

# 6

## DESARROLLO DE LA APLICACIÓN



### **Contenido:**

#### **Desarrollo de la Aplicación**

- 6.1 Introducción
- 6.2 Gestión del Proyecto
- 6.3 Modelo del Negocio
- 6.4 Requisitos
- 6.5 Análisis de Diseño
- 6.6 Implementación
- 6.7 Documentación
- 6.8 Documentación



## CAPITULO VI

### 1. DESARROLLO DE LA APLICACIÓN

#### 1.1. INTRODUCCIÓN

En la sección de **Gestión del Proyecto** se muestran las planificaciones de desarrollo del proyecto, así como el cronograma de ejecución del proyecto, de construcción de la aplicación y cumplimiento de los plazos estimados.

En la sección de **Modelado del Negocio** se encuentran los artefactos utilizados de la metodología RUP para definir un modelo del negocio, modelos de objetos del negocio y el modelo del dominio.

En la sección **Requisitos** se encuentra los artefactos definidos según la metodología RUP, es decir, el documento plan de desarrollo de software, el documento visión, el documento glosario, matrices de atributos de todos los requerimientos, los casos de uso y sus especificaciones.

En la sección **Análisis/Diseño** se muestran tanto el modelo de análisis/diseño (diagrama de clases) como el modelo de datos (modelo entidad – relación).

En la sección **Implementación** se muestran los prototipos de interfaces de usuario de la aplicación.



A continuación se detalla las Tecnologías y Herramientas usadas en el Desarrollo del Sistema.

- ✓ **Base de Datos:** Oracle Data Base 10g versión 10.2.0.1.0
- ✓ **Plataforma de Desarrollo:** Java con JDK 1.5
- ✓ **IDE de Desarrollo:** JDeveloper 11g.
- ✓ **Diseñador de Reportes:** iReport-3.0.2
- ✓ **Framework de Desarrollo:** Oracle ADF
- ✓ **Servidor de Aplicaciones:** Web logic.

## **1.2. GESTIÓN DEL PROYECTO**

En esta sección se detalla la planificación inicial del proyecto para la fase de inicio y la fase de elaboración (según la definición de la metodología RUP).

### **PLAN DE DESARROLLO DEL SOFTWARE**

#### **INTRODUCCIÓN**

Este Plan de Desarrollo de Software es una versión preliminar preparada para ser incluida en la propuesta elaborada como respuesta al trabajo final de grado previo a la obtención del Título de Ingeniera en Sistemas Computacionales de la Facultad de Ingeniería en Ciencias Aplicadas de la Universidad Técnica del Norte.



El enfoque de desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de RUP.

## **PROPÓSITO**

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él se describe el enfoque de desarrollo de software.

Los usuarios del Plan de Desarrollo de Software son:

- ✓ El jefe de proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.
- ✓ Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y que otras actividades dependen de ello.

## **ALCANCE**

El Plan de Desarrollo de Software describe el plan global usado para el desarrollo de la **“IMPLEMENTACIÓN DE UNA BASE DE DATOS DISTRIBUIDA EN EL SISTEMA ACADÉMICO DE LA FICA”**. Durante el proceso de desarrollo en el artefacto “Visión” se definen las características del producto a desarrollar, lo cual constituye la base para la planificación de las iteraciones.



Para la versión 1.0 del Plan de Desarrollo de Software nos hemos basado en la captura de requisitos por medio del stakeholder representante para hacer una estimación aproximada, una vez comenzado el proyecto y durante la fase de Inicio se generará la primera versión del artefacto “Visión”, el cual se utilizará para refinar este documento.

Posteriormente, el avance del proyecto y el seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas.

## **RESUMEN**

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

**Vista General del Proyecto:** Proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.

**Organización del Proyecto:** Describe la estructura organizacional del equipo de desarrollo.

**Gestión del Proceso:** Explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.

**Planes y Guías de aplicación:** Proporciona una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.



## **VISTA GENERAL DEL PROYECTO**

### **PROPÓSITO, ALCANCE Y OBJETIVOS**

La información que a continuación se incluye a sido extraída de las diferentes reuniones que se han celebrado con el stakeholder desde el inicio del proyecto.

El proyecto debe proporcionar una respuesta para el desarrollo de todos los módulos implicados en la **“IMPLEMENTACIÓN DE UNA BASE DE DATOS DISTRIBUIDA EN EL SISTEMA ACADÉMICO DE LA FICA”**. Estos módulos se pueden diferenciar en:

- Roles
- Usuarios
- Estudiantes
- Facultades
- Docentes
- Parámetros
- Escuelas
- Matriculas
- Ingreso de Notas
- Periodos Académicos
- Crear Paralelos



## **SUPOSICIONES Y RESTRICCIONES**

Las suposiciones y restricciones respecto del sistema, y que se derivan directamente de las entrevistas con el stakeholder son:

- a) Debe contemplarse las implicaciones de los siguientes puntos críticos:
  - Sistemas seguros: protección de información, seguridad en las transmisiones de datos, etc.
  - Gestión de flujos de trabajo, seguridad de las transacciones e intercambio de información.
- b) La automatización de la gestión interna del registro debe ajustarse a la legislación vigente.
- c) El módulo de matriculación debe ser desarrollado como un sistema independiente para ser utilizado por todas las Facultades.

Como es natural, la lista de suposiciones y restricciones se incrementará durante el desarrollo del proyecto, particularmente una vez establecido el artefacto “Visión”.

## **ENTREGABLES DE PROYECTO**

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituyen la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto.



Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

### **1) Plan de Desarrollo del Software**

Es el presente documento.

### **2) Modelo de Casos de Uso del Negocio**

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas etc.). Permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representa con un Diagrama de Casos de Uso usando estereotipos específicos para este modelo.

### **3) Modelo de Objetos del Negocio**

Es un modelo que describe la realización de cada caso de uso del negocio, estableciendo los actores internos, la información que en términos generales manipulan y los flujos de trabajo asociados al caso de uso del negocio. Para la representación de este modelo se utilizan Diagramas de Colaboración (para mostrar actores





externos, internos y las entidades (información) que manipulan, un Diagrama de Clases para mostrar gráficamente las entidades del sistema y sus relaciones, y Diagramas de Actividad para mostrar los flujos de trabajo.

#### **4) Glosario**

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada. .

#### **5) Modelo de Casos de Uso**

El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Se representa mediante Diagramas de Casos de Uso.

#### **6) Visión**

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

#### **7) Especificaciones de Casos de Uso**

Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, post-condiciones, flujo de eventos, requisitos no-funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá



adjuntarse una representación gráfica mediante un Diagrama de Actividad.

### **8) Especificaciones Adicionales**

Este documento capturará todos los requisitos que no han sido incluidos como parte de los casos de uso y se refieren requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc.

### **9) Prototipos de Interfaces de Usuario**

Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.



### **10) Modelo de Análisis y Diseño**

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto.

### **11) Modelo de Datos**

Previendo que la persistencia de la información del sistema será soportada por una base de datos distribuida, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.).

### **12) Modelo de Implementación**

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento).



### **13) Modelo de Despliegue**

Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

### **14) Casos de Prueba**

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

### **15) Solicitud de Cambio**

Los cambios propuestos para los artefactos se formalizan mediante este documento. Mediante este documento se hace un seguimiento de los defectos detectados, solicitud de mejoras o cambios en los requisitos del producto. Así se provee un registro de decisiones de cambios, de su evaluación e impacto, y se asegura que éstos sean conocidos por el equipo de desarrollo. Los cambios se establecen respecto al último estado del conjunto de los artefactos en un momento determinado del proyecto.



