**<Plataforma de encuestas universidad del Quindío>**

Software Architecture Document

Version <1.1>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <13/09/15> | <0.7> | <Versión preliminar de Arquitectura de Software > | <Miller Stiven Aguirre Herrera>  <Joan Alejandro Martinez Murcia> |
| <18/09/15> | <0.8> | <Correcciones sugeridas para la Arquitectura de Software > | <Miller Stiven Aguirre Herrera>  <Joan Alejandro Martinez Murcia> |
| <26/09/15> | <0.9> | <Cambio de imágenes por modelos de UML, e inclusión de vista de datos > | <Miller Stiven Aguirre Herrera> |
| <01/10/15> | <.1.0> | <Corrección de los casos de uso en los diagramas UML, imagen 6 y 7 > | <Joan Alejandro Martinez Murcia> |
| <03/10/15> | <.1.1> | <Mayor aclaración en las capas de implementación describiendo que componentes ayudan el acoplamiento con cada una de las capas y cambios de la Size and Performance> | <Miller Stiven Aguirre Herrera> |

Table of Contents

1. Introduction

1.1 Purpose

1.2 Scope

1.3 Definitions, Acronyms, and Abbreviations

1.4 References

1.5 Overview

2. Architectural Representation

3. Architectural Goals and Constraints

4. Use-Case View

5. Logical View

5.1 Overview

5.2 Architecturally Significant Design Packages

5.3 Use-Case Realizations

6. Process View

7. Deployment View

8. Implementation View

8.1 Overview

8.2 Layers

9. Data View (optional)

10. Size and Performance

11. Quality

Software Architecture Document

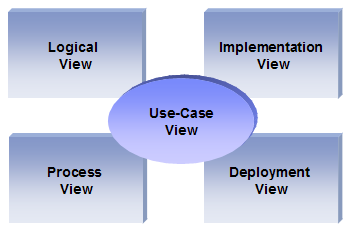
# Introduction

Este artefacto de Arquitectura de software es una versión preliminar preparada para ser incluida en la propuesta elaborada como respuesta al proyecto del espacio académico de Ingeniería de Software III de la Universidad del Quindío.

El proyecto está basado en una metodología ágil, la cual se está definiendo actualmente para verificar en cuál se adapta mejor el proyecto, en este documento solo se evidencian los espacios estrictamente solicitados por el artefacto de arquitectura.

## Purpose

Este documento de arquitectura de software tiene como propósito brindar una visión comprensible de la arquitectura general del software planificación y control de la plataforma de encuestas utilizando diferentes vistas de la arquitectura para ilustrar diferentes aspectos del mismo



## Scope

La plataforma de encuestas de la universidad del Quindío es una aplicación que sirve para la ejecución de encuestas en línea para los estudiantes y docentes de la universidad del Quindío, la cual tiene como propósito analizar, segmentar, e identificar falencias en los procesos

## Definitions, Acronyms, and Abbreviations

## 

**Definiciones:**

**Actualización**.- Insertar, eliminar, modificar los registros de los usuarios.

**Almacenamiento**.- En relación con ordenadores o computadoras, cualquier dispositivo capaz de almacenar información procedente de un sistema informático.

**Backup**.- Las copias de seguridad en un sistema informático tienen por objetivo el mantener cierta capacidad de recuperación de la información ante posibles pérdidas. Esta capacidad puede llegar a ser algo muy importante, incluso crítico, para las empresas. Se han dado casos de empresas que han llegado a desaparecer ante la imposibilidad de recuperar sus sistemas al estado anterior a que se produjese un incidente de seguridad grave.

**Base de Datos**.- Cualquier conjunto de datos organizados para su almacenamiento en

la memoria de un ordenador o computadora, diseñado para facilitar su mantenimiento y

acceso de una forma estándar. La información se organiza en campos y registros. Un campo se refiere a un tipo o atributo de información, y un registro, a toda la información sobre un individuo.

