**: el servidor Apache puede instalarse en la mayoría de los sistemas operativos más populares, por ejemplo: Windows, Linux, Unix y Macintosh.

- **Es un sistema modular**: cualquiera que posea una modesta experiencia en los lenguajes de programación C o Perl, puede desarrollar un módulo que permita ampliar las funcionalidades de Apache a su conveniencia. Además, en la actualidad existe una variedad de módulos que, luego de una previa configuración, pueden adaptarse a la versión estándar de Apache.

- **Es una tecnología gratuita de código abierto**: esto significa que no hay que pagar ninguna licencia para su utilización. Además es posible modificar el programa a las necesidades de cada uno e incluso realizar aportes a la comunidad Apache mediante el desarrollo de alguna funcionalidad. Cabe

aclarar, que para que dicho aporte se incluya en la versión estándar de Apache, este debe ser aprobado por la comunidad Apache.

- **Documentación:** existe una gran cantidad de libros y ejemplos de administración del servidor Apache, lo cual es de gran importancia para aquellas personas que se inician con esta tecnología.

- **Soporte del protocolo HTTP**(Hipertext Transfer Protocol, Protocolo de Transferencia de Hipertexto): es el protocolo usado en cada transacción de la World Wide Web.

MySql

MySQL es un sistema gestor de base de datos SQL relacionales. Actualmente, se ha convertido en el SGBD de código abierto más popular debido a su alto rendimiento, alta fiabilidad y facilidad de uso.

Algunas de sus características más importantes son:

Interioridades y portabilidad:

- MySQL se ejecuta en más de 20 plataformas, incluyendo Linux, Windows, Mac OS, Solaris, AIX de IBM, etc.

- APIs disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, Tcl , etc.

- Proporciona sistemas de almacenamientos transaccionales y no transaccionales.

- Joins muy rápidos usando un multi-join de un paso optimizado.

- Tablas hash en memoria, que son usadas como tablas temporales.

- Las funciones SQL están implementadas usando una librería altamente optimizada y deben ser tan rápidas como sea posible.

- El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.

Seguridad:

- Cuenta con un sistema de privilegios y contraseñas muy flexibles y seguro, que permite verificación basada en el host. Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptado cuando se conecta con un servidor.

Escalabilidad y límites:

MySQL posee la capacidad de almacenar más de 50 millones de registros y más de 60 mil tablas.

- Se permiten hasta 64 índices por tabla. Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2). Un índice puede usar prefijos de una columna para los tipos de columna CHAR, VARCHAR, BLOB, o TEXT.

Conectividad:

- Los clientes pueden conectar con el servidor MySQL usando sockets TCP/IP en cualquier plataforma.

Sistema Operativo

CentOS (Community ENTerprise Operating System) es una bifurcación a nivel binario de la distribución Linux Red Hat Enterprise Linux RHEL, compilado por voluntarios a partir del código fuente publicado por Red Hat, siendo la principal diferencia con este la remoción de todas las referencias a las marcas y logos propiedad de Red Hat.

Es un sistema operativo de código abierto, basado en la distribución Red Hat Enterprise Linux, operándose de manera similar, y cuyo objetivo es ofrecer al usuario un software de "clase empresarial" gratuito. Se define como robusto, estable y fácil de instalar y utilizar. Desde la versión 5, cada lanzamiento recibe soporte durante diez años, por lo que la actual versión 7 recibirá actualizaciones de seguridad hasta el 30 de junio de 2024.

Soporte y Respaldo de la información

A nivel de sistema operativo se utiliza **Crontab**. En el sistema operativo Unix, crontab es un administrador regular de procesos en segundo plano (demonio) que ejecuta procesos o guiones a intervalos regulares. A través del mismo se configuró el servidor para realizar el respaldo de la base de datos todos los días a las 14:00 hs. permitiendo en caso de fallo la recuperación de los mismos.

Seguridad de la red

El sistema se aseguró en la red por medio de IPTABLES. Iptables, una herramienta de cortafuegos que permite no solamente filtrar paquetes, sino también realizar traducción de direcciones de red (NAT) para IPv4 ó mantener registros de log.
Se utilizó como política de seguridad el cerrado de todos los puertos excepto los que utiliza el sistema para su funcionamiento.

Certificados SSL

Los certificados SSL (capa de sockets seguros) son una pieza esencial de la seguridad de los sitios web. Al visitar un sitio web con SSL, el certificado SSL del sitio web permite cifrar los datos que se envían, como la información sobre tarjetas de créditos, nombres y direcciones de modo que ningún hacker pueda acceder a ellos. Para comprobar si un sitio web usa SSL correctamente, escriba la dirección del sitio web en nuestro Comprobador de instalación SSL.

El protocolo TLS (seguridad de la capa de transporte) es solo una versión actualizada y más segura de SSL. Si bien aún se denomina a los certificados de seguridad SSL por su uso más común.-