### **16) Plan de Iteración**

Es un conjunto de actividades y tareas ordenadas temporalmente, con recursos asignados, dependencias entre ellas. Se realiza para cada iteración, y para todas las fases.

### **17) Evaluación de Iteración**

Este documento incluye la evaluación de los resultados de cada iteración, el grado en el cual se han conseguido los objetivos de la iteración, las lecciones aprendidas y los cambios a ser realizados.

### **18) Lista de Riesgos**

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

### **19) Manual de Instalación**

Este documento incluye las instrucciones para realizar la instalación del producto.

### **20) Material de Apoyo al Usuario Final**

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Guías del Usuario, Guías de Operación, Guías de Mantenimiento.

### **21) Producto**

Los ficheros del producto empaquetados y almacenados en un CD con los mecanismos apropiados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de



construcción es desarrollado incremental e iterativamente, obteniéndose un nuevo reléase al final de cada iteración.

Los artefactos 19, 20 y 21 se generarán a partir de la fase de construcción, con lo cual se han incluido aquí sólo para dar una visión global de todos los artefactos que se generarán en el proceso de desarrollo.

## **EVOLUCIÓN DEL PLAN DE DESARROLLO DEL SOFTWARE**

El Plan de Desarrollo del Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.

## **ORGANIZACIÓN DEL PROYECTO**

### **PARTICIPANTES EN EL PROYECTO.**

El personal participante en el proyecto se encuentra formado por los siguientes puestos de trabajo y personal asociado:

**Jefe de Proyecto:** Ing. Miguel Orquera.

**Arquitecto de Software:** Egda. Verónica Alexandra López.

**Ingeniero de Software:** Egda. Verónica Alexandra López.

**Programador:** Egda. Verónica Alexandra López.



## **INTERFACES EXTERNAS**

La empresa definirá los participantes del proyecto que proporcionarán los requisitos del sistema, y entre ellos quiénes serán los encargados de evaluar los artefactos de acuerdo a cada módulo y según el plan establecido.

El equipo de desarrollo interactuará activamente con los participantes de la empresa para especificación y validación de los artefactos.

## **ROLES Y RESPONSABILIDADES**

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP

<b>PUESTO</b>	<b>RESPONSABILIDADES</b>
JEFE DE PROYECTO	El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.
ARQUITECTO DE SOFTWARE	Se encargará de supervisar el establecimiento de la Arquitectura del Sistema, es decir, definir la vista arquitectónica, los estilos arquitectónicos, el patrón de arquitectura y la arquitectura Tecnológica a utilizar.
INGENIERO DE SOFTWARE	Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar



	modelos de implementación y despliegue. Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos. Encargado además de la puesta en producción.
PROGRAMADOR	Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario.
TESTER	Se encargará de la realización de las pruebas funcionales, de conectividad y rendimiento del sistema.

**Roles y Responsabilidades**

## **GESTIÓN DE PROCESOS**

### **ESTIMACIONES DEL PROYECTO**

El presupuesto del proyecto y los recursos involucrados se adjuntan en un documento por separado.

### **PLAN DEL PROYECTO**

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.





## **PLAN DE FASES**

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra la distribución de tiempos y el número de iteraciones de cada fase.

<b>FASE</b>	<b>Nro INTERACIONES</b>	<b>DURACIÓN</b>
Fase de Inicio	1	4 semanas
Fase de Elaboración	2	3 semanas
Fase de Construcción	3	9 semanas
Fase de Transición	2	4 semanas

**Plan de Fase del Proyecto**

Los hitos que marcan el final de cada fase se describen en la siguiente tabla:

<b>DESCRIPCIÓN</b>	<b>HITOS</b>
Fase de Inicio	En esta fase se desarrollará los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento al Plan de Desarrollo de Software. La aceptación del cliente/usuario del artefacto Visión y el Plan de Desarrollo de Software marcan el final de esta fase.
Fase de Elaboración	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y/o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera fase de Construcción deben ser analizados y diseñados (en el Modelo de Análisis/Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase.



Fase de Construcción	Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis/Diseño. El producto se construye en base a dos iteraciones, cada una produciendo un realce a la cual se le aplican las pruebas y se valida con el cliente/usuario. Se comienza la elaboración del material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la release 3.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para la prueba beta.
Fase de Transición	En esta fase se prepararán dos releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto

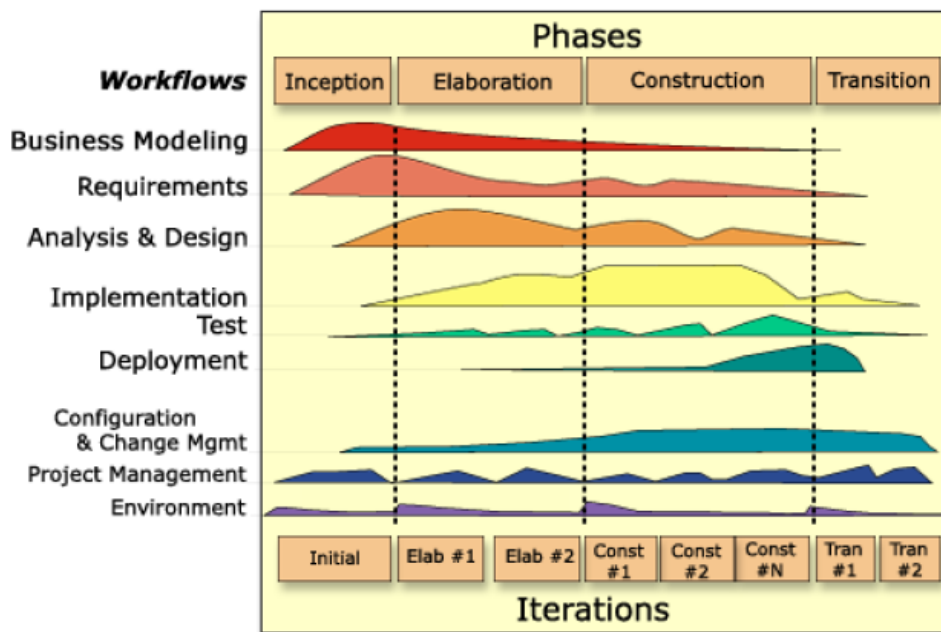
**Hitos de las Fases del Proyecto**

## **CALENDARIO DEL PROYECTO**

A continuación se presenta un calendario de las principales tareas del proyecto.

Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto.

La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (workflow) en un momento determinado del desarrollo.



1. Interacciones del Proyecto

Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero eso no quita la posibilidad de su posterior refinamiento y cambios

Disciplinas / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
<b>Modelado del Negocio</b>		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 1	Semana 3
<b>Requisitos</b>		
Glosario	Semana 1	Semana 3
Visión	Semana 2	Semana 3
Modelo de Casos de Uso	Semana 3	siguiente fase
Especificación de Casos de Uso	Semana 3	siguiente fase
Especificaciones Adicionales	Semana 3	siguiente fase
<b>Análisis / Diseño</b>		



**“FRAGMENTACIÓN DE TABLAS EN BASES DE DATOS DISTRIBUIDAS”**

Modelo de Análisis / Diseño	Semana 2	siguiente fase
Modelo de Datos	Semana 2	siguiente fase
<b>Implementación</b>		
Prototipos de Interfaces de Usuario	Semana 3	siguiente fase
Modelo de Implementación	Semana 3	siguiente fase
<b>Pruebas</b>		
Casos de Pruebas Funcionales	Semana 3	siguiente fase
<b>Despliegue</b>		
Modelo de Despliegue	Semana 3	siguiente fase
<b>Gestión de Cambios y Configuración</b>	Durante todo el proyecto	
<b>Gestión del proyecto</b>		
Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones	Semana 1	Semana 3
<b>Ambiente</b>	Durante todo el proyecto	

**Calendario del proyecto. Fase de Inicio**

Disciplinas / Artefactos generados o modificados durante la Fase de Elaboración	Comienzo	Aprobación
<b>Modelado del Negocio</b>		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 1	Aprobado
<b>Requisitos</b>		
Glosario	Semana 1	Aprobado
Visión	Semana 2	Aprobado
Modelo de Casos de Uso	Semana 3	Semana 5
Especificación de Casos de Uso	Semana 3	Semana 5
Especificaciones Adicionales	Semana 3	Semana 5
<b>Análisis / Diseño</b>		
Modelo de Análisis / Diseño	Semana 2	Revisar en cada iteración
Modelo de Datos	Semana 2	Revisar en cada iteración
<b>Implementación</b>		
Prototipos de Interfaces de Usuario	Semana 3	Revisar en cada iteración
Modelo de Implementación	Semana 3	Revisar en cada iteración
<b>Pruebas</b>		
Casos de Pruebas Funcionales	Semana 3	Revisar en cada iteración



<b>Despliegue</b>		
Modelo de Despliegue	Semana 3	Revisar en cada iteración
<b>Gestión de Cambios y Configuración</b>	Durante todo el proyecto	
<b>Gestión del proyecto</b>		
Plan de Desarrollo del Software en su versión 2.0 y planes de las Iteraciones	Semana 4	Revisar en cada iteración
<b>Ambiente</b>	Durante todo el proyecto	

**Calendario del Proyecto. Fase de Elaboración**

## **SEGUIMIENTO Y CONTROL DEL PROYECTO**

### **Gestión de Requisitos**

Los requisitos del sistema son especificados en el artefacto Visión. Cada requisito tendrá una serie de atributos tales como importancia, estado, iteración donde se implementa, etc. Estos atributos permitirán realizar un efectivo seguimiento de cada requisito. Los cambios en los requisitos serán gestionados mediante una Solicitud de Cambio, las cuales serán evaluadas y distribuidas para asegurar la integridad del sistema y el correcto proceso de gestión de configuración y cambios.

### **Control de Plazos**

El calendario del proyecto tendrá un seguimiento y evaluación semanal por el jefe de proyecto.



### **Control de Calidad**

Los defectos detectados en las revisiones y formalizados también en una Solicitud de Cambio tendrán un seguimiento para asegurar la conformidad respecto de la solución de dichas deficiencias. Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las guías de revisión y checklist (listas de verificación) incluidas en RUP.

### **Gestión de Riesgos**

A partir de la fase de Inicio se mantendrá una lista de riesgos asociados al proyecto y de las acciones establecidas como estrategia para mitigarlos o acciones de contingencia. Esta lista será evaluada al menos una vez en cada iteración.

### **Gestión de Configuración**

Se realizará una gestión de configuración para llevar un registro de los artefactos generados y sus versiones. También se incluirá la gestión de las Solicitudes de Cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles a todo los participantes en el proyecto. Al final de cada iteración se establecerá una baseline (un registro del estado de cada artefacto, estableciendo una versión), la cual podrá ser modificada sólo por una Solicitud de Cambio aprobada.

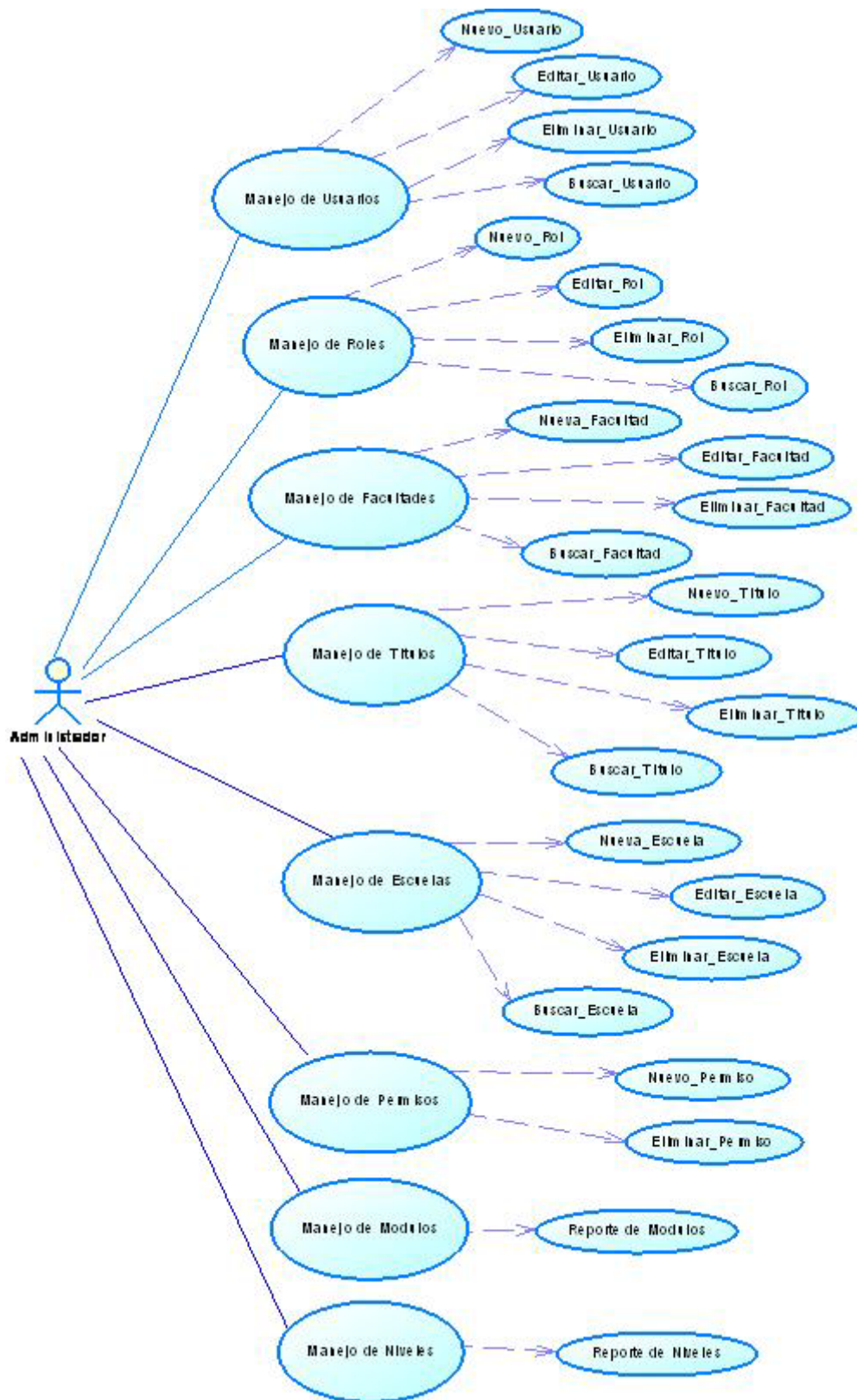


### **1.3. MODELADO DEL NEGOCIO**

A continuación se presentan los modelos definidos en RUP como modelo del negocio, modelo de datos y modelo de análisis y diseño

- Administrador
- Usuario
- Docente
- Secretaria Escuela

## ADMINISTRADOR

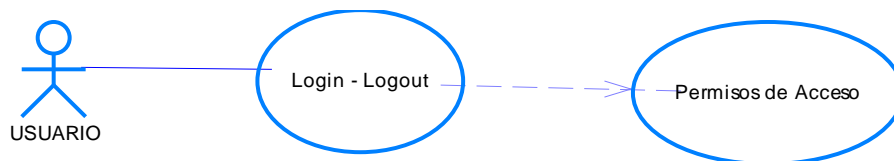


2. Diagrama de caso de Uso del Administrador. Fuente Propia



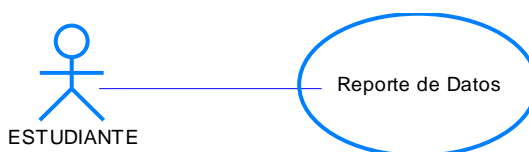


## **USUARIO**



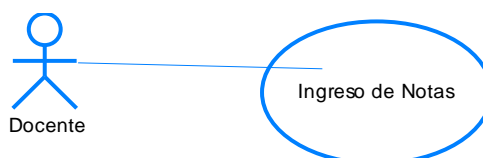
**3. Diagrama de caso de Uso del Usuario. Fuente Propia**

## **ESTUDIANTE**



**4. Diagrama de caso de Uso del Estudiante. Fuente Propia**

## **DOCENTE**



**5. Diagrama de caso de Uso del Docente. Fuente Propia**



## SECRETARIA



6. Diagrama de caso de Uso de la Secretaria de la Escuela. Fuente Propia



## **1.4. REQUISITOS**

A continuación se presentan las herramienta utilizadas para declarar los requisitos software, es decir, el documento plan de desarrollo software, el documento visión, el documento glosario y tanto las especificaciones de los casos de uso.

### **VISIÓN**

#### **INTRODUCCIÓN**

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo a los requisitos del sistema

#### **PROPÓSITO**

El propósito de éste documento es recoger, analizar y definir las necesidades de alto nivel y las características del sistema de la “Implementación de una Base de Datos Distribuida en el Sistema Académico e la FICA”.

Los detalles de cómo el sistema cubre los requerimientos se pueden observar en la especificación de los casos de uso y otros documentos adicionales.



## **ALCANCE**

El documento Visión se ocupa, como ya se ha detallado, el sistema de “Implementación de una Base de Datos Distribuida en el Sistema Académico e la FICA”, dicho sistema será desarrollado por Verónica Alexandra López Tulcanaza.

El sistema permitirá a los encargados de la empresa controlar todo lo relativo al Sistema Académico (ingreso de estudiantes, docentes, materias, crear periodos académicos, matriculación etc).

## **POSICIONAMIENTO**

### **OPORTUNIDAD DE NEGOCIO**

Este sistema permitirá a la empresa informatizar el control de sus actividades (ingresos de datos como estudiantes, docentes y materias, matriculas, reportes, etc.), lo cual supondrá un acceso rápido y sencillo a los datos, gracias a interfaces gráficas sencillas y amigables. Además los datos accedidos estarán siempre actualizados, lo cual es un factor muy importante.



## DESCRIPCIÓN DE STAKEHOLDERS (PARTICIPANTES EN EL PROYECTO) Y USUARIOS

Para proveer de una forma efectiva productos y servicios que se ajusten a las necesidades de los usuarios, es necesario identificar e involucrar a todos los participantes en el proyecto como parte del proceso de modelado de requerimientos. También es necesario identificar a los usuarios del sistema y asegurarse de que el conjunto de participantes en el proyecto los representa adecuadamente. Esta sección muestra un perfil de los participantes y de los usuarios involucrados en el proyecto, así como los problemas más importantes que éstos perciben para enfocar la solución propuesta hacia ellos. No describe sus requisitos específicos ya que éstos se capturan mediante otro artefacto. En lugar de esto proporciona la justificación de por qué estos requisitos son necesarios.

### RESUMEN DE STAKEHOLDERS

Nombre	Descripción	Responsabilidades
Jefe del Proyecto	Representante Global	El stakeholder realiza: Representa a todos los usuarios posibles del sistema. Seguimiento del desarrollo del proyecto. Aprueba requisitos y funcionalidades

Resumen de los Stakeholders del Proyecto.



## RESUMEN DE USUARIOS

Nombre	Descripción	Responsabilidades
Administrador	Responsable de la gestión de usuarios como es: roles y permisos.	Seguridad
Secretaria	Responsable del correcto ingreso de datos como de estudiante, docentes, materias y matriculación.	Integridad de Datos
Docente	Encargado de ingreso de notas de los estudiantes	Ingreso notas

Resumen de los Usuarios del Proyecto

## ENTORNO DE USUARIOS

Los usuarios entrarán al sistema mediante un navegador web, identificándose de acuerdo al tipo de usuario con su login y password, tras este paso accederán a la aplicación diseñada de acuerdo al los requerimientos de cada usuario. Los reportes serán generados en pdf como en Microsoft Excel, lo cual le resultará familiar.

## PERFIL DE LOS STAKEHOLDERS

### REPRESENTANTE DEL ÁREA TÉCNICA Y SISTEMAS

#### DE INFORMACIÓN

<b>Representante</b>	Jefe del proyecto
<b>Descripción</b>	Representante Global
<b>Tipo</b>	Experto de Sistemas.
<b>Responsabilidades</b>	Encargado de mostrar las necesidades de cada usuario del sistema. Además, lleva a cabo un seguimiento del desarrollo del proyecto y aprobación de los requisitos y funcionalidades del sistema
<b>Criterio de Éxito</b>	A definir por el cliente
<b>Grado de participación</b>	Revisión de requerimientos, estructura del sistema

Representante del Área Técnica del Proyecto



## PERFILES DE USUARIOS

### ADMINISTRADOR

<b>Representante</b>	Seguridad
<b>Descripción</b>	Administrador
<b>Tipo</b>	Experto de Sistemas.
<b>Responsabilidades</b>	Responsable de la gestión de usuarios como es: roles y permisos.
<b>Criterio de Éxito</b>	A definir por el cliente
<b>Grado de participación</b>	A definir por el cliente

**Usuario Administrador del Proyecto**

### SECRETARIA

<b>Representante</b>	Integridad de Datos
<b>Descripción</b>	Secretaria de cada Escuela
<b>Tipo</b>	Usuario experto
<b>Responsabilidades</b>	Responsable del correcto ingreso de datos como de estudiante, docentes, materias y matriculación
<b>Criterio de Éxito</b>	A definir por el cliente
<b>Grado de participación</b>	A definir por el cliente

**Usuario Secretaria del Proyecto**

### DOCENTE

<b>Representante</b>	Ingreso de Notas
<b>Descripción</b>	Docente
<b>Tipo</b>	Usuario Experto
<b>Responsabilidades</b>	Encargado de ingreso de notas de los estudiantes
<b>Criterio de Éxito</b>	A definir por el cliente
<b>Grado de participación</b>	A definir por el cliente

**Usuario Docente del Proyecto**



## DESCRIPCIÓN GLOBAL DE PRODUCTO

### PERSPECTIVA DEL PRODUCTO

El producto a desarrollar es “Implementación de una Base de Datos Distribuida en el Sistema Académico de la FICA”, con la intención de agilizar su funcionamiento. Las áreas a tratar por el sistema son: Ingreso de estudiantes, docentes, materias y la gestión de matriculación.

### RESUMEN DE LAS CARACTERÍSTICAS

A continuación se mostrará un listado de los beneficios que obtendrá el cliente a partir del producto.

<b>Beneficio del cliente</b>	<b>Características que lo apoyan</b>
Mayor agilidad y rapidez para ver datos y la posibilidad de hacerlo vía servicios web.	Aplicación web.
Mayor control de información.	Base de datos distribuida ya que desde cualquier sitio pueden realizar sus consultas como si fuera en el sitio mismo.
Seguridad.	El ingreso del sistema se controla por medio de un usuario y contraseña, se controla el acceso a las opciones a través de permisos y roles.

Resumen de las características del Proyecto

## DESCRIPCIÓN DE PRODUCTO

### SEGURIDAD

El administrador del sistema será encargado del manejo del la Seguridad.





## **SECRETARIA ESCUELA**

La Secretaria de cada Escuela tendrá acceso a todo el modulo de matriculación, como también de ingreso de estudiantes, docentes y materias.

## **DOCENTE**

El Docente se encarga del ingreso de notas de los estudiantes por materia que el imparte en cada escuela.

## **RESTRICCIONES**

## **OTROS REQUISITOS DEL PRODUCTO**

## **ESTÁNDARES APLICABLES**

## **REQUISITOS DE SISTEMA**

### **SERVIDOR**

Windows XP.

Espacio libre en Disco Duro de 5 GB o mayor.

Procesador Pentium IV o mayor.

Memoria RAM 1G o mayor.

Mouse / Teclado

Monitor

## **REQUISITOS DE DESEMPEÑO**

El mayor requisito de desempeño es la facilidad y rapidez para el acceso de datos debido a que sus datos se encuentran distribuidos evitando los cuellos de botellas. Y el sistema es agradable a la vista del usuario.



## **REQUISITOS DE ENTORNO**

El hardware debe estar conectado de manera adecuada. Se debe mantener siempre los dispositivos de red y servidores web para el perfecto ingreso al sistema.

## **REQUISITOS DE DOCUMENTACIÓN**

### **MANUAL DE USUARIO**

El manual de usuario se encuentra en los anexos el cual contendrá información como: instalación del sistema, modo de acceder a cada rol.

### **GUÍAS DE INSTALACIÓN, CONFIGURACIÓN Y**

### **FICHERO LÉAME**

Las guías de instalación, configuración y el fichero léame se encuentra en los anexos del proyecto



## **CASOS DE USO**

A continuación se presentan los diagramas de casos definidos. Cabe destacar que los casos de uso que no se incluyeron en la fase de construcción sólo figuran en estado de propuestos.

- **USUARIO**

En este caso de uso permite al usuario poder ingresar al sistema según el rol y los privilegios asignados por el administrador. Este puede ser: Usuario del Sistema, Docente o Estudiante.

## **FLUJO DE EVENTOS**

### **FLUJO BÁSICOS**

#### **LOGIN**

1. El sistema solicita al usuario su cuenta este puede ser:  
Usuario del Sistema, Docente o Estudiante.
2. El usuario ingresa a su cuenta con su respectivo nombre de usuario, ingresando también su contraseña mediante la interfaz del sistema (“Autenticación de Usuarios”).
3. Una vez ingresado su login y contraseña debe seleccionar la opción “*Entrar*” para poder ingresar al sistema.



## **FLUJOS ALTERNATIVOS**

### **ERROR DE CUENTA O CONTRASEÑA DE USUARIO**

Según el punto dos. Si el usuario ingresa de manera errónea su cuenta o clave el sistema se encarga de mostrar un mensaje de error, que dice lo siguiente: **“Usuario no Valido”**

### **PRECONDICIONES**

El usuario debe tener previamente creado su cuenta. En caso de no tenerla deberá hablar con el administrador para que le proporcione una.

### **POSTCONDICIONES**

El sistema estará conectando mediante la sesión y el perfil que el usuario ingrese, una vez ingresado podrá acceder a todos los privilegios asignados para su cuenta.

- **USUARIO DEL SISTEMA (SECRETARIA ESCUELA)**

La Secretaria inicia el caso de uso. Una vez ingresado al sistema este le muestra las diferentes opciones que ella puede administrar. Por ejemplo, Estudiante, Docente, Ingreso notas, Matricula, Periodo Académico, Crear Paralelo etc. Ella podrá crear, editar y eliminar según sea el criterio o las necesidades. Para su entendimiento vamos a explicar cómo crear un Estudiante



## FLUJO DE EVENTOS

### FLUJO BÁSICO

El sistema le permite manejar diferentes operaciones como ya lo mencionamos. Puede buscar a un estudiante mediante: cedula, nombre, apellido, fecha nacimiento, etc. Además puede crear, editar, eliminar y exportar xls.

#### CREAR ESTUDIANTE.

1. El caso de uso comienza cuando la secretaria en el sistema académico hace clic en Estudiante.
2. El sistema académico muestra todos los datos que debe ingresar para crear un nuevo Estudiante.
3. Una vez ingresado todos los datos debe elegir la opción de “*Guardar*” para guardar al nuevo Estudiante.
4. El sistema se encarga de guardar al estudiante.
5. Una vez guardado el estudiante se debe elegir la opción “*Regresar*” para regresar al menú donde partimos inicialmente.

#### EDITAR ESTUDIANTE

1. La secretaria define la búsqueda ya sea mediante cedula, nombre, apellido etc.
2. Una vez localizado el estudiante que se desea editar, procedemos a editar los campos que se crea conveniente.



3. Una vez actualizados los datos debe elegir la opción de “*Guardar*” para guardar los datos modificados del Estudiante.
4. El sistema se encarga de guardar los datos editados del estudiante.
6. Una vez guardado el estudiante se debe elegir la opción “*Regresar*” para regresar al menú donde partimos inicialmente.

### **ELIMINAR ESTUDIANTE**

1. La secretaria define la búsqueda ya sea mediante cedula, nombre, apellido etc.
2. Una vez localizado el estudiante que se desea eliminar.
3. Ahora se debe presiona el botón “*Eliminar*”. Y el estudiante se eliminara automáticamente.
4. El sistema elimina al estudiante de la base de datos.

### **NO COLOCA CRITERIO DE BÚSQUEDA**

Si en editar y eliminar estudiante la secretaria no ingresa nada en la opción de búsqueda, el sistema mostrará todos los estudiantes ingresados en el sistema hasta el momento.

### **PRECONDICIONES**

La secretaria debe ingresar con su cuenta y contraseña y perfil para poder acceder a los diferentes módulos.



## **POSTCONDICIONES**

El sistema estará conectando mediante la sesión y el rol que el usuario ingrese, una vez ingresado podrá acceder a todos los privilegios asignados para su cuenta

- **DOCENTE**

El docente inicia el caso de uso ingresando al sistema según el rol y los privilegios asignados por el administrador. Este puede ingresar notas de estudiantes de cada materia que el imparte en las diferentes escuelas.

## **FLUJO DE EVENTOS**

### **FLUJO BÁSICOS**

1. El sistema solicita al usuario su perfil este puede ser:  
Usuario del Sistema, Docente o Estudiante.
2. El Docente debe ingresar su nombre de usuario, su clave y elegir su perfil, en este caso es el de Docente.
3. Debe elegir la escuela a la que pertenece la materia que va ingresar la nota
4. Ahora se debe elegir la materia.
5. Y por ultimo elige Ingreso de Notas, una vez ingresada las notas de los estudiante matriculados para esta materia tiene que guardar.



## **FLUJOS ALTERNATIVOS**

### **ERROR DE CUENTA O CONTRASEÑA DE USUARIO**

Según el punto dos. Si el usuario ingresa de manera errónea su cuenta o clave el sistema se encarga de mostrar un mensaje de error, que dice lo siguiente: **“Usuario no Valido”**

### **PRECONDICIONES**

El usuario debe tener previamente creado su cuenta. En caso de no tenerla deberá hablar con el administrador par que le proporcione una.

### **POS CONDICIONES**

El sistema estará conectando mediante la sesión y el rol que el usuario ingrese, una vez ingresado podrá acceder a todos los privilegios asignados para su cuenta.

- **ESTUDIANTE**

El estudiante inicia el caso de uso ingresando al sistema según el rol y los privilegios asignados por el administrador. Este puede consultar sus notas en el determinado periodo académico.

## **FLUJO DE EVENTOS**

### **FLUJO BÁSICOS**

1. El sistema solicita al usuario su perfil este puede ser:  
Usuario del Sistema, Docente o Estudiante.





2. El Estudiante debe ingresar su nombre de usuario, su clave y elegir su perfil, en este caso es el de Estudiante.
3. Enseguida le aparece el reporte de todas las materias que se encuentra matriculado en dicho periodo académico con sus respectivas notas, el estudiante si desea puede imprimir su reporte, simplemente haciendo click en imprimir se bajara el archivo en pdf, el cual podrá imprimirlo.

## **FLUJOS ALTERNATIVOS**

### **ERROR DE CUENTA O CONTRASEÑA DE USUARIO**

Según el punto dos. Si el usuario ingresa de manera errónea su cuenta o clave el sistema se encarga de mostrar un mensaje de error, que dice lo siguiente: **“Usuario no Valido”**

### **PRECONDICIONES**

El usuario debe tener previamente creado su cuenta. En caso de no tenerla deberá hablar con el administrador para que le proporcione una.

### **POS CONDICIONES**

El sistema estará conectando mediante la sesión y el rol que el usuario ingrese, una vez ingresado podrá acceder a todos los privilegios asignados para su cuenta.



## REQUERIMIENTOS

### STAKEHOLDERS

Los representantes de los usuarios y portavoces de las necesidades son los stakeholders. En este proyecto solamente se ha tratado con un stakeholder como representante de los usuarios y necesidades, sin embargo se han dividido representativamente.

La matriz de atributos de los stakeholders es la siguiente:

Requerimientos	Representante	Ubicación
STK 1 Administrador	Responsable de la gestión de usuarios como es: roles y permisos.	Documento Visión
STK 2 Secretaria	Responsable del correcto ingreso de datos como de estudiante, docentes, materias y matriculación.	Documento Visión
STK 3 Docente	Encargado de ingreso de notas de los estudiantes	Documento Visión

Matriz de atributos de los Stakeholders del Proyecto

### ACTORES

Se define este requerimiento para listar los usuarios potenciales del sistema, en este proyecto se han definido los siguientes actores: *Administrador, Usuario del Sistema (Secretaria Escuela), Docente y Estudiante.*

Requerimientos	Ubicación	Modulo
ACT1: Administrador	Documento Visión	Seguridad
ACT2: Secretaria Escuela	Documento Visión	Ingreso de Datos y Matriculas
ACT3: Docente	Documento Visión	Ingreso de Notas
ACT4: Estudiante	Documento Visión	Reportes de Notas

Matriz de atributos de los Actores del Proyecto



## CARACTERÍSTICAS DE SOFTWARE

Las características de software son las necesidades de los usuarios propuestas por los stakeholders, son los requisitos que debe cumplir el sistema para satisfacer las necesidades.

Las características definidas son las que aparecen en la matriz de atributos, siendo las indicadas como subcaracterísticas las derivadas según una clasificación jerárquica.

Requerimientos	Asignado a
CSW 1: <b>Seguridad</b> Seguridad	
CSW1.1: Manejo de Usuarios Manejo de Usuarios	Equipo completo de análisis, desarrollo e implementación
CSW1.2: Manejo de Roles Manejo de Roles	Equipo completo de análisis, desarrollo e implementación
CSW1.3: Manejo de Permisos Manejo de Permisos	Equipo completo de análisis, desarrollo e implementación
CSW1.4: Manejo de Niveles Manejo de Niveles	Equipo completo de análisis, desarrollo e implementación
CSW1.5: Manejo de Facultades Manejo de Facultades	Equipo completo de análisis, desarrollo e implementación
CSW1.6: Manejo de Escuelas Manejo de Escuelas	Equipo completo de análisis, desarrollo e implementación
CSW1.7: Manejo de Títulos Manejo de Títulos	Equipo completo de análisis, desarrollo e implementación
CSW 2: <b>Usuario del Sistema</b> Secretaria Escuela	
CSW 2.1: Gestión de Matricula Secretaria	Equipo completo de análisis, desarrollo e implementación
CSW 2.1.1: Manejo de Estudiante Manejo de Estudiante	Equipo completo de análisis, desarrollo e implementación
CSW 2.1.2: Manejo de Materia Manejo de Materia	Equipo completo de análisis, desarrollo e implementación
CSW 2.1.3: Manejo de Docente Manejo de Docente	Equipo completo de análisis, desarrollo e implementación
CSW 2.2: Manejo de Periodo Académico Manejo de Periodo Académico	Equipo completo de análisis, desarrollo e implementación
CSW 2.3: Manejo Crear Paralelo Manejo Crear Paralelo	Equipo completo de análisis, desarrollo e implementación
CSW 2.4: Manejo de Horarios Manejo de Horarios	Equipo completo de análisis, desarrollo e implementación
CSW 3: <b>Docente</b> Docente	
CSW 3.1: Manejo de Ingreso de Notas Manejo de Ingreso de Notas	Equipo completo de análisis, desarrollo e implementación
CSW 4: <b>Manejo De Estudiante</b>	



Manejo De Estudiante	
CSW 4.1: Manejo de Reporte Manejo de Reportes	Equipo completo de análisis, desarrollo e implementación

**Matriz de atributos de las características del Proyecto**

## CASOS DE USO

Derivados de las características de software, son el resultado del análisis de las necesidades de los usuarios. La matriz de atributos es la siguiente:

<b>Requerimientos</b>	<b>Asignado a</b>
ECU1: Gestión de Usuarios	Equipo de desarrollo
ECU2: Buscar Usuarios	Equipo de desarrollo
ECU3: Manejo de Roles	Equipo de desarrollo
ECU4: Buscar Roles	Equipo de desarrollo
ECU5: Manejo de Permisos	Equipo de desarrollo
ECU6: Buscar Permisos	Equipo de desarrollo
ECU7: Manejo de Niveles	Equipo de desarrollo
ECU8: Buscar Niveles	Equipo de desarrollo
ECU9: Manejo de Facultades	Equipo de desarrollo
ECU10: Buscar Facultades	Equipo de desarrollo
ECU11: Manejo de Escuelas	Equipo de desarrollo
ECU12: Buscar Escuelas	Equipo de desarrollo
ECU13: Manejo de Títulos	Equipo de desarrollo
ECU14: Buscar de Títulos	Equipo de desarrollo
ECU15: Gestión de Matricula	Equipo de desarrollo
ECU16: Manejo de Estudiante	Equipo de desarrollo
ECU17: Buscar Estudiante	Equipo de desarrollo
ECU18: Eliminar Estudiante	Equipo de desarrollo
ECU19: Reporte de Estudiante	Equipo de desarrollo
ECU20: Manejo de Materia	Equipo de desarrollo
ECU21: Buscar Materia	Equipo de desarrollo
ECU22: Eliminar Materia	Equipo de desarrollo
ECU23: Reporte Materia	Equipo de desarrollo
ECU24: Manejo de Docente	Equipo de desarrollo
ECU25: Buscar Docente	Equipo de desarrollo
ECU26: Eliminar Materia	Equipo de desarrollo
ECU27: Reporte Materia	Equipo de desarrollo
ECU28: Manejo de Periodo Académico	Equipo de desarrollo
ECU29: Buscar Periodo Académico	Equipo de desarrollo
ECU30: Eliminar Periodo Académico	Equipo de desarrollo
ECU27: Reporte de Periodo Académico	Equipo de desarrollo
ECU28: Manejo Crear Paralelo	Equipo de desarrollo
ECU29: Manejo de Horarios	Equipo de desarrollo

**Matriz de atributos de los Casos de Uso del Proyecto**



## **CLASES**

Las clases son requerimientos derivados de los casos de uso como necesidad de representación del modelo de datos.

<b>Requerimientos</b>	<b>Ubicación</b>
CLS1: Docentes	Base de Datos
CLS2: Escuelas	Base de Datos
CLS 3: Estudiantes	Base de Datos
CLS 4: Facultades	Base de Datos
CLS 5: horarios	Base de Datos
CLS 6: Materias	Base de Datos
CLS 7: Materias_Matriculas	Base de Datos
CLS 8: Matriculas	Base de Datos
CLS 9: Modulos	Base de Datos
CLS 10: Niveles	Base de Datos
CLS 11: Paralelos_Materias	Base de Datos
CLS 12: Periodo_Academico	Base de Datos
CLS 13: Permisos	Base de Datos
CLS 14: Roles	Base de Datos
CLS 15: Títulos	Base de Datos
CLS 16: Usuarios	Base de Datos

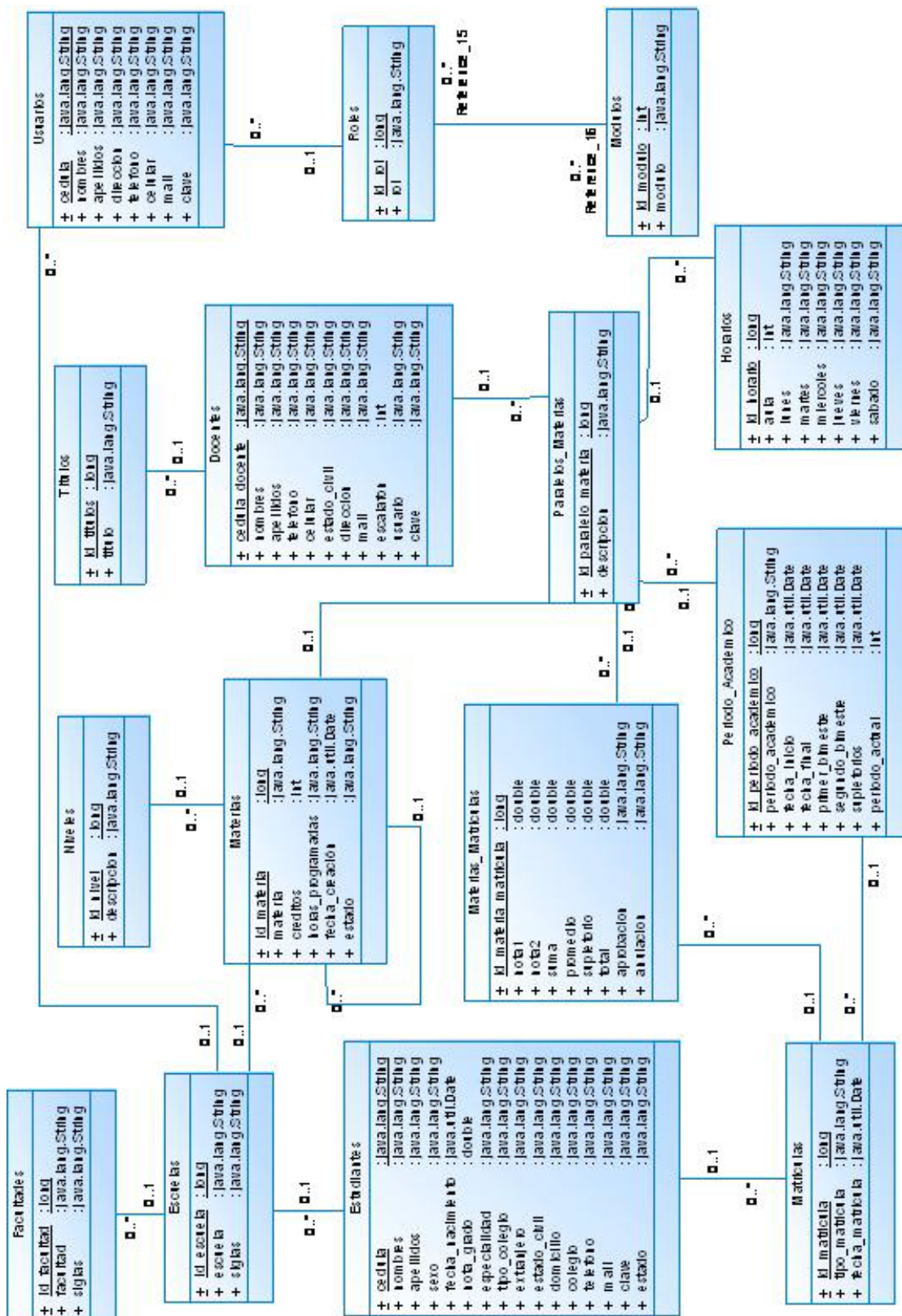
**Matriz de atributos de las Clases del Proyecto**

### **1.5. ANALISIS DE DISEÑO**

A continuación se presentan los modelos definidos en RUP como modelo de datos y modelo de análisis / diseño. Constará de un diagrama de clases, y de un modelo de datos (modelo relacional).



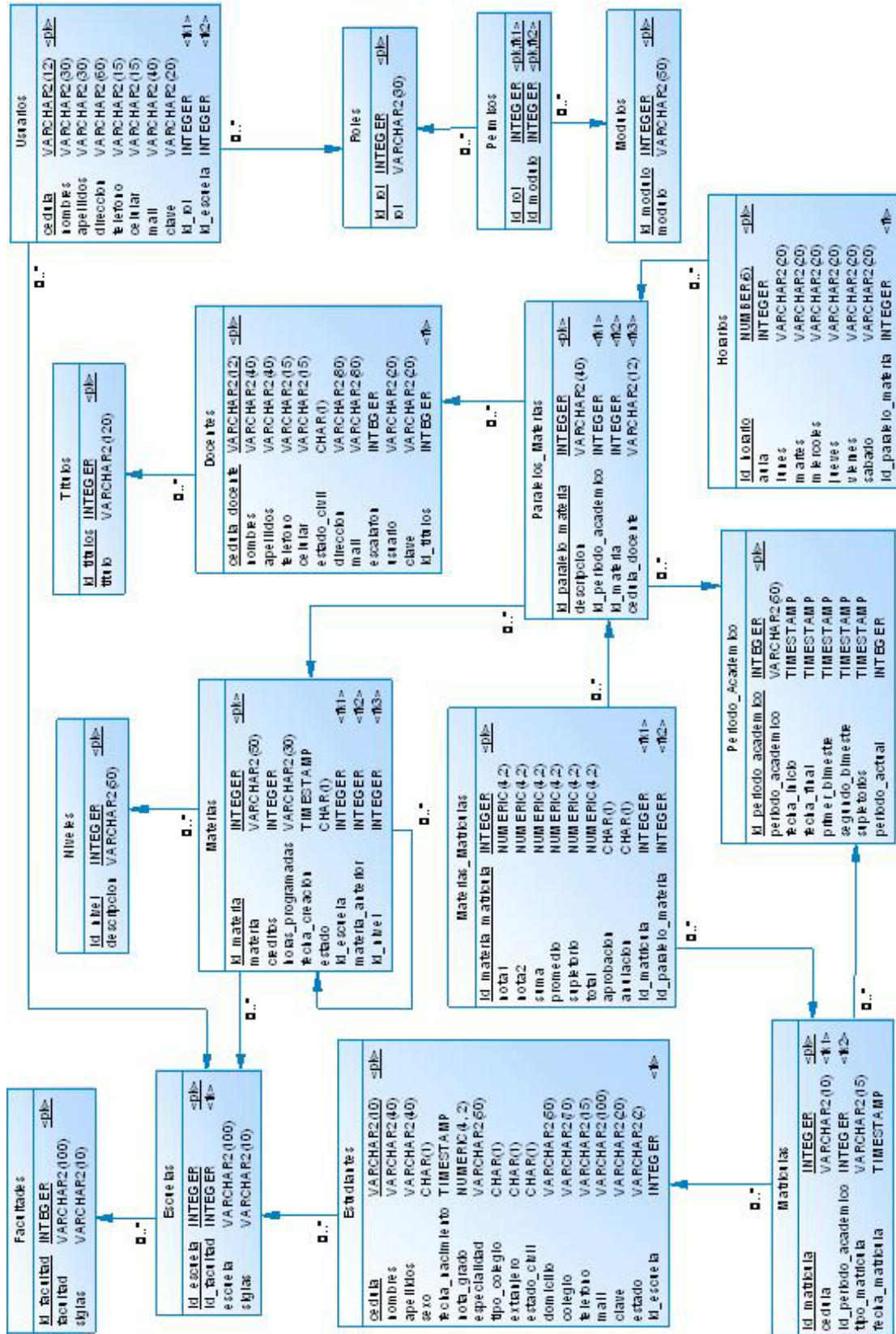
## DIAGRAMA DE CLASES



7. Diagrama de Clase del Sistema Académico



### MODELO ENTIDAD – RELACIÓN



8. Diagrama Relacional del Sistema Académico



## 1.6. IMPLEMENTACIÓN

A continuación se presentan los modelos definidos en RUP como prototipos de interfaces gráficas de usuario diseñados para la aplicación final.

### INTERFACES DE USUARIOS

#### INICIO



#### MENÚ PRINCIPAL







### Listado de Estudiantes

http://127.0.0.1:8080/...tate=6xk75bm6d\_4

## Sistema Académico

Administración ▾ Generales ▾ Académico ▾ Sesión de Administrador null

**Estudiantes**

Ver ▾

Cedula	Nombres	Apellidos	FechaNacimiento	Domicilio	Telefono	Mail	Escuela	Facultad
0400697496	TIRSO ATILO	ERAZO RUEDA	14/05/1988	AZAYA	2602564	tilo_erazo@yahoo.co	Ingeniería Mecatronica	Facultad de Ingenieri
0400775235	LUIS EDMUNDO	BRACHO GONZALEZ	12/03/1990	SAN FRANCISCO	2967143	lucho_bracho@yahoo	Ingeniería Diseño Tex	Facultad de Ingenieri
0500217773	CESAR ANTONIO	SANTO MENA	19/06/1990	GUAYAQUIL DE ALPA	2608726	cesr_nacho19@yahoo	Ingeniería Diseño Tex	Facultad de Ingenieri
0500799168	YOLANDA CONCEPCI	SALAS ECHEVERRIA	19/09/1990	AV. EL RETORNO	2609456	concep_salas17@yah	Ingeniería Diseño Tex	Facultad de Ingenieri
0702388075	FATIMA MARIBEL	TORRES ROMAN	10/04/1988	LA VICTORIA	2604132	globmari_78@yahoo	Ingeniería Diseño Tex	Facultad de Ingenieri
1704589264	CLARA EMPERATRIZ	ESPINOZA PEREZ	18/09/1990	CARANQUI	2605398	clara_20_perez@yah	Ingeniería Diseño Tex	Facultad de Ingenieri
1800985077	LUIS EDGAR	CHILUIZA CARRILLO	28/11/1990	LOS CEIBOS	2609376	luchito_27_ed@yah	Ingeniería Diseño Tex	Facultad de Ingenieri
1001692365	LUIS RAMIRO	LIMA MOROCHO	30/08/1990	OTAVALO	2601562	lu_ramiro@yahoo.co	Ingeniería Diseño Tex	Facultad de Ingenieri
0400912036	JHANET EREDMITA	OVIEDO PANTOJA	25/06/1990	ATUNTAQUI	2608726	ered_ovie_990@yah	Ingeniería Diseño Tex	Facultad de Ingenieri
1001284023	WASHINGTON GUSTA	PROAÑO PAEZ	18/09/1990	LA VICTORIA	2956569	gus_paez@yahoo.co	Ingeniería Diseño Tex	Facultad de Ingenieri
1001298635	AMANDA GERMANIA	CAPELO ALBUJA	22/07/1990	ATUNTAQUI	2608574	amand_gema187@y	Ingeniería Diseño Tex	Facultad de Ingenieri
1002877270	SILVIA MAGALY	CHANDI IMBAQUING	25/01/1990	AV. 17 DE JULIO	2955674	silvi_maga35@yahoo	Ingeniería Diseño Tex	Facultad de Ingenieri

### CREACIÓN Y EDICIÓN

http://127.0.0.1:8080/...tate=6xk75bm6d\_4

## Sistema Académico

Administración ▾ Generales ▾ Académico ▾ Sesión de Administrador null

\* Cedula: 0400697496

Nombres: TIRSO ATILO

Apellidos: ERAZO RUEDA

Sexo:  Masculino  Femenino

FechaNacimiento: 14/05/1988

NotaGrado: 17

Especialidad: QUIMICO BIOLOGO

TipoColegio: Particular

Extranjero:  Si  No

EstadoCivil: Soltero

Domicilio: AZAYA

Colegio: SAN FRANCISCO

Telefono: 2602564

Mail: tilo\_erazo@yahoo.com

Clave: 0400697496

Estado:  Activo  Deshabilitado

IdFacultadBind: Facultad de Ingeniería en Ciencias Aplicadas