**Botón.**- Es un objeto tangible que realiza un evento tras su activación.

**Conexión.**- Comunicación entre dos entes que tienen características similares de comunicación.

**Interfaz.**- Medio que permite la comunicación entre el usuario y el sistema.

**Internet**.- interconexión de redes informáticas que permite a los ordenadores o computadoras conectadas comunicarse directamente, es decir, cada ordenador de la red puede conectarse a cualquier otro ordenador de la red. El término suele referirse a una interconexión en particular, de carácter planetario y abierto al público, que conecta

redes informáticas de organismos oficiales, educativos y empresariales.

**Login**.- Nombre o alias que se le da a una persona para permitirle el acceso al sistema

siempre y cuando estén registrados.

**PASSWORD**.- Contraseña o clave para autentificar el ingreso a un lugar o sitio.

**Protocolo**.- Señal mediante la cual se reconoce que puede tener lugar la comunicación o la transferencia de información.

**Servidor**.- Computadora conectada a una red que pone sus recursos a disposición del resto de los integrantes de la red. Suele utilizarse para mantener datos centralizados o para gestionar recursos compartidos.

**Servidor de archivos**.- Dispositivo de almacenamiento de archivos en una red de área local, o en Internet, al que los distintos usuarios de la red pueden acceder, en función de los privilegios que les hayan sido dados por parte del administrador.

**ACRÓNIMOS:**

**DBA**.- Es aquella persona que tiene el control central del sistema de base de datos.

**GUI** o acrónimo de Graphical User Interface.- En informática, tipo de entorno que permite al usuario elegir comandos, iniciar programas, ver listas de archivos y otras opciones utilizando las representaciones visuales (iconos) y las listas de elementos del menú. Las selecciones pueden activarse bien a través del teclado o con el ratón.

**TCP/IP**.- Acrónimo de Transmisión Control Protocol/Internet Protocol (protocolo de control de transmisiones/protocolo de Internet), protocolos usados para el control de la transmisión en Internet. Permite que diferentes tipos de ordenadores o computadoras se comuniquen a través de redes heterogéneas.

**URL.**- Un URL es una cadena de caracteres que identifica el tipo de documento, la computadora, el directorio y los subdirectorios en donde se encuentra el documento y su nombre.

**ABREVIATURAS**

HW: Hardware

SW: Software

Sr. Señor

Sra. Señora

Ing. Ingeniero(a)

## References

<http://www.slideshare.net/rickkazman>

L. Chung and J. do Prado Leite, “On Non-Functional Requirements in Software Engineering,” Conceptual Modeling:

Foundations and Applications, Springer- Verlag, 2009, pp. 363–379; <http://dx.doi.org/10.1007/978-3-642-02463-4_19>.

B. Nuseibeh, “Weaving Together Requirements and Architectures,” Computer, vol. 34, no. 3, 2001, pp.115–119.

P. Avgeriou et al., Relating Software Requirements and Architectures, Springer, 2011.

## Overview

En este documento se muestra las vistas del sistema en base a las metas y restricciones de la arquitectura. Además se muestran las siguientes vistas: vista lógica y vista de. Cada una de ellas incluye una breve descripción del contenido.

# Architectural Representation

En este documento se detalla la arquitectura utilizando las vistas definidas en el modelo de "4 + 1", pero utilizando la convención de nomenclatura RUP. Las vistas que se utilizan para documentar la plataforma encuestas uniquindio son:  
  
**Vista lógica**  
Audiencia: Diseñadores.  
Área: Requisitos funcionales: describe modelo de objetos de diseño. También se describen las más importantes realizaciones de casos de uso.  
Los artefactos relacionados: modelo de Diseño  
  
**Vista Proceso**  
Audiencia: Integradores.  
Área: requisitos no funcionales: describe concurrencia y sincronización aspectos del diseño.  
Los artefactos relacionados: (ningún artefacto específico).  
  
