



12/30/2013

Documento de Arquitectura de Software

Versión 0.1

Preparado por: Ana Irina Calvo Carvajal

Iteración II

SISTEMA AUTOMATIZADO DE INCLUSIONES

Tabla de Contenidos

1	Control del Documento	2
1.1	Historial de cambios	2
1.2	Aprobación del documento.....	2
2	Introducción	3
2.1	Propósito.....	3
2.2	Alcance	3
2.3	Definiciones, acrónimos y abreviaciones.....	4
2.4	Referencias	5
3	Representación de la arquitectura	5
4	Metas y restricciones de la arquitectura	6
4.1	Metas	6
4.2	Restricciones	6
5	Vista de casos de uso.....	6
6	Vista de lógica.....	7
6.1	Resumen.....	7
6.2	Paquetes de diseño significativos.....	8
6.3	Realizaciones de casos de uso	10
7	Vista de procesos	11
7.1	Caso de uso CU-06: Solicitar Inclusión.....	11
8	Vista de despliegue.....	11
9	Vista de implementación.....	12
9.1	Resumen.....	12
9.2	Capas	12
9.2.1	Presentación.....	12
9.2.2	Negocio	13
9.2.3	Datos	13
10	Vista de datos	14
11	Rendimiento.....	14
12	Calidad.....	15

1 Control del Documento

1.1 Historial de cambios

Versión	Fecha	Autor	Cambios realizados
0.0	11/12/2013	Ana Irina Calvo Carvajal	Creación de la plantilla del documento de arquitectura de software.
0.1	12/12/2013	Andrés González Ortiz	Inicio de la sección 'Introducción'.
0.2	13/12/2013	Andrés González Ortiz	Modificación de la sección 'Introducción', inicio de la sección 'Representación de la Arquitectura'.
0.3	15/12/2013	Andrés González Ortiz	Inicio de la sección 'Metas y restricciones de la arquitectura'.
0.4	16/12/2013	Andrés González Ortiz	Modificación de la sección 'Metas y restricciones de la arquitectura', inicio de las secciones 'Vista de casos de uso', 'Vista de lógica' y 'Vista de datos'.
0.5	17/12/2013	Andrés González Ortiz	Modificación de las secciones 'Vista de casos de uso', 'Vista de lógica' y 'Vista de datos'. Inicio de la sección 'Vista de procesos'.
1.0	17/12/2013	Andrés González Ortiz	Versión final.
1.1	23/12/2013	Andrés González Ortiz	Corregir sección 7 'Vista de procesos', corregir sección 6.3 'Realizaciones de casos de uso', corregir vista de casos de uso.
1.2	23/12/2013	Juan José Rojas Valverde	Agregar sección 11 'Rendimiento', Agregar sección 12 'Calidad'
1.3	1/2/2014	Ana Irina Calvo Carvajal	Modificar como plantilla para segunda iteración, agregar información de segunda iteración.

1.2 Aprobación del documento

Fecha	Nombre	Título	Firma
11/12/2013	Adriana Álvarez Figueroa	Profesora de la Escuela de Ingeniería en Computación del TEC (cliente)	
11/12/2013	Jaime Solano Soto	Profesor del curso de Proyecto de la Escuela de Ingeniería en Computación del TEC	

