



Dirección de Admisión y Control de Estudios

Documento de la Arquitectura del Software
Proyecto: Sistema de seguimiento y control de
Solicitudes en DACE

Versión 1.0

Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

Historia de Revisiones

Fecha	Versión	Descripción	Autor
05/12/2012	1.0	Documento de arquitectura de Software	Arturo Murillo

Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

Tabla de Contenido

1. Introducción	4
1.1 Propósito	4
1.2 Alcance	4
1.3 Definiciones, Acrónimos y Abreviaturas	4
1.1 Vista Global	5
2. Representación Arquitectónica	5
3. Metas y Restricciones Arquitectónicas	5
4. Vista de Casos de Uso	6
Diagrama de Caso de uso	8
5. Vista Logica	9
6. Representación Arquitectónica	9
7. Vista de Implementación	9
8. Vista de Datos	10
9. Representación Arquitectónica	11
10. Representación Arquitectónica	11

Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

Documento de la Arquitectura del Software

1. Introducción

El documento de la Arquitectura del Software provee al usuario especializado una descripción de la arquitectura del Sistema de seguimiento y control de Solicitudes en DACE. La elaboración del mismo está basada en las especificaciones de RUP (Rational Unified Process) para el documento de arquitectura del software, y principalmente en las vistas arquitectónicas del modelo “4+1” de Kruchten (Vista Lógica, Vista de Procesos, Vista de Desarrollo, Vista Física y Escenarios). Además, a través de esta herramienta se pueden decisiones que tienen alto impacto en la fase de implementación y también es una forma de describir de forma concisa los elementos que componen el sistema.

1.1 Propósito

El propósito general del DAS, es dar una visión clara de los aspectos más relevantes del sistema. Esto se logra haciendo un análisis minucioso de las vistas que se cubren en éste documento. Este documento tiene además, el propósito de brindar una descripción de la arquitectura del sistema en vías de desarrollo. Para lograr dicho objetivo se utilizarán distintos diagramas UML, los cuales ilustran las funcionalidades que brindará el sistema; otro aspecto que contempla, es la realización de un análisis de las vistas señaladas del modelo “4+1”. De esta manera se brinda una visión clara de los aspectos más importantes del sistema. A nivel de usuario, programador o arquitecto de software este es el documento que genera la base técnica que permite el conocimiento posterior del Sistema.

1.2 Alcance

El Documento de Arquitectura del Software describe los aspectos más relevantes del sistema Sistema de Solicitudes e influye en la elaboración del mismo, centrándose en los casos de uso (funcionalidades), en el diseño y en la toma de decisiones. El DAS del sistema describe y documenta en cuanto a detalles de implementación, el 100% de la funcionalidad del sistema.

1.3 Definiciones, Acrónimos y Abreviaturas

- *DACE: Dirección de Administración y Control de Estudios*
- *Caso de Uso: acción realizada por el sistema, la cual devuelve un resultado observable.*
- *USB: Acrónimo de Universidad Simón Bolívar.*
- *CAS: Central Authentication Service*
- *RUP: Rational Unified Process,*
- *UML: Unified Modeling View.*
- *MVC: Model View Controller.*

Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

1.4 Vista Global

En este documento se presentan las diferentes vistas que conforman la arquitectura del software del sistema a desarrollar, teniendo como punto de partida el Modelo de Casos de Uso del mismo. Estas vistas sirven de guía para los desarrolladores, en la implementación y el análisis del sistema. Desarrollar detalladamente la representación arquitectónica, permite tener una idea de que se busca funcional y estructuralmente del sistema. El documento se encuentra organizado en diversas visiones que ilustran la composición interna del sistema como las relaciones y posibles interacciones entre sus partes.

2. Representación Arquitectónica

El Sistema, esta desarrollado bajo la arquitectura de cliente-servidor. Esta aplicación tendrá un acceso privado a través del USBID, el cual tendrá un perfil restringido, y un acceso privado, en el que sólo los usuarios con un USBID podrán tener acceso, a excepción de los estudiantes que pueden acceder a través de cuentas de organizaciones estudiantiles. La arquitectura será presentada como un conjunto de vistas usando el modelo "4+1" de Kruchten, el cual contiene las vistas: Lógica, Procesos, Implantación e Implementación.