**Vista de la implementación**Audiencia: Los programadores.  
Área: componentes de software: describe las capas y subsistemas de la aplicación.  
Los artefactos relacionados: modelo de implementación, componentes  
  
**Vista de despliegue**  
Audiencia: Gestores de despliegue.  
Área: Topología: describe el mapeo del software en el hardware y muestra aspectos distribuidos del sistema.  
Los artefactos relacionados: modelo de despliegue.  
  
**vista de casos de uso**  
Público: todos los actores del sistema, incluidos los usuarios finales.  
Área: describe el conjunto de escenarios y / o casos de uso que representan algunas funciones importantes, el centro del sistema.  
Los artefactos relacionados: Modelo de casos de uso, los documentos de casos de uso  
  
**Vista de datos (opcional)**  
Público: especialistas de datos, administradores de bases de datos  
Área: Persistencia: describe los elementos persistentes arquitectónicamente significativos en el modelo de datos  
Los artefactos relacionados: modelo de datos.

# Architectural Goals and Constraints

3.1 La solicitud en de una nueva encuesta desplegará en un servidor en el cual estara contenida la aplicación y mostrará en pantalla la encuesta después del proceso de logueo.

3.2 El proceso de transacción es el envío de la encuesta es un proceso transaccional, el aprovechamiento de las capacidades de la plataforma técnica. Modelo de gestión de transacciones de la plataforma utilizada que actualmente se está definiendo y depurando.

3.3 La aplicación debe implementar conductas básicas de:

* **Autenticación**: inicia sesión usando al menos un nombre de usuario y una contraseña.
* **Autorización**: de acuerdo a su perfil permite la visualización de una encuesta u otra
* **Para el acceso a Internet, los siguientes requisitos son obligatorios Confidencialidad**: Los datos sensibles deben ser reservados para que las encuestas no se conviertan en persecución por parte de directivos o interesados en tergiversar las percepciones de los encuestados.
* **Integridad de datos:** Los datos enviados a través de la red no puede ser modificado por un nivel.

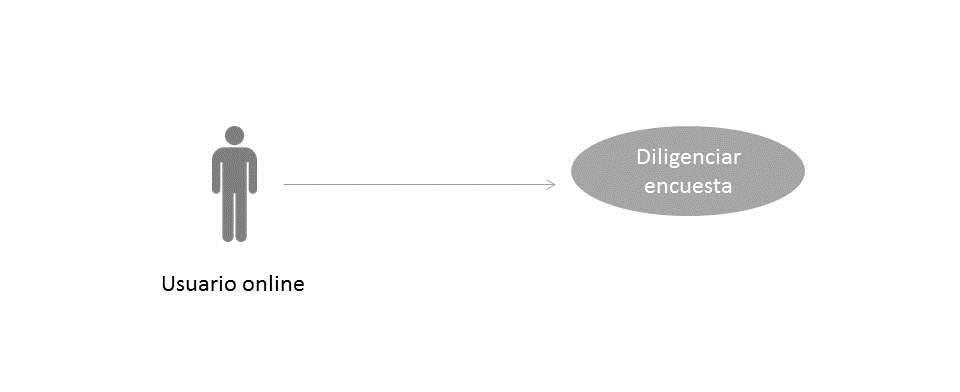
3.4 La persistencia de datos se abordará desde el uso de una base de datos relacional, además de aplicar en la autenticación de los usuarios y en el almacenamiento de las respuestas..

3.5 En el modelo de arquitectura del sistema el cual es un requisito clave por la naturaleza, ya que el proyecto es un sistema de encuestas. La arquitectura seleccionada debe garantizar Fiabilidad / disponibilidad.