2 Introducción

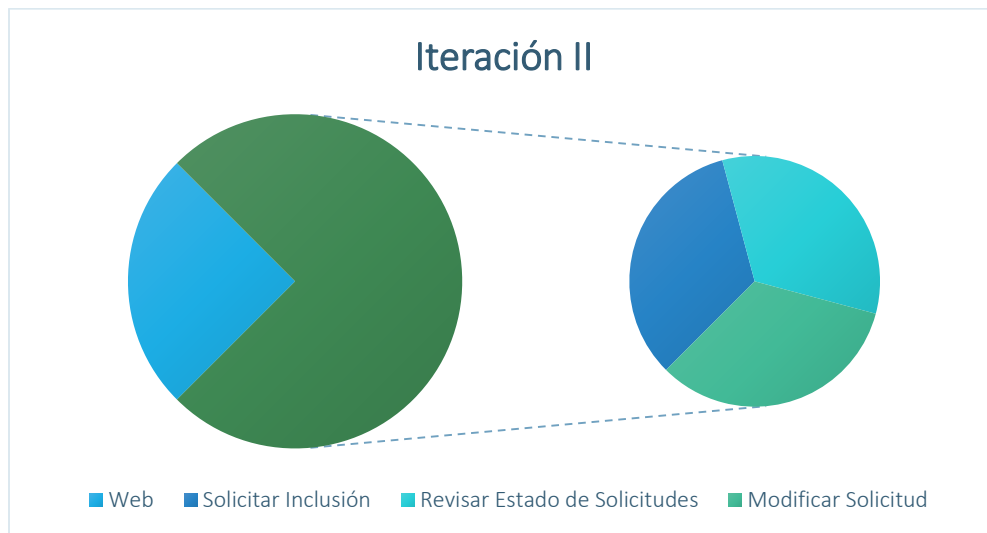
2.1 Propósito

Con este documento se busca proveer una descripción detallada de la arquitectura del Sistema Automatizado de Inclusiones de la escuela de Ingeniería en Computación del Instituto Tecnológico de Costa Rica. Para alcanzar esta meta se utilizarán varias distintas vistas y diagramas de arquitectura de software que reflejan la estructura interna del sistema, de modo que las decisiones de diseño de arquitectura tomadas para esta iteración sean claras.

En especial, se quiere detallar la arquitectura relacionada con uno de los casos de uso específicos de la segunda iteración del proyecto.

2.2 Alcance

En este documento se describe la arquitectura del sistema de inclusiones durante la segunda iteración, la cual involucra cuatro partes (una web y tres móviles) que se pueden observar en el siguiente gráfico.



La asignación de partes de la segunda iteración en el equipo de trabajo se muestra a continuación:

- Android CU-06 Solicitar Inclusión – Ana Irina Calvo Carvajal
- Android CU-07 Revisar Estado de Solicitudes – Andrés Eduardo González O
- Web CU-08 Gestionar Excepciones de Proceso – Juan José Rojas Valverde
- Android CU-09 Modificar Solicitud – Jose Arnoldo Segura Campos

En este documento de arquitectura de software se detalla la arquitectura referente a uno de los casos de uso de la segunda iteración, que es el de “Enviar Solicitudes”.

2.3 Definiciones, acrónimos y abreviaciones

Las siguientes son algunas definiciones de acrónimos y abreviaciones utilizadas a lo largo del documento, para que sea más fácil la comprensión del mismo para el lector.

Android: uno de los sistemas operativos para teléfonos inteligentes más utilizados.

App: aplicación móvil para teléfonos inteligentes.

BD: Base de datos.

CC: Centro de Cómputo.

CU-06: Caso de Uso 06, Enviar Solicitud

DAR: Departamento de Admisión y Registro.

DAS: Documento de Arquitectura de Software.

ERS: Documento de Especificación de Requerimientos de Software

IncluTec: Sistema Automatizado de Inclusiones, que es el sistema a desarrollar en el proyecto.

ITCR: Instituto Tecnológico de Costa Rica.

It.: Iteración.

MVC: Model View Controller o Modelo-Vista-Controlador es una arquitectura de software usada para implementar un programa mediante la separación de su capa lógica, su capa de datos y su capa de presentación.

TEC, Tec: Tecnológico de Costa Rica.

2.4 Referencias

Los documentos referenciados en este DAS son los siguientes:

- DVA01 – Documento de Visión y Alcance Iteración 1
- ERS02i – Especificación de Requerimientos Iteración 2 (CU-06)

3 Representación de la arquitectura

En este DAS se presenta la arquitectura del sistema como una serie de vistas, las cuales se muestran como diagramas UML desarrollados con la herramienta Visual Paradigm. Las vistas son las siguientes: **vista de casos de uso**, **vista de lógica**, **vista de procesos**, **vista de despliegue**, **vista de implementación** y **vista de datos**.

- **Vista de casos de uso:** Resume las interacciones más destacables entre el usuario y el sistema, utilizando actores y casos. Para esta iteración se trabajarán 4 casos de uso, pero en este documento solo se detallara uno de ellos (CU-06).
- **Vista de lógica:** Descompone la estructura del sistema en paquetes y subsistemas, los cuales se descomponen a su vez en clases.
- **Vista de procesos:** Describe las responsabilidades del sistema y su relación con elementos lógicos, de manera que se cumpla con la arquitectura MVC.
- **Vista de despliegue:** Muestra la manera en la que se distribuye el sistema de manera física, es decir, a nivel de hardware.
- **Vista de implementación:** Muestra la descomposición del sistema en capas lógicas y sus componentes internos, así como la manera en la que interactúan entre sí.

- **Vista de datos:** Muestra la composición interna de la BD, refiriéndose a tablas, tipos de dato y relaciones entre tablas.