- - **Vista de Procesos:** Toma en cuenta los requisitos no funcionales del aspecto dinámico del sistema: desempeño, concurrencia, integradores y escalabilidad.

- - **Vista Física:** Todos los requerimientos no funcionales del sistema son presentados y descritos en esta vista. Esta vista también se llama de implantación, y se trata acerca del entorno de hardware y software donde se espera instalar el sistema. En este documento se especifican los requisitos de desempeño, mantenibilidad, usabilidad, disponibilidad, portabilidad, y herramientas de desarrollo.

- - **Vista de desarrollo:** Tiene en cuenta los requisitos internos relativos a la facilidad de desarrollo, administración del software, reutilización y elementos comunes y restricciones impuestas por las herramientas o el lenguaje de programación que se use. Esta vista será elaborada siguiendo el patrón MVC (Modelo Vista Controlador), el cual separa los elementos de la aplicación en tres capas. Se centra en la organización real de los módulos del software, sirviendo así como un mapa del software dentro del hardware.

- - **Vista Lógica:** Su principal objetivo es modelar los requisitos funcionales que deberá brindar el sistema a sus usuarios. , además, trata acerca de las clases que componen el sistema y las relaciones entre ellas. Se utiliza el diseño orientado a objetos y todos los conceptos relacionados a éste. Incluye los diagramas: Modelo de Casos de Uso, Diagrama Conceptual, Diagrama de Clases y Diagrama de Actividades.

- - **Vista Escenario** (casos de uso): Esta vista consiste en la integración de las otras cuatro.

3. Metas y Restricciones Arquitectónicas

En cuanto al desempeño del sistema, el mismo deberá estar preparado para responder y recuperarse de las fallas técnicas que puedan presentarse, y en condiciones normales su tiempo de respuesta deberá ser mínimo. El sistema deberá ser de fácil mantenimiento, esto se refiere a que actualizar, encontrar fallas y corregir los errores de la aplicación, no deben requerir de mucho esfuerzo y tiempo. Esto es facilitado por la cantidad de documentación y material informativo que se está elaborando, y por el uso de herramientas de Software Libre, las cuales no generan costo alguno. En

Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

cuanto a la usabilidad del sistema, se espera que el entrenamiento inicial para el uso del mismo, no sea exhaustivos. Además, el desarrollo de una interfaz sencilla y amigable permitirá que cualquier usuario pueda interactuar con el sistema, sin necesidad de que posea algún conocimiento previo de computación. Tomando en cuenta la disponibilidad del sistema, el mismo deberá estar disponible, en condiciones normales, las 24 horas del día, los 7 días de la semana, el promedio de atención de fallas no deberá exceder de un par de días. Por último, tomando en consideración la portabilidad del sistema, éste deberá ser soportado por cualquier navegador web, lo cual facilitará el acceso al mismo por parte de los usuarios, desde cualquier lugar. Además, la aplicación podrá ser instalada en cualquier computadora que soporte el Software utilizado, lo cual no tiene mayor complicación debido a que es libre.

Durante la planificación de este proceso se han presentado las siguientes restricciones arquitectónicas:

- Restricciones de uso de herramientas de desarrollo. La aplicación será desarrollada bajo software libre.
- Restricciones de seguridad y privacidad: es necesario brindarle a los usuarios un sistema que garantice la protección de su información y datos, por lo que se deberán elaborar mecanismos de seguridad como la encriptación de datos.
- Restricciones de tiempo: se ha fijado una fecha de entrega, en la cual debe estar finalizado e instalado el sistema.

4. Vista de Casos de Uso

Actores

Actores	Descripción
Usuario	Miembros de la comunidad universitaria, profesores, unidad administrativa, empleados y DACE.
<i>Solicitante</i>	<i>Usuario del sistema que es capaz de enviar solicitudes a DACE.</i>
<i>Profesor</i>	<i>Profesores miembro de la USB.</i>
<i>Unidad Administrativa</i>	<i>Departamento o Coordinación de la USB.</i>
<i>Empleado</i>	<i>Empleado miembro de la USB.</i>
<i>DACE</i>	<i>Dirección de Admisión y Control de Estudio, Administrador del Sistema.</i>