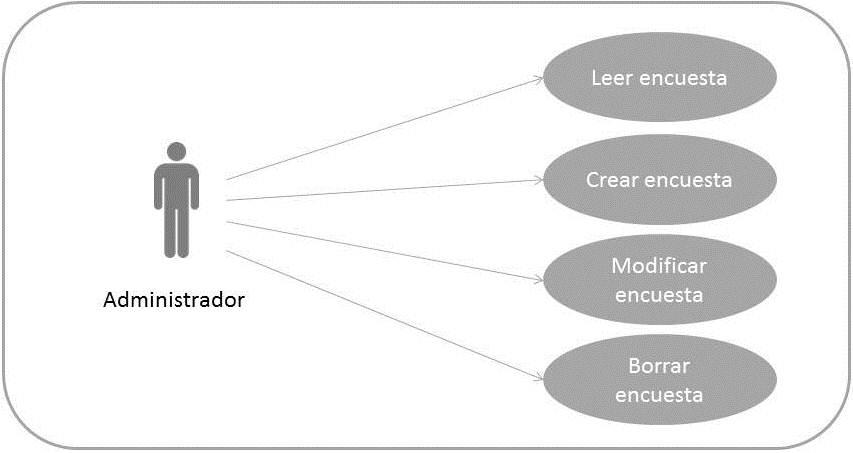
3.6 El contenido comprensible se garantiza con un diseño centrado en el usuario y con elaboración de pruebas en que los mismos participan.

# Use-Case View

4.1 Diligenciar encuesta  
Un Usuario accede a la aplicación de encuestas en línea y se logue, a continuación según su rol aparece en pantalla la encuesta correspondiente e inicia el diligenciamiento de la misma. Una vez que la encuesta ha sido diligenciada, el usuario confirma el envío, y toda la información relevante se envía al servicio de almacenamiento donde se tabula y almacena en una base de datos.