4 Metas y restricciones de la arquitectura

4.1 Metas

Los objetivos de la arquitectura de este sistema son los siguientes:

- Demarcar una división lógica por capas, de manera que haya un bajo acoplamiento y la modificación del contenido de una capa no vaya a dar problemas con las otras capas.
- Definir el uso de componentes, de manera que se pueda dar reutilización de código y modularidad.
- Establecer los patrones de diseño estructurales, así como los patrones de arquitectura utilizados tanto por el sistema como por cualquier subsistema que se vaya a desarrollar.

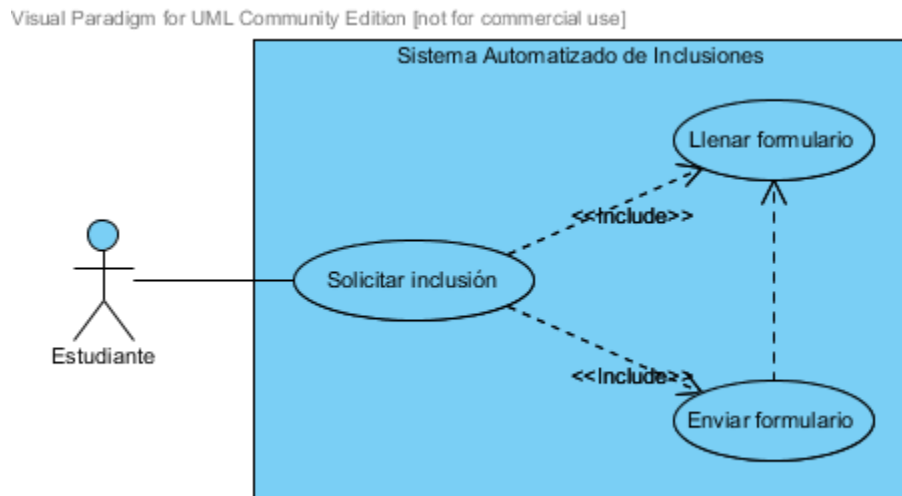
4.2 Restricciones

Se deben tomar en cuenta ciertas limitaciones respecto al diseño de la arquitectura del sistema:

- La aplicación móvil será desarrollada para el sistema operativo Android de Google.
- Se hará uso de los web services ofrecidos por el CC para acceder a información vital del DAR.

5 Vista de casos de uso

En esta sección se mostrarán las diferentes maneras en las que los actores interactúan con el sistema en este caso de uso. El caso de uso tratado en todo el documento será el CU-06 detallado en el documento ERS02i, que es el caso de uso referente a “Enviar Solicitud de Inclusión”. El siguiente diagrama muestra a grandes rasgos dicha interacción.



CU-06 Enviar Solicitudes (Aplicación Android): El estudiante llena y envía un formulario de solicitud de inclusión, realizando una solicitud de inclusión en el sistema de esa forma.

6 Vista de lógica

6.1 Resumen

El sistema se dividirá en tres paquetes lógicos principales:

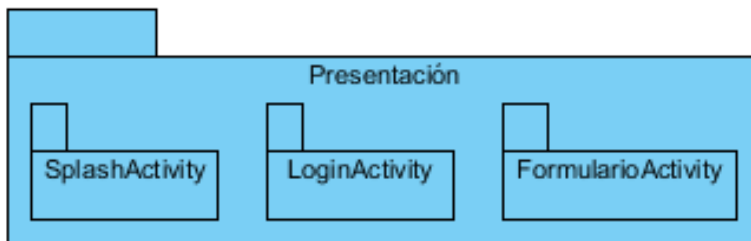
1. **Paquete de presentación:** Contiene todo lo relacionado con la interfaz gráfica de usuario y la interacción entre la interfaz y los componentes del sistema.
2. **Paquete de negocios:** Trabaja la lógica de negocio de la aplicación, así como el manejo de clases y sus conexiones con los paquetes de presentación y de datos.
3. **Paquete de datos:** Encargado de la comunicación con la base de datos, tanto para carga de información como para persistencia de datos.

Estos tres paquetes corresponden a la arquitectura MVC en el programa.

