Comunidad Colciencias SII

Software Architecture Document

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 06/sep/14 | 1.0 | Creación inicial del documento | Higgor Alexander Vargas  Andrés David Montoya |
| 15/sep/14 | 1.1 | Corrección de errores ortográficos y gramaticales | Higgor Alexander Vargas  Andrés David Montoya |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms, and Abbreviations 4

1.4 References 4

1.5 Overview 5

2. Architectural Representation 5

3. Architectural Goals and Constraints 5

3.1 Architectural Goals [2] 5

3.2 Constraints [2] 5

4. Use-Case View – Módulo Persona [2]. 5

5. Logical View (Diagrama de clases) [2] 6

5.1 Overview 7

5.2 Architecturally Significant Design Packages [5]. 8

5.3 Use-Case Realizations 8

6. Process View 8

7. Deployment View 9

8. Implementation View 10

9. Size and Performance [2] 12

10. Quality [2] 12

Software Architecture Document

# Introduction

La arquitectura es una descripción de las estructuras del sistema, es un artefacto que se debe analizar para determinar el logro de los atributos de calidad, provee una visión del modelo del proyecto, manifestando las decisiones de diseño [1].

En este documento se puede encontrar el desarrollo de componentes importantes en la determinación de una correcta arquitectura de software que soporte de manera adecuada el funcionamiento de la aplicación COMUNIDAD COLCIENCIAS SII.

## Purpose

El propósito de este documento es tener un consenso en el diseño de la arquitectura de software de la aplicación COMUNIDAD COLCIENCIAS SII entre los miembros del equipo de trabajo, su fin específico es la documentación de las decisiones de arquitectura que sean significativas para el sistema.

## Scope

El alcance de este documento, establece actividades que se encuentran involucradas durante el proceso de selección y aplicación de la arquitectura del software a desarrollar, define la distribución de los paquetes del sistema, garantizando que su implementación se realice siguiendo los estándares de calidad establecidos por el docente.

## Definitions, Acronyms, and Abbreviations

**HTML:** HyperText Markup Language (Lenguaje de Marcado de Hipertexto)

**PHP:** Hypertext PreProcesor (Pre Procesador de Hipertexto)

**SGBD:** Sistema de Gestion de Bases de Datos

**CSS:** Cascade Style Sheets (Hojas de estilo en cascada)

## References

1. Software Architecture in Practice 3rd Edition – Len Bass., Et All, obtenido el 27 de Agosto de 2014, obtenido desde:

<http://ptgmedia.pearsoncmg.com/images/9780321815736/samplepages/0321815734.pdf>

1. Especificaciones funcionales y técnicas – Comunidad Colciencias, Abril de 2014, Obtenida el 06 de agosto de 2014, obtenida desde:

<https://www.contratos.gov.co/consultas/detalleProcesoBM.do?numConstancia=14-6-12678>

1. DAS – Banamex, Ramés Rodríguez Vargas, 14 de mayo de 2008, obtenido el 07 de septiembre de 2014, obtenida desde:

<http://www.aplicacionesgis.com/>

1. El modelo 4+1, vistas como guía de los workshops de la ingeniería de software – Sorey Gracia, obtenido el 07 de septiembre de 2014, obtenido desde:

<http://blog.soreygarcia.me/2013/04/el-modelo-4-1-vistas-como-guia-de-los.html>

1. Entendiendo el Modelo – Vista – Controlador – CookBook, Obtenido el 07 de septiembre de 2014, obtenido desde:

http://book.cakephp.org/2.0/es/cakephp-overview/understanding-model-view-controller.html

## Overview

En las 10 secciones e este documento se detalla la arquitectura del software a desarrollar, para ello se presentan las siguientes secciones, así:

* Sección 1 resume la descripción del documento, definiciones, acrónimos y referencias.
* Sección 2 describe la representación arquitectónica del proyecto
* Sección 3 lista las metas y las restricciones del sistema.
* Sección 4 a 8 describen las diferentes vistas de la que constará el sistema
* Sección 9 Describe la tabla de requerimientos no funcionales a tener en cuenta para la puesta en funcionamiento de la aplicación.
* Sección 10 Incluye los componentes de calidad a tener en cuenta para la puesta en funcionamiento de la aplicación.