**Imagen 1** caso uso usuario

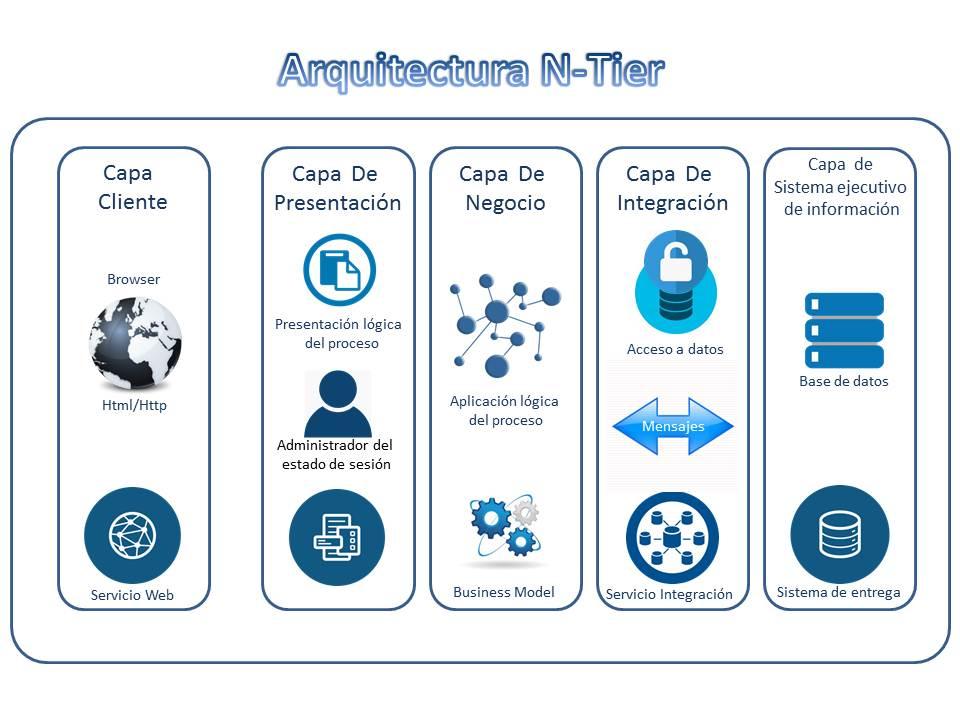


**Imagen 2** casos de uso Administrador

# Logical View

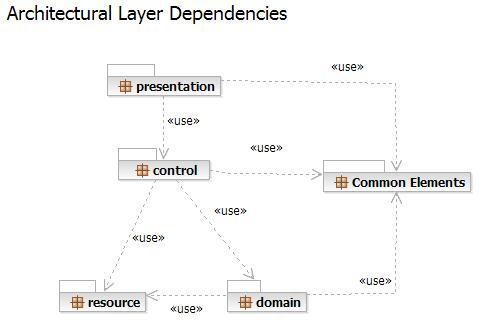
## Overview

La plataforma en línea se divide en capas basado en la arquitectura N -tier



**Imagen 3** Arquitectura N-Tier

El modelo de capas de la aplicación de encuestas en línea se basa en una estrategia de estratificación responsabilidad que asocia cada capa con una responsabilidad particular . Esta estrategia ha sido elegida porque aísla diversas responsabilidades del sistema entre sí , por lo que mejora tanto el desarrollo como también el mantenimiento del sistema .



**Imagen 4** dependencias de capas arquitectónicas

Cada capa tiene responsabilidades específicas:

* La capa de presentación es aquella que contiene la lógica de presentación, es decir que unifica con la capa de cliente e interactúa por medio de la las páginas de cara al usuario, o interfaz de usuario.
* La capa de control gestiona el acceso a la capa de dominio que permite que se ejecuten los procesos del negocio y la aplicación tenga sentido.
* La capa de recursos: es la responsable del acceso al sistema de información, es decir el acceso a las bases de datos, es decir la capa de interacción que comunica.
* La capa de dominio está relacionada con la lógica de negocio y gestiona los accesos a la capa de recursos.
* La capa de elementos comunes reúne los objetos similares reutilizados a través de todas las capas.