6.2 Paquetes de diseño significativos

Para este caso de uso, los paquetes de diseño significativos son los siguientes:

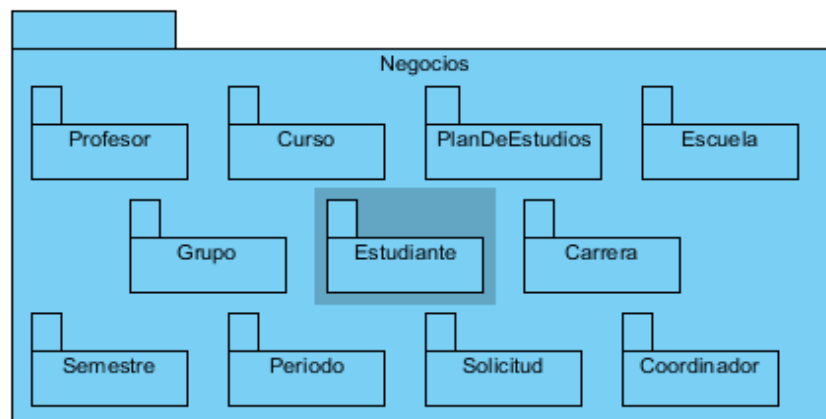
Paquete de presentación:



- SplashActivity: Pantalla de carga del app.
- LoginActivity: Pantalla para ingresar al sistema mediante el carné de estudiante y el PIN.
- FormularioActivity: Pantalla para ver la inclusión solicitada.

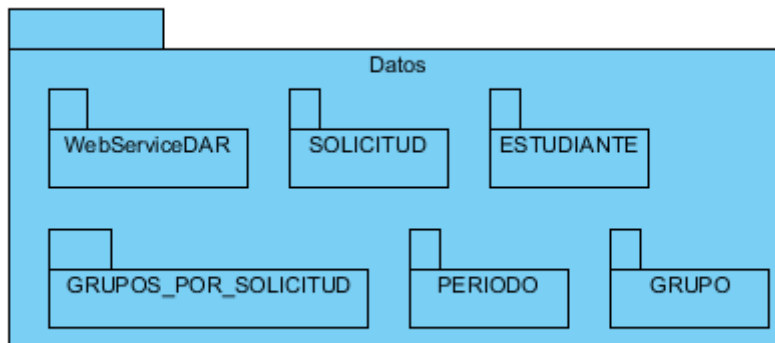
Es importante destacar que en Android se usan actividades como pantallas, por lo cual cada una de las actividades mencionadas anteriormente corresponde a una pantalla en la aplicación.

Paquete de negocios:



- Profesor: Representa a los profesores a cargo de los cursos y que debe recibir notificaciones sobre las inclusiones en sus cursos.
- Curso: Representa a la materia del plan de estudios con la que se relacionan los grupos.
- PlanDeEstudios: Corresponde a un conjunto de cursos que debe seguir el estudiante para graduarse.
- Escuela: Representa al departamento para el cual se está ejecutando el proceso de Inclusión.
- Grupo: Corresponde al grupo al cual el estudiante busca entrar mediante el proceso de inclusión.
- Estudiante: Representa al estudiante que está ejecutando y que le pertenece la solicitud de inclusión.
- Carrera: Simboliza la carrera a la cual pertenece el plan de estudios al que se le está realizando el proceso de inclusiones.
- Semestre: Corresponde al actual período del año para el que se está ejecutando el proceso.
- Período: Corresponde al período habilitado para la recepción de solicitudes al sistema.
- Solicitud: Representa al objeto de negocio que pretende buscar la inclusión para el estudiante.
- Coordinador: Representa a la persona encargada del proceso de inclusión de una determinada carrera.

Paquete de datos:



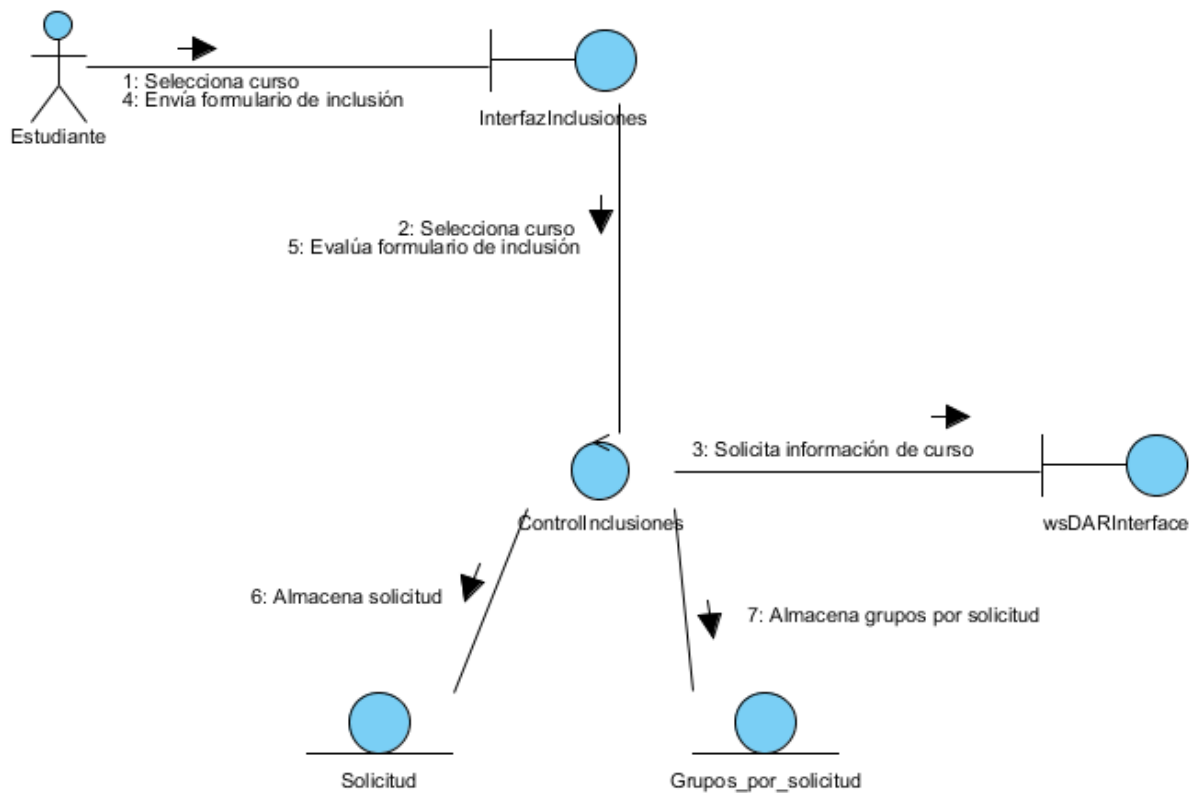
- WebServiceDAR: Web service usado por CC para acceder a los datos del DAR.
- SOLICITUD: Tabla que contiene los datos de la solicitud.

- ESTUDIANTE: Tabla que contiene algunos datos del estudiante que podrían no estar en el DAR.
- GRUPOS POR SOLICITUD: Tabla que contiene referencias a los grupos que haya seleccionado el estudiante en la solicitud de inclusión.
- PERIODO: Tabla que contiene las fechas de inicio y fin del período de recepción de solicitudes.
- GRUPO: Tabla que contiene los datos de los distintos grupos, información que podría no estar en el DAR.

Para más detalle, referirse a la vista de datos.