Lista de Casos de Uso por Actor

Usuario

Iniciar USBID
Consultar Archivo
Editar Permisos
Eliminar Archivo

Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

[Descargar Archivo](#)
[Subir Archivo](#)
[Salir](#)

Solicitante

[Especificar Dependencia](#)
[Agregar Solicitud](#)
[Enviar Email](#)
[Consultar Solicitud](#)
[Editar Solicitud](#)

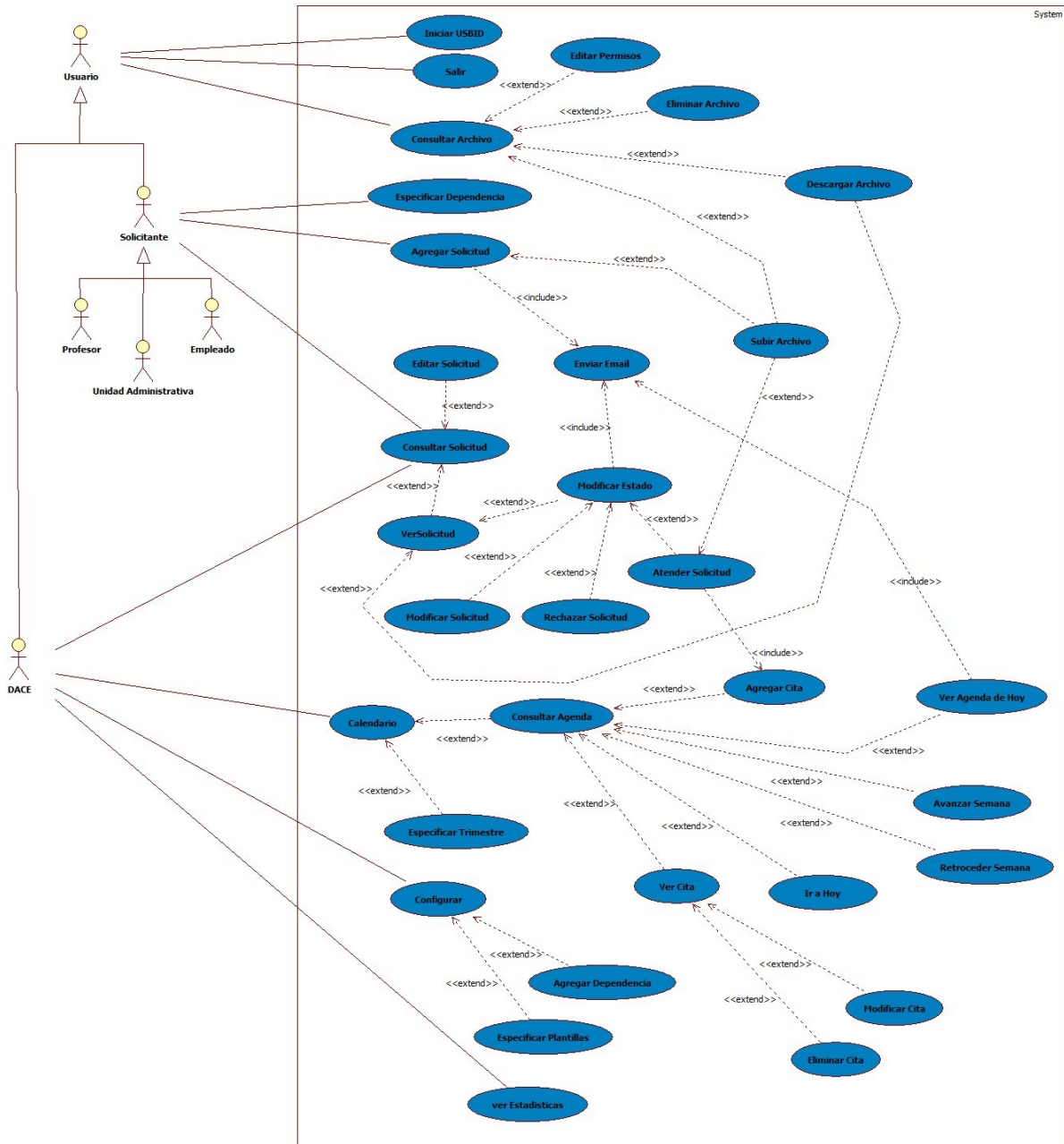
DACE

[Consultar Solicitud](#)
[Ver Solicitud](#)
[Modificar Estado](#)
[Modificar Solicitud](#)
[Rechazar Solicitud](#)
[Atender Solicitud](#)
[Calendario](#)
[Enviar Email](#)
[Consultar Agenda](#)
[Agregar Cita](#)
[Ver Agenda Hoy](#)
[Avanzar Semana](#)
[Retroceder Semana](#)
[Ir a Hoy](#)
[Ver Cita](#)
[Modificar Cita](#)
[Eliminar Cita](#)
[Especificar Trimestre](#)
[Configurar](#)
[Agregar Dependencia](#)
[Especificar Plantillas](#)
[Ver Estadísticas](#)

Total Casos de Uso: 35

Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

Diagrama de Caso de uso



Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

5. Vista Lógica

La Vista Lógica del Sistema está formada por tres componentes principales.

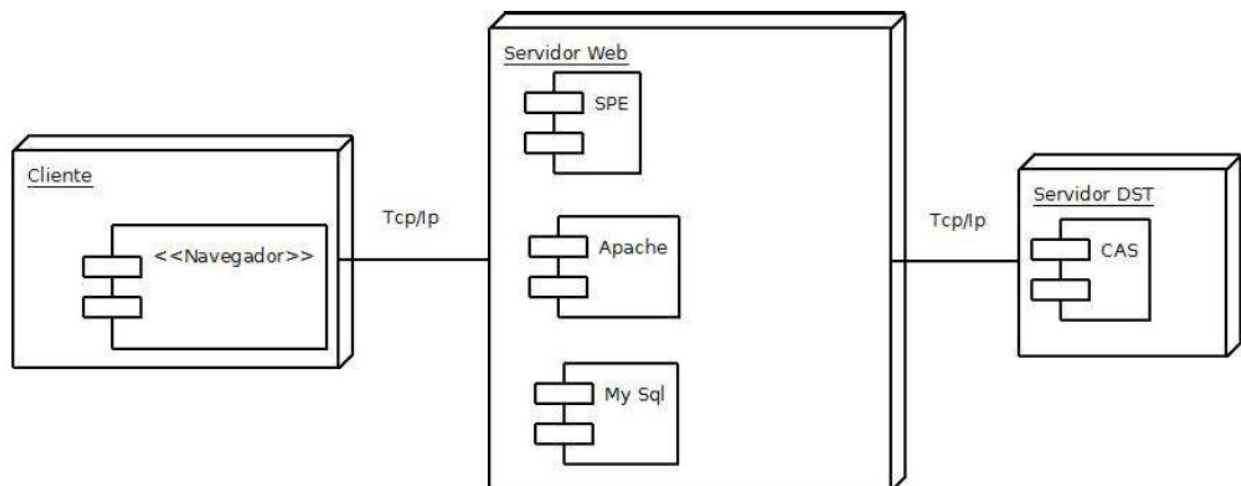
Las Vistas: Todos los Archivos que corresponden a las vistas con la interfaz del Sistema, se nombran por convención de la forma vNombreArchivo.php. Sólo contienen interfaz, no se realizan consultas en ellas.

Los Controladores: Todos los Archivos que corresponden a los controladores del Sistema, se nombran por convención de la forma cNombreArchivo.php, donde NombreArchivo se corresponde con su respectiva vista.

Las Verificaciones: Todas las Verificaciones de los formularios se encuentran en los archivos que siguen la notación cVerifNombreArchivo.js

6. Vista de Implantación

diagrama principal de diseño.