## Architecturally Significant Design Packages

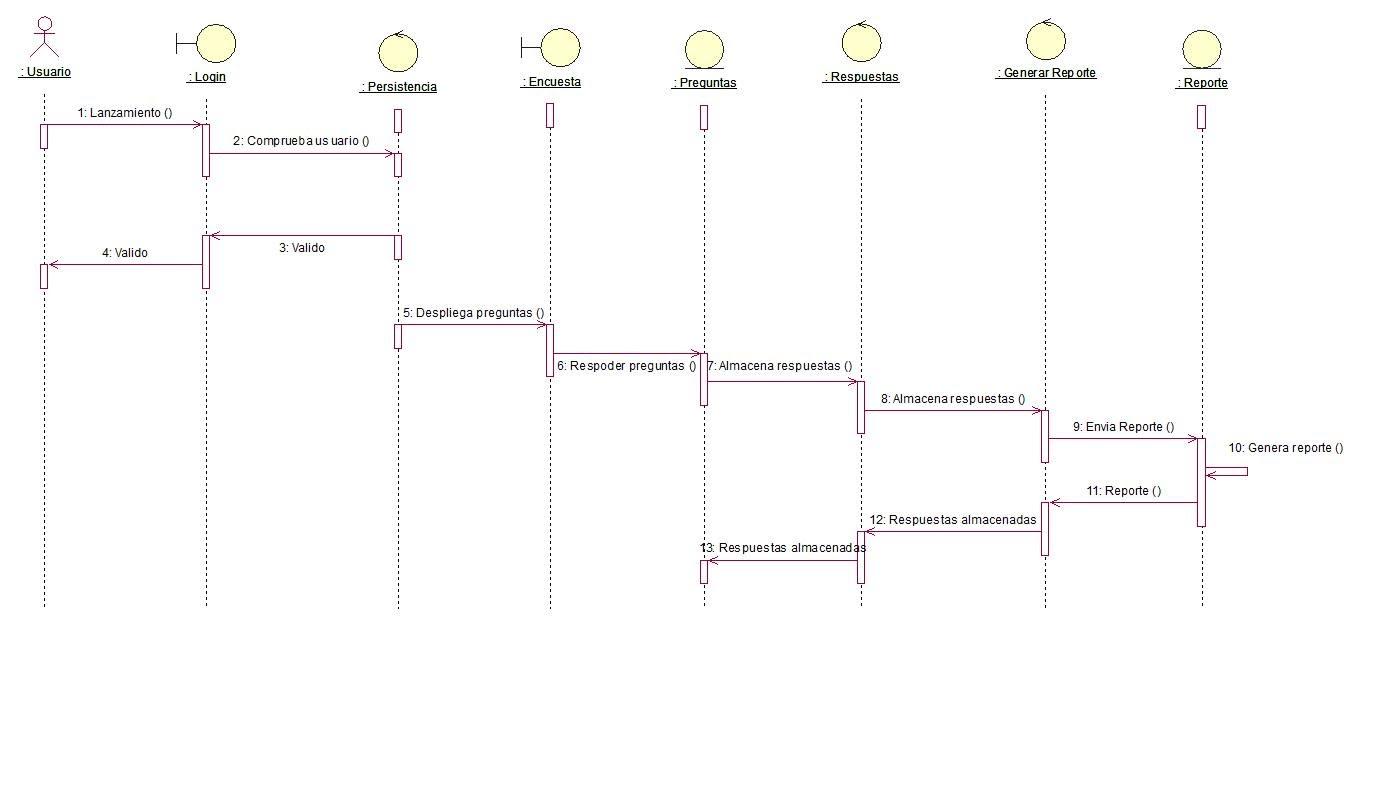
## 

## admin.jpg

**Imagen 5** Paquetes de Diseño Arquitectónico

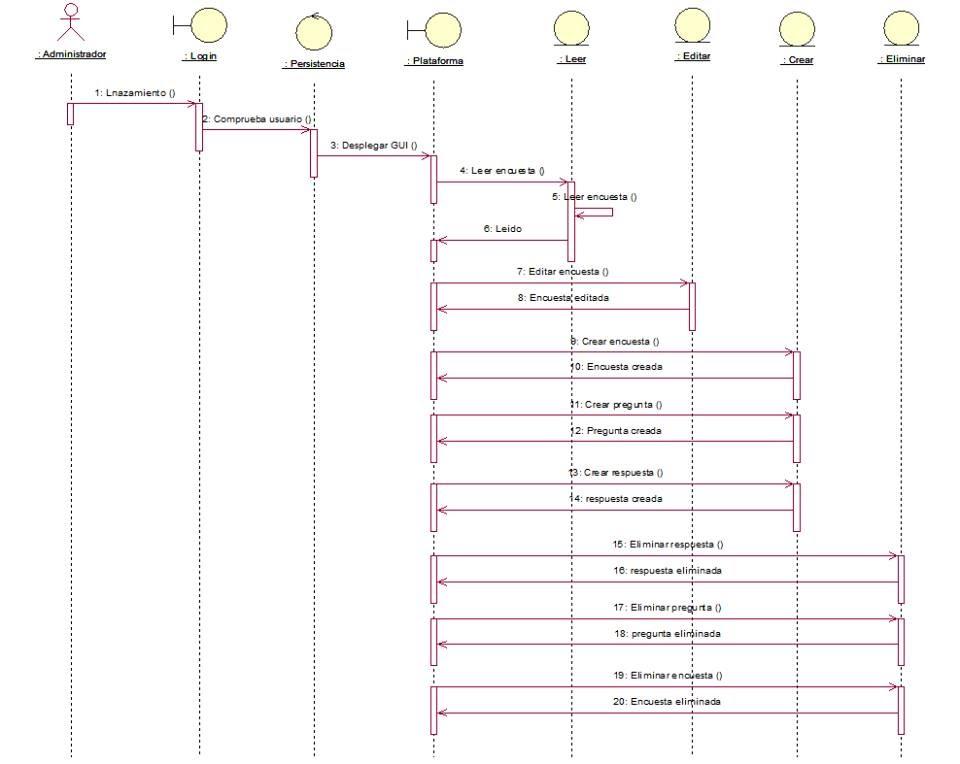
En esta vista podemos observar toda la lógica relaciona con las encuestas, ya que proporciona un completo esquema del proceso, iniciando con la interfaz la cual está relacionada con el proceso control que está dado por un logueo, seguidamente al realizar la autenticación este redirecciona a un usuario ya sea administrador o encuestado para desplegar el proceso de encuesta según el perfil, en el caso del administrador tiene acceso al proceso encuesta que tiene las entidades del CRUD, para el encuestado realizar el proceso de diligenciamiento de la encuesta.