6.3 Realizaciones de casos de uso

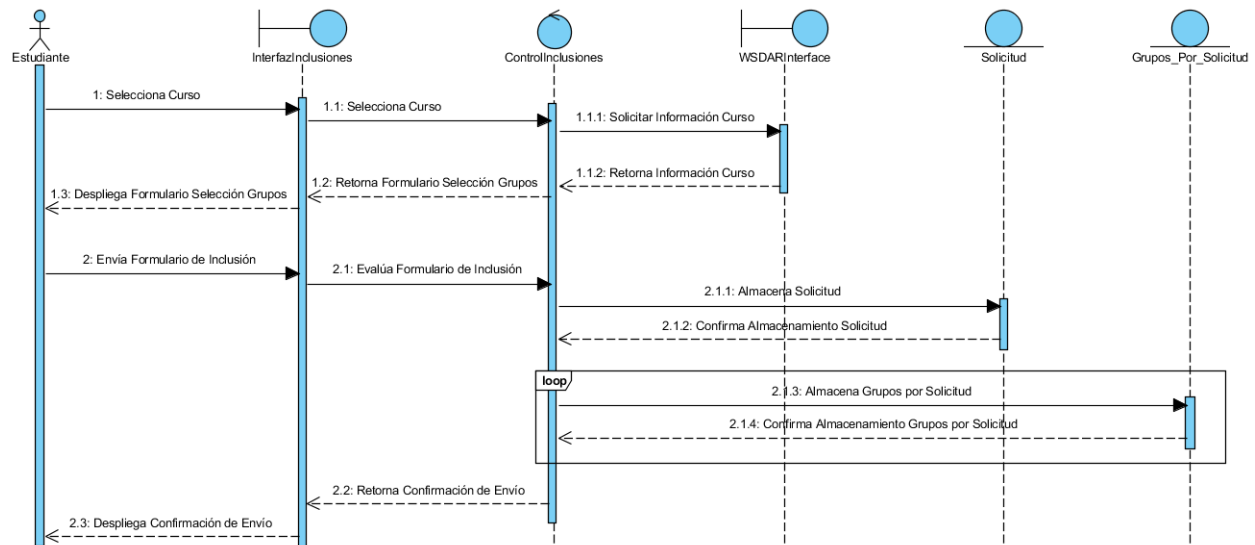
Caso de uso CU-06: Solicitar Inclusión



7 Vista de procesos

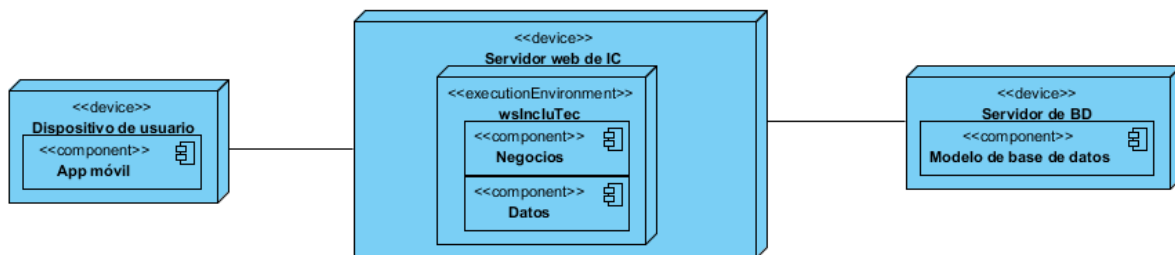
En esta sección se describen los procesos que se seguirán en esta iteración del sistema.

7.1 Caso de uso CU-06: Solicitar Inclusión



8 Vista de despliegue

Se describe la manera en la que se distribuye el sistema a nivel de hardware.

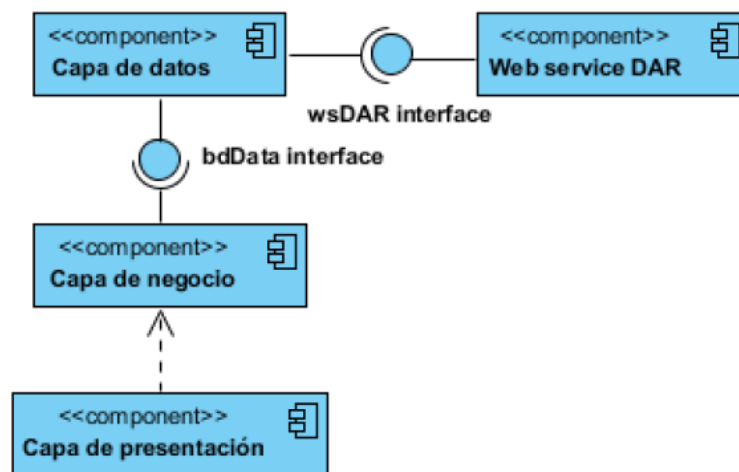


9 Vista de implementación

9.1 Resumen

El sistema está dividido en tres capas lógicas: una capa de presentación (contiene las interfaces gráficas de usuario de la aplicación web), una capa de negocios (contiene la lógica de negocios del sistema, así como los métodos para modificación y consulta de la capa de datos) y una capa de datos (maneja las conexiones a BD, las conexiones al web service de CC, y la persistencia de datos).

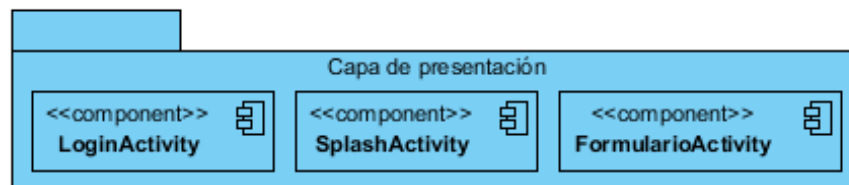
Diagrama de componentes general:



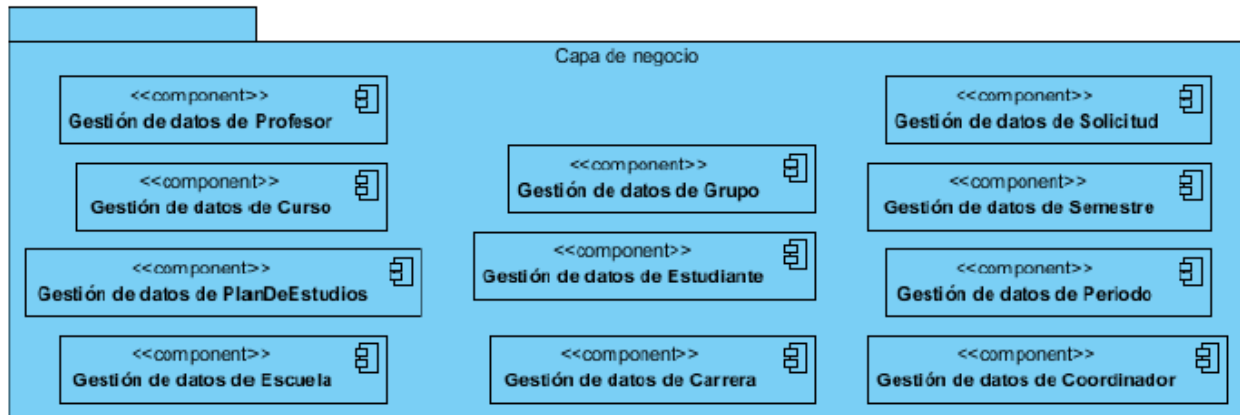
9.2 Capas

Se describe la estructura interna de cada capa lógica por medio de diagramas de componentes. Las capas usadas en este proyecto son las correspondientes a MVC.