# Architectural Representation

Para este proyecto se utilizará el modelo arquitectónico “4+1” (Kruchten, 1995). Este modelo se basa en varios elementos arquitectónicos con el fin de satisfacer los requerimientos de funcionalidad y rendimiento del sistema, igualmente buscan satisfacer componentes no funcionales como lo son la confiabilidad, usabilidad, escalabilidad, entre otros [3].

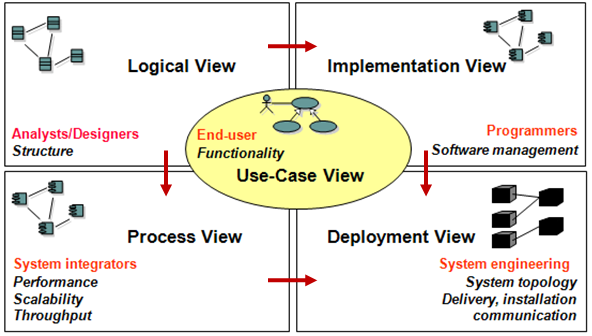


Imagen 1 – Representación gráfica de la arquitectura 4+1 [4].

# Architectural Goals and Constraints

## Architectural Goals [2]

Las metas de la arquitectura están completamente descritas en el documento de especificaciones funcionales y técnicas – Comunidad Colciencias SII, así:

1. Página 797.
2. Página 798.
3. Página 799.

## Constraints [2]

Igualmente, las restricciones del funcionamiento, están descritas en el documento de especificaciones funcionales y técnicas – Comunidad Colciencias SII, así:

1. Página 796.

# Use-Case View – Módulo Persona [2].

Para el diagrama de casos de casos de uso asignados, se puede referenciar la página 355 en el punto 2.1.9 Módulo Persona, del documento de especificaciones y requerimientos – Comuidad Colciendias, en dicho diagrama se pueden observar, además, otros casos de uso que se encuentran fuera del alcance.

# Logical View (Diagrama de clases) [2]

En la vista lógica se describen los diagramas de clases, y las realizaciones de los casos de uso que contienen los diagramas de secuencia y colaboración para cada uno de los casos de uso asignados por el docente, en el documento de especificaciones y requerimientos – Comunidad Colciencias, se encuentran los diagramas de clase separados, los cuales se analizaron y adoptaron obteniendo el siguiente diagrama de clases:

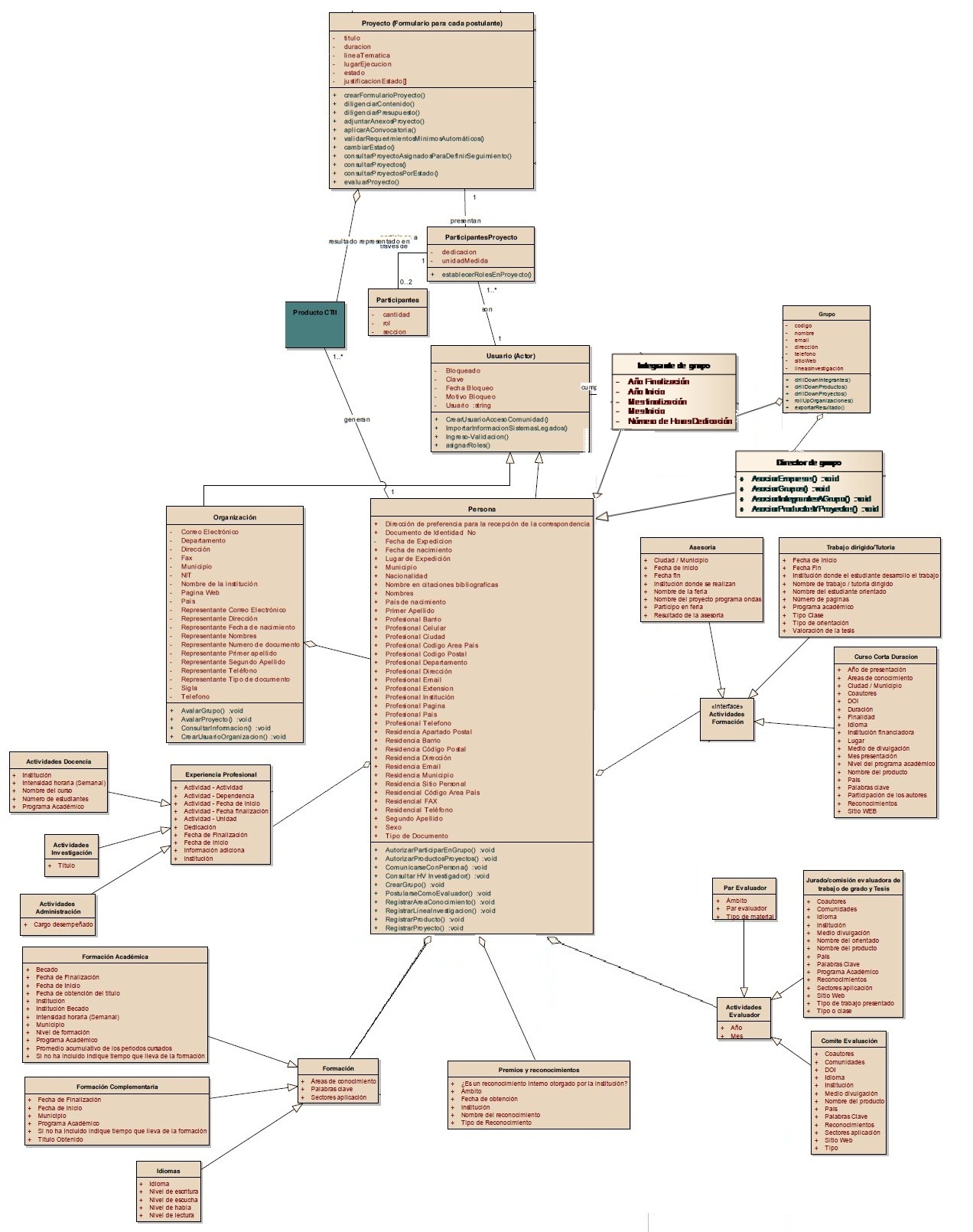


Imagen 2 – Diagrama de clases que involucra los 6 casos de uso (Diseño Adaptado) [4].

## Overview

La arquitectura está organizada bajo la estructura Modelo – Vista – Controlador, en diferentes agrupaciones lógicas, cada agrupación cumple un propósito específico. Cada agrupación contiene los componentes necesarios para cumplir con su trabajo y todos los que sean necesarios para comunicarse con las otras agrupaciones.

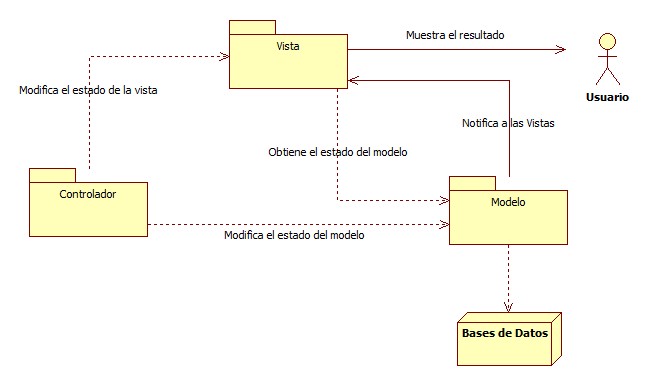


Imagen 3 – Vista General de la arquitectura [Diseño Propio]

## Architecturally Significant Design Packages [5].

**Modelo:** Contiene el núcleo de la aplicación, es la parte que implementa la lógica del negocio, es el responsable de la recuperación de los datos, su procesamiento, validación, asociación y cualquier tarea relacionada con la manipulación de los datos.

**Vista:** Presenta los datos en el modelo, separando los objetos de dicho modelo, es el responsable del uso de la información disponible ante cualquier petición que se realice por parte del usuario.

**Controlador:** es la parte que gestiona las peticiones de los usuarios, responde por la información solicitada, tanto con la ayuda del modelo como de la vista.