# Deployment View



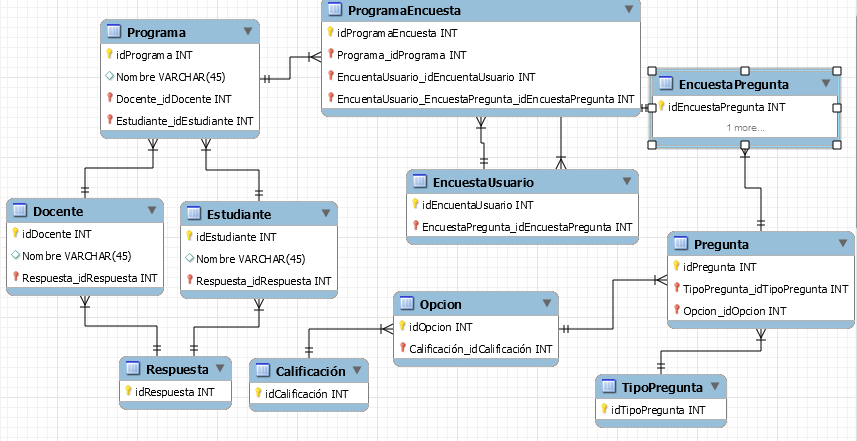
**Imagen 6** Diagrama secuencia Usuario

En el diagrama de secuencia se muestra la interacción del Usuario con los componentes, clases y subsistemas de la plataforma para poder llevar a cabo el caso de uso, logrando identificar cómo interactúan los elementos internos del componente y logran el resultado necesario para cada mensaje entrante, además de identificar patrones de interacción que pueden llegar a dificultar la actualización del software.



**Imagen 7** Diagrama secuencia Administrador

En el diagrama de secuencia se muestra la interacción del Administrador con los componentes, clases y subsistemas de la plataforma para poder llevar a cabo en los casos de uso del DRUD, logrando identificar cómo interactúan los elementos internos del componente y logran el resultado necesario para cada mensaje entrante, además de identificar patrones de interacción que pueden llegar a dificultar la actualización del software.



**Imagen 8**  Vista de datos

La vista de datos es muy útil para conocer como está estructurada la información y las relaciones internas que se realizan a el nivel más básico del manejo de los datos.

# Implementation View

## Overview

La vista de implementación representa la composición física de la aplicación en términos de subsistemas de aplicación, y Elementos de Implementación (directorios y archivos, incluyendo el código fuente, los datos y los archivos ejecutables).  
Por lo general, las capas de la vista de la implementación hacen encajar la estratificación se define en la vista lógica.

## Layers

7.2.1 Capa de presentación  
La capa de presentación contiene todos los componentes necesarios para permitir la interacción con un usuario final. Abarca la interfaz de usuario

7.2.2 Capa de Control  
La capa de control contiene todos los componentes que se utilizan para acceder a la capa de dominio o directamente a la capa de recurso cuando esto es apropiado.  
  
7.2.3 Capa de Recursos  
La capa de recursos contiene los componentes necesarios para permitir la comunicación entre la capa de negocio y de los sistemas de información empresarial (bases de datos, servicios externos, ERP, etc ...)  
  