7. Vista de Implementación

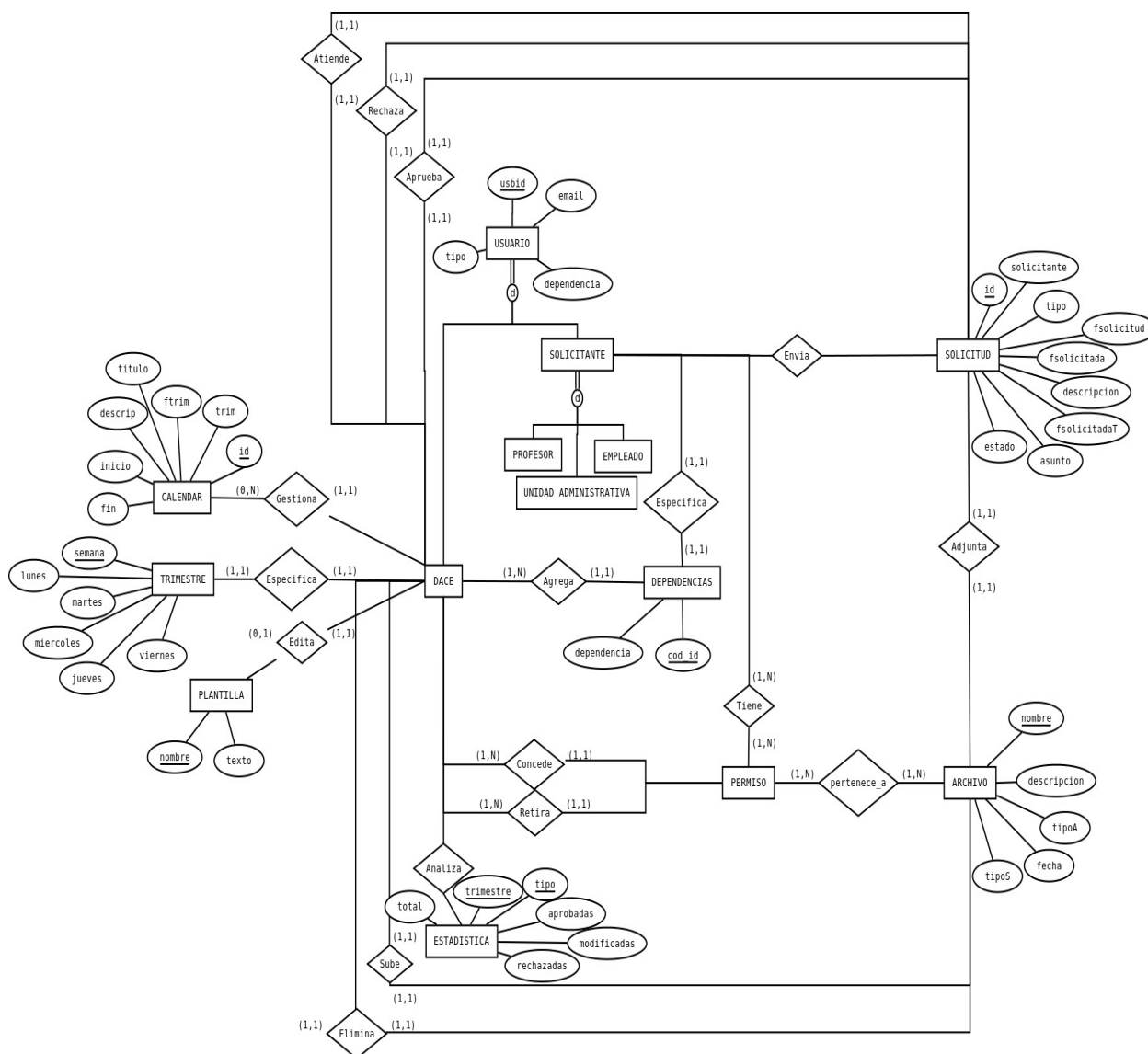
Cuando se desarrolla un sistema, una de las prácticas más necesarias y útiles para el desarrollador es que el código utilizado sea reutilizable. En numerosas ocasiones, es posible que un mismo problema se nos presente en situaciones y formas distintas, por lo tanto es posible que ya hayamos resuelto dicho problema anteriormente y podríamos ser capaces de reutilizar el código de la solución. Esto se logra mucho más fácilmente si el código implementado se encuentra separado en segmentos o módulos (no modificados), que podrían ser de beneficio a futuro. Por esto es necesario el uso de patrones de diseño dentro de las arquitecturas de software.