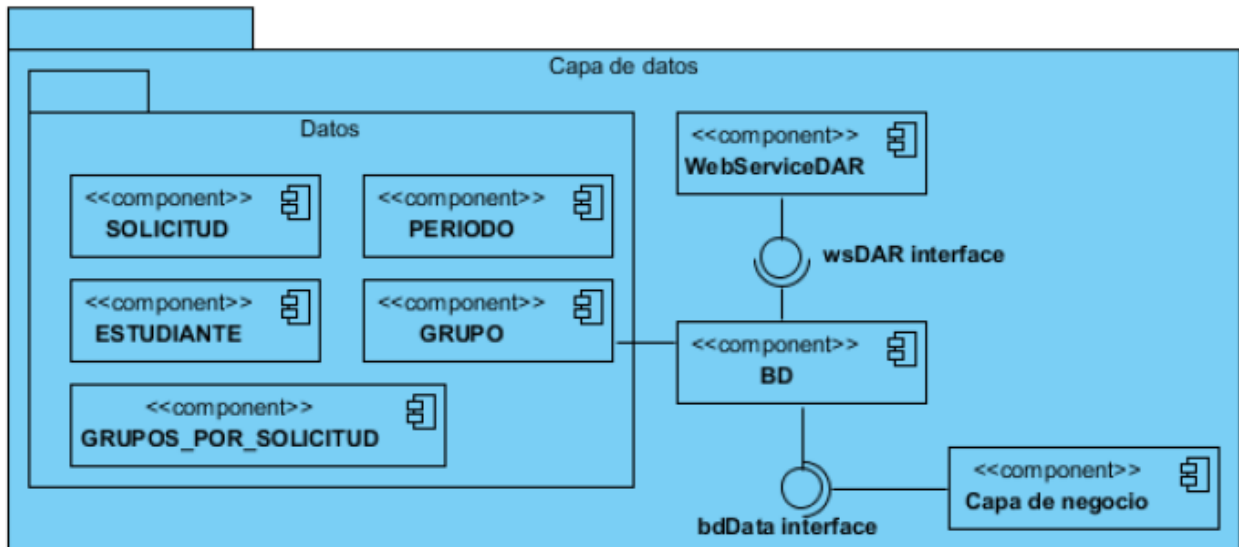
9.2.1 Presentación



9.2.2 Negocio

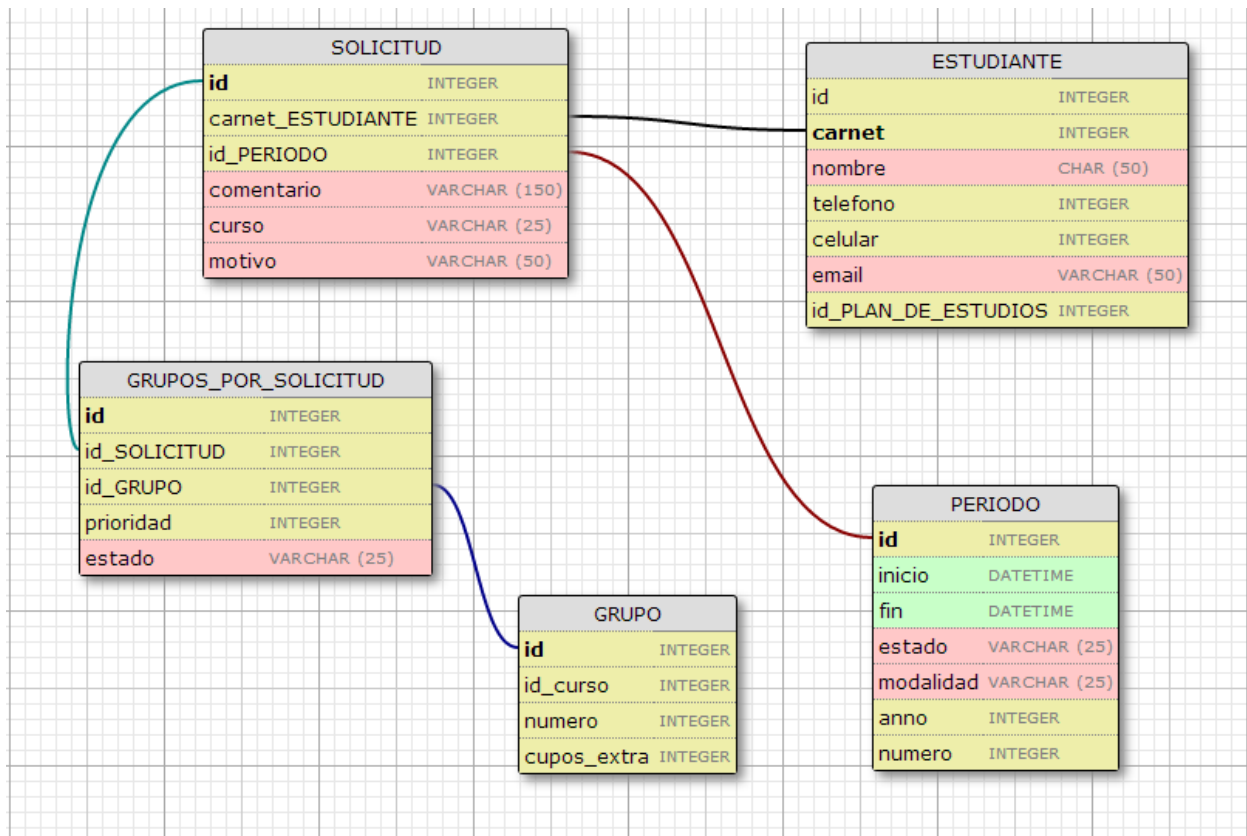


9.2.3 Datos



10 Vista de datos

Se presenta la manera en la que se manejan los datos del sistema en la BD.



11 Rendimiento

El rendimiento de la aplicación está determinado por la cantidad de usuarios que soporte los equipos donde se hará el despliegue de la aplicación, así también por las características de hardware que este tenga.

La eficiencia de la aplicación está determinada por los requerimientos del usuario y al ser implementado el sistema como una aplicación móvil, la manera de comunicación entre el cliente (dado por un app de Android) y el servidor Web, está establecida por el protocolo HTTP.

El funcionamiento de dicho protocolo es mediante dos acciones Request (por parte del cliente) y Response (por parte del servidor), por lo cual se requiere que la comunicación entre ambas partes establecida por el protocolo no sea mayor a 2 minutos para respuesta de los casos de uso planteados anteriormente.

12 Calidad

La arquitectura actual del sistema ayuda en varios aspectos esperados, además de la funcionalidad, los cuales le dan un valor agregado al sistema. Se enumeran a continuación:

1. **Tiempo de Respuesta:** Al ser el sistema modelado en capas, cada capa tiene una responsabilidad asociada bien definida, según los patrones de diseño GRASP, lo que hace que los tiempos de respuesta mejoren considerablemente. De igual forma, se considera que el mayor cuello de botella en cuanto a tiempo de respuesta de la aplicación será heredado del web service de wsDar y la base de datos de DAR, por lo que está fuera del alcance del proyecto.
2. **Mantenimiento:** otro beneficio de la arquitectura de N-Capas es el mantenimiento, ya que si en determinado momento se deben realizar los mantenimientos, solamente se van a ver afectados los componentes internos de las particiones de las capas
3. **Seguridad de los datos:** se maneja una bitácora la cual registrará todos los métodos que los usuarios realicen, además las conexiones externas a otros subsistemas también estarán determinadas con estándares de seguridad y encriptación de la información para el ocultamiento de la misma.