Capa 7.2.4 Dominio  
La capa de dominio contiene todos los componentes relacionados con la lógica de negocio. Reúne todos los subsistemas que cumplan con las necesidades de un dominio de negocio en particular. También contiene el modelo de objetos de negocio.  
.  
7.2.5 Elementos comunes de Capa  
La capa Elemento de uso contiene los componentes reutilizados dentro de varias capas.

# Size and Performance

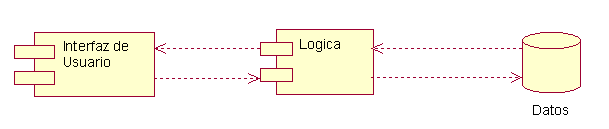
Volúmenes:

* Estimación de encuestas en línea: 2000 al día.
* Encuestas Uniquindio online usuarios estudiantes: alrededor de 16.000.
* Encuestas Uniquindio online usuarios administrativos y docentes: alrededor de 300.

Actuación:

* Tiempo para diligenciar y enviar encuesta: menos de 10 minutos sugerido.

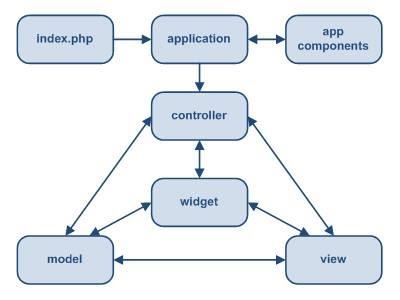
## Overview



**Imagen 9**  diagrama general de componentes

La vista de implementación que se muestra en la Figura 9 se puede relacionar con el patrón de diseño MVC (Modelo vista controlador) y es la que implementa la arquitectura a trabajar en el proyecto, separando la interfaz de usuario (vista), lógica (del negocio) y los datos (bases de datos) de la aplicación.

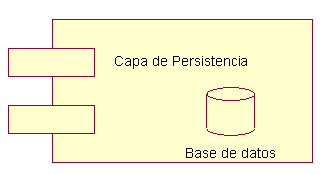
**8.2.1**



**Imagen 10**  diagrama de componentes componentes de la capa lógica

En la Figura 9 se presentan todos los componentes que interactúan en la capa lógica, los cuales se relacionan para que tanto la lógica de la presentación y del negocio estén acopladas y sincronizadas para el total funcionamiento de la aplicación, todas estas utilizando el modelo otorgado por el framework de desarrollo el cual provee elementos estructurados y en su mayoría con comunicación bidireccional.

**8.2.2 C**apa de persistencia.



**Imagen 11** diagrama de componentes de la capa de persistencia

En esta capa se maneja toda la persistencia, con una base de datos MySQL de los datos de la aplicación (Plataforma de encuestas universidad del Quindío), la capa de persistencia se acopla con la capa lógica por medio de un componente del framework Yii ilustrado en la Figura 10.

# Quality

1. La plataforma del sistema operativo deberá ser Windows XP y en adelante.

2. La interfaz de usuario del sistema deberá ser gráfica, diseñada para despliegue en navegadores web, y su manejo se realizará por medio de teclado y ratón. Deberá además ser diseñada teniendo siempre en cuenta la facilidad de uso y el hecho de que está orientada a una comunidad de usuarios con conocimientos básicos en computación.

3. El sistema contará con una ayuda plana, es decir con la documentación y especificación de los diferentes componentes que forman parte del sistema que permita resolver cualquier interrogante del usuario con respecto al uso de este.

4. El usuario deberá tener alguna forma de conocer el resultado de sus acciones, mediante mensajes de confirmación o variaciones en pantalla, como selección visible de las respuestas por medio de una marcación del punto seleccionado..

5. La codificación debe ser clara y que permita fácil entendimiento para los programadores que posteriormente le realicen ajustes o actualizaciones