Para mantener la modularidad y flexibilidad al programar, se ha decidido emplear el patrón arquitectural MCV (Modelo-Vista-Controlador). El MVC es un patrón de desarrollo que separa la parte lógica y la interfaz de una aplicación. Dicho patrón permite organizar el código de una aplicación separando los

Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

datos, la interfaz de usuario, y la lógica de control en tres componentes distintos. La principal ventaja de esta separación reside en la facilidad para realizar cambios en la aplicación puesto que: Cuando realizamos un cambio de bases de datos, programación o interfaz de usuario solo tocaremos uno de los componentes. Podemos modificar uno de los componentes sin conocer cómo funcionan los otros.

8. Vista de Datos



Sistema de seguimiento y control de Solicitudes en DACE	Versión: 1.0
Documento de la Arquitectura del Software	Fecha: 05/12/2012
0002	

10. Tamaño y Desempeño

El Sistema seguimiento y control de Solicitudes en DACE, será un sistema de mediana envergadura, ya que, el número de usuarios que usaran el sistema no es un número considerablemente grande, sin embargo, el propósito general del sistema es automatizar y facilitar el envío y atención de solicitudes en DACE, es decir, el procesamiento de esta información si representa un trabajo grande para el sistema y se puede considerar como un punto crítico del mismo. Es por esta razón, que durante la implementación y desarrollo del sistema se tomo especial cuidado respecto a este punto y se realizaron las pruebas necesarias para que de ese modo se logre el mayor desempeño posible.

Es importante mencionar que para asegurar un buen desempeño hay varios aspectos que se tomaran en cuenta. El sistema debe poseer una codificación robusta para poder soportar las constantes modificaciones que serán realizadas sobre su base de datos interna. Estas modificaciones deben ser llevadas a cabo de manera eficiente y segura, de tal manera que se mantenga la consistencia y permanencia absoluta de la información. Con el tiempo el sistema debe generar los menores errores posibles, siendo estos además de baja escala, para así disminuir los costos y trabajos de mantenimiento. Además, cada cierto tiempo deben realizarse copias de seguridad que aseguren la información y la envíen a un sitio seguro.

Los tiempos de respuesta, también se toman en consideración, tanto de consultas como de acceso, modificación, creación, validación y eliminación de datos deben ser mínimos, para que de ese modo se logre cumplir el objetivo del sistema de una manera eficiente y robusta. Por otro lado, el sistema debe responder de manera rápida y efectiva incluso si hay una notable cantidad de usuarios autenticados y realizando acciones sobre el sistema. Otro aspecto donde es importante minimizar el tiempo de respuesta es durante los accesos a la Base de Datos, en este sentido, de estos factores dependerá el tiempo de respuesta al acezar a la página Web del sistema desde la Red. En una página de servicios en línea no se puede permitir que el acceso a la misma, tarde muchos segundos en mostrarse al Usuario, por tanto, el diseño de la interfaz gráfica debe ser controlado (no recargado) y reducido de tal manera que no obstaculice la fluidez en la navegación por el sistema.

11. Calidad

El Sistema seguimiento y control de Solicitudes en DACE, contribuirá en DACE la automatización el control de solicitudes en DACE, logrando mejorar el proceso de gestión que realiza dicha unidad. El proceso de atención y envío de información que se realiza actualmente de manera manual ocasiona posible perdida de información y en casos olvido, además causando repetición de solicitudes que son periódicas. El principal objetivo de este sistema es facilitar y agilizar dichas tareas de la dirección, es por esto que la calidad del sistema es prioridad en el desarrollo de este sistema. En este sentido, la elaboración de una buena arquitectura del software es fundamental para lograr dicho objetivo y para la realización de esta buena arquitectura es necesario tomar en cuenta tanto los requerimientos funcionales como los no funcionales. Los primeros satisfacen las necesidades prácticas del cliente, ya que serán los servicios que el sistema deberá brindar; por otro lado, los segundos son aquellos aspectos que mejoran, completan y fortalecen al sistema. Entre los requerimientos no funcionales se tiene: la confiabilidad, la portabilidad, el mantenimiento, la usabilidad, entre otros.

Cuando se trata el tema de calidad, la realización de pruebas al sistema es primordial para la determinación de dicha calidad. En este sentido, el protocolo de envío de solicitud y el protocolo de atención de las mismas ha sido probados en diversos casos críticos y han superado cada uno de ellos con un óptimo desempeño, logrando así un alto grado de confiabilidad y veracidad para estos protocolos desarrollados.