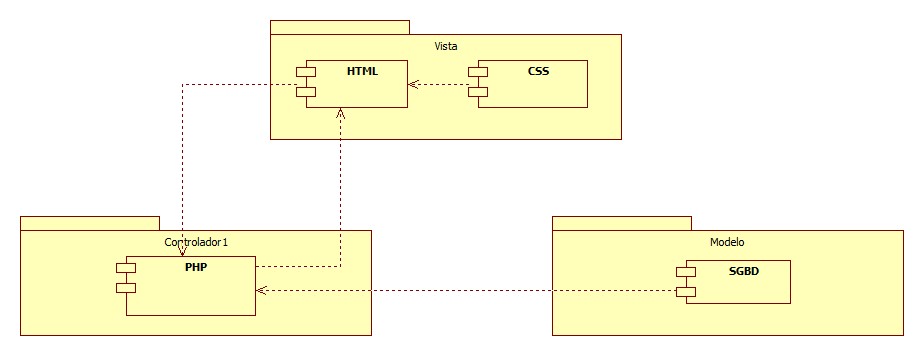


Imagen 4 – Diseño de paquetes significantes para la arquitectura [Diseño Propio]

## Use-Case Realizations

Para describir esta sección, observar el documento de especificación de casos de uso:

* 01 UCRS - RIHVP Registrar Información HV Persona
* 02 UCRS Registrar Información Formación Académica
* 03 UCRS Registrar Experiencia Investigador
* 04 UCRS Registrar Información Producción (CTII): este caso de uso, por su nivel de complejidad, el escaso tiempo y la disponibilidad de recursos humanos, es fuerte candidato a no ser implementado.
* 05 UCRS Registrar Información en Proyectos
* 06 UCRS Autorizar Participación en Grupos

# Process View

Los procesos son administrados por la capa de control, la capa de vista se encarga directamente de la interacción con el usuario y de recibir las solicitudes, en el siguiente diagrama se pueden observar las interacciones que se pueden llevar a cabo en el funcionamiento de la aplicación

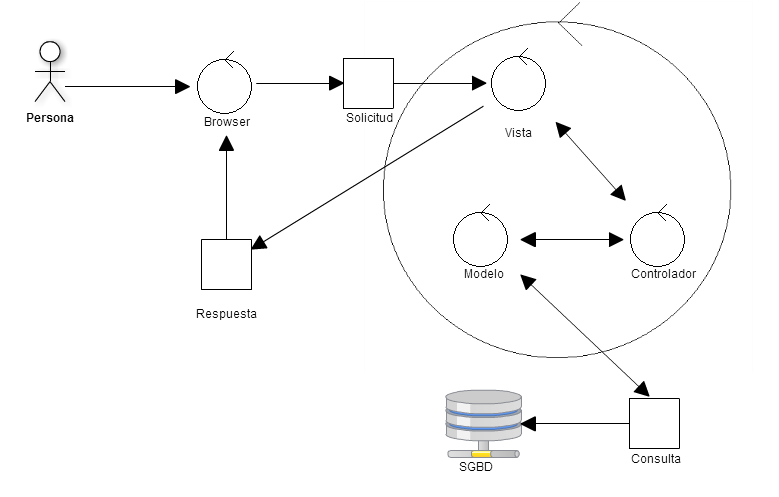


Imagen 5 – Vista de Procesos [Diseño Propio]

# Deployment View

En la vista de despliegue se puede observar la interacción entre los elementos del hardware para así cumplir con las necesidades que solicita la aplicación Comunidad Colciencias SII.

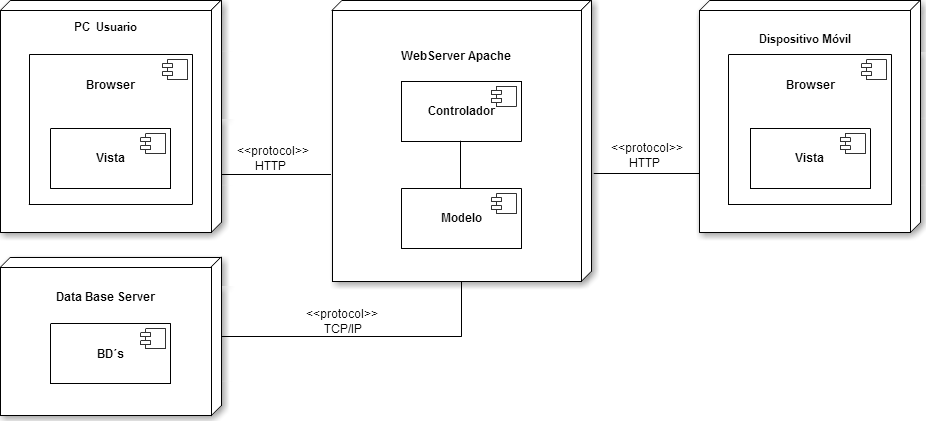


Imagen 6 – Vista de Despliegue [Diseño Propio]

**Características:**

* **Servidor** 
  + Procesador 2.50GHz,10M Cache, 6.4GT/s QPI, No Turbo, 4C, 80W, Max Mem 1333MHzProcesador Intel® Xeon® E5-2640 v2 2.00GHz, 20M Cache, 7.2GT/s QPI, Turbo, HT, 8C, 95W, Max Mem 1600MHz
  + -RDIMM de 16 GB, 1600 MT/s, bajo voltaje, clasificación doble, ancho de datos x4
  + Disco duro SATA 2 TB 7.2K RPM 3Gbps 3.5 pulgadas Cabled
  + Chasis PowerEdge R720, Intel® Xeon® E-26XX Processors
* **PC Cliente:** 
  + Procesador Core i5 de Intel de doble núcleo a 1,4 GHz (Turbo Boost de hasta 2,7 GHz) con 3 MB de caché de nivel 3 compartida
  + 4 GB de memoria LPDDR3 integrada a 1.600 MHz
  + Disco duro de 500 GB (a 5.400 rpm)
  + HD Graphics 5000 de Intel
* **Dispositivo Móvil**
  + 2 GB RAM
  + GPRS
  + WIiFi
  + CPU Quad-core 1.4 GHz

# Implementation View

La vista de implementación describe los componentes con los uqe cuenta la arquitectura de la aplicación, y su interaccion con los servicios que ofrecen, en la imagen 8, se puede observar la trazabilidad de cada uno de estos componentes, desde el diagrama de clases.

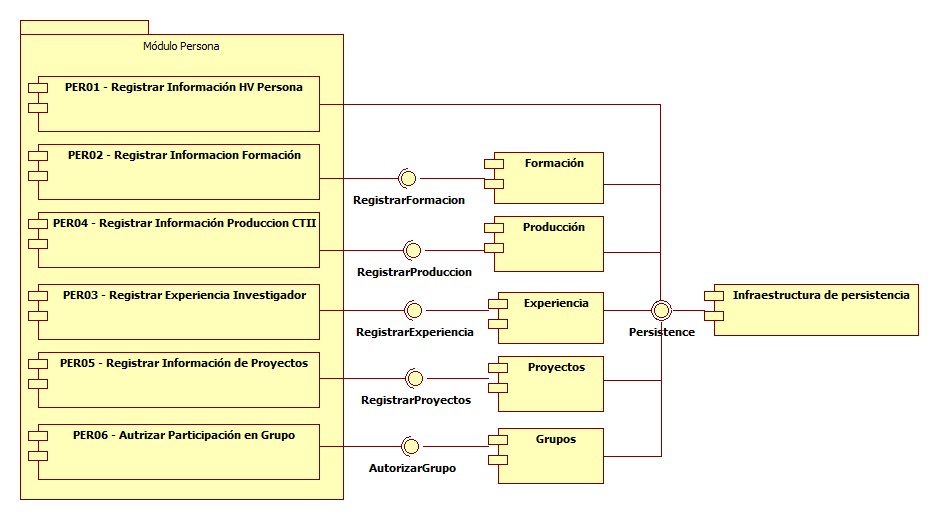


Imagen 7 – Vista de Implementación [Diseño Propio]

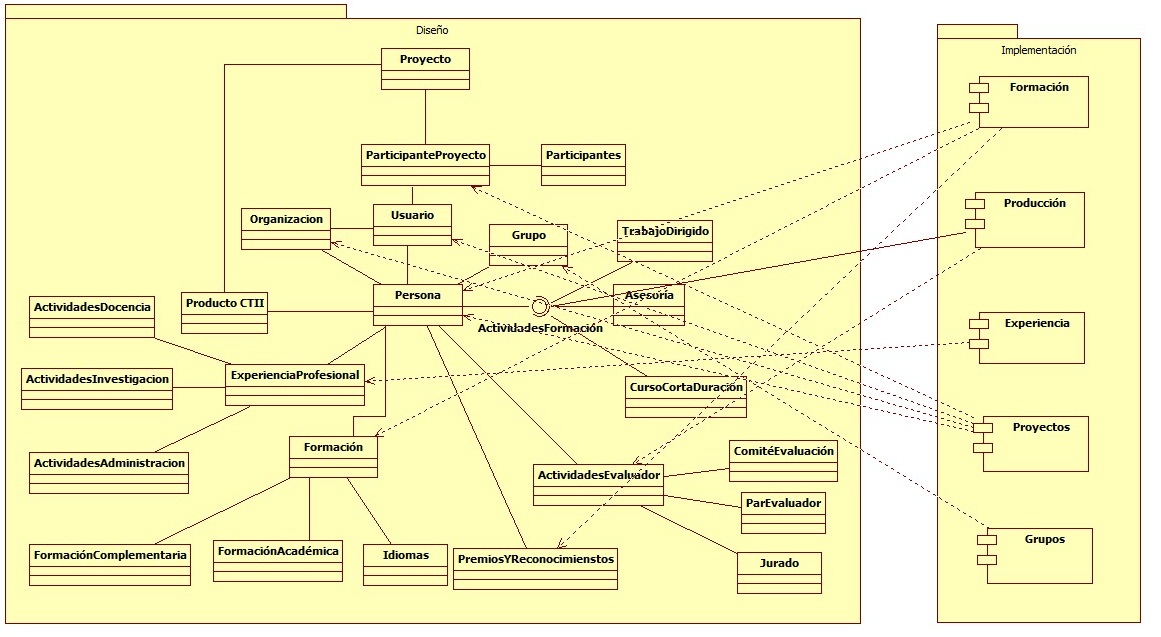


Imagen 8 – Trazabilidad - Diagrama de Componentes [Diseño Propio]

# Size and Performance [2]

La arquitectura elegida, brinda el apoyo necesario para cumplir con los requisitos de capacidad, concurrencia y almacenamiento a tener en cuenta para la puesta en servicio y operación de la aplicación Comunidad Colciencias SII, dichos requisitos son descritos en la tabla: Requerimientos no funcionales

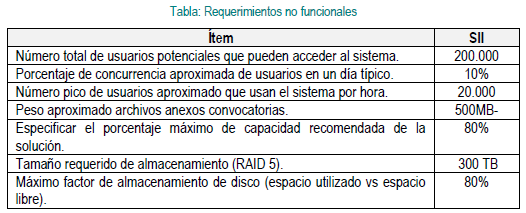


Imagen 9 – Tabla de requerimientos [2]

# Quality [2]

Para lograr un mejor aprovechamiento de la arquitectura, se especifican las caracteristicas de calidad a tener en cuenta para la puesta en servicio y operación de la aplicación Comunidad Colciencias SII, dichas caracteristicas son nombradas a continuación:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Caracteristica de Calidad** | **Disponibiliad** | | | |
| **Interés del atributo** | Implementación de la aplicación. | | | |
| **Descripción** | La aplicación debe estar disponible 24 horas x 7 días, con una disponibilidad del 99,95% | | | |
| **ID Escenario** | E001 | | | |
| **Escenario** | Módulo persona. | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estimulo** | Usuarios, actores del sistema. |  |
|  |  | **Estímulo** | Ingreso por parte de un usuario a cualquier hora y se debe permitir el funcionamiento de cualquiera de las funcionalidades de la aplicación. |  |
|  |  | **Artefacto** | Base de datos, interfaz de usuario. |  |
|  |  | **Ambiente** | Modo normal. |  |
|  |  | **Respuesta** | Mensajes de confirmación de ingreso de datos |  |
|  |  | **Medida de respuesta** | Menos de 5 segundos. |  |
|  |  |  |  |  |
| **Importancia** | Alta | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Caracteristica de Calidad** | **Usabiliad** | | | |
| **Interés del atributo** | Implementación de la aplicación. | | | |
| **Descripción** | Ofrecer un manual en línea, igualmente la aplicación debe ser altamente usable, además de ser intuitiva para reducir la capacitación al usuario final % | | | |
| **ID Escenario** | E002 | | | |
| **Escenario** | Módulo persona. | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estimulo** | Usuarios, actores del sistema. |  |
|  |  | **Estímulo** | Interacción de un usuario con cualquier elemento de la aplicación, especialmente aquellas que correspondan al módulo persona. |  |
|  |  | **Artefacto** | Interfaz de usuario |  |
|  |  | **Ambiente** | Modo normal. |  |
|  |  | **Respuesta** | Alto grado de usabiliad |  |
|  |  | **Medida de respuesta** | Alta |  |
|  |  |  |  |  |
| **Importancia** | Alta | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Caracteristica de Calidad** | **Mantenibilidad** | | | |
| **Interés del atributo** | Pruebas de la aplicación | | | |
| **Descripción** | El sistema debe permitir y/o incluir funcionalidades. | | | |
| **ID Escenario** | E003 | | | |
| **Escenario** | Codificación Módulo persona. | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estimulo** | Tester - Codificador |  |
|  |  | **Estímulo** | Inclusión o exclusión de funcionalidades que se presenten durante la ejecución y puesta en funcionamiento de la aplicación. |  |
|  |  | **Artefacto** | Interfaz de usuario |  |
|  |  | **Ambiente** | Pruebas - Codificación |  |
|  |  | **Respuesta** | Diagnóstico de errores o fallos |  |
|  |  | **Medida de respuesta** | Mínima cantidad de errores presentados por línea de código. |  |
|  |  |  |  |  |
| **Importancia** | Alta | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Caracteristica de Calidad** | **Desempeño** | | | |
| **Interés del atributo** | Pruebas e implementación | | | |
| **Descripción** | Las consultas a las bases de datos internas o externas no deben afectar el desempeño. | | | |
| **ID Escenario** | E004 | | | |
| **Escenario** | Implantacion y pruebas del Módulo persona. | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estimulo** | Tester – Usuario |  |
|  |  | **Estímulo** | El tester o el usuario generan pruebas de desempeño al momento del ingreso, consulta, modificación y eliminación de información relacionada. |  |
|  |  | **Artefacto** | Múltiples pruebas a las bases de datos relacionadas con la aplicación |  |
|  |  | **Ambiente** | Modo normal - Pruebas |  |
|  |  | **Respuesta** | Diagnóstico de errores o fallos |  |
|  |  | **Medida de respuesta** | Mínima cantidad de errores presentados por línea de código. |  |
|  |  |  |  |  |
| **Importancia** | Alta | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Caracteristica de Calidad** | **Seguridad** | | | |
| **Interés del atributo** | Pruebas – Implementación de la aplicación. | | | |
| **Descripción** | El sistema debe garantizar la autenticación de usuarios y la recuperación segura de contraseña, además debe permitir autorización de funcionalidades mediante el uso de roles, igualmente, debe contener niveles de confidencialidad de la información registradas por los usuarios, no obstante, debe permitir clasificar la información entre privada, confidencia y/o pública, sin embargo, debe contener mecanismos para la detección de virus, intrusos, y evitar el manejo de la información por fuera del flujo normal del sistema. | | | |
| **ID Escenario** | E005 | | | |
| **Escenario** | Implantación y pruebas del Módulo persona. | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estímulo** | Tester – Usuarios - Actores |  |
|  |  | **Estímulo** | El tester o el usuario generan pruebas de seguridad al momento del ingreso, asignación de roles, ataques de extracción de información, etc. |  |
|  |  | **Artefacto** | Múltiples pruebas a las bases de datos relacionadas con la aplicación. |  |
|  |  | **Ambiente** | Modo normal - Pruebas |  |
|  |  | **Respuesta** | Diagnóstico de seguridad de la plataforma de acceso a la aplicación, al igual que la definición de permisos dependiendo el nivel de seguridad. |  |
|  |  | **Medida de respuesta** | Ninguna información extraída de manera fraudulenta, usuarios con roles establecidos. |  |
|  |  |  |  |  |
| **Importancia** | Alta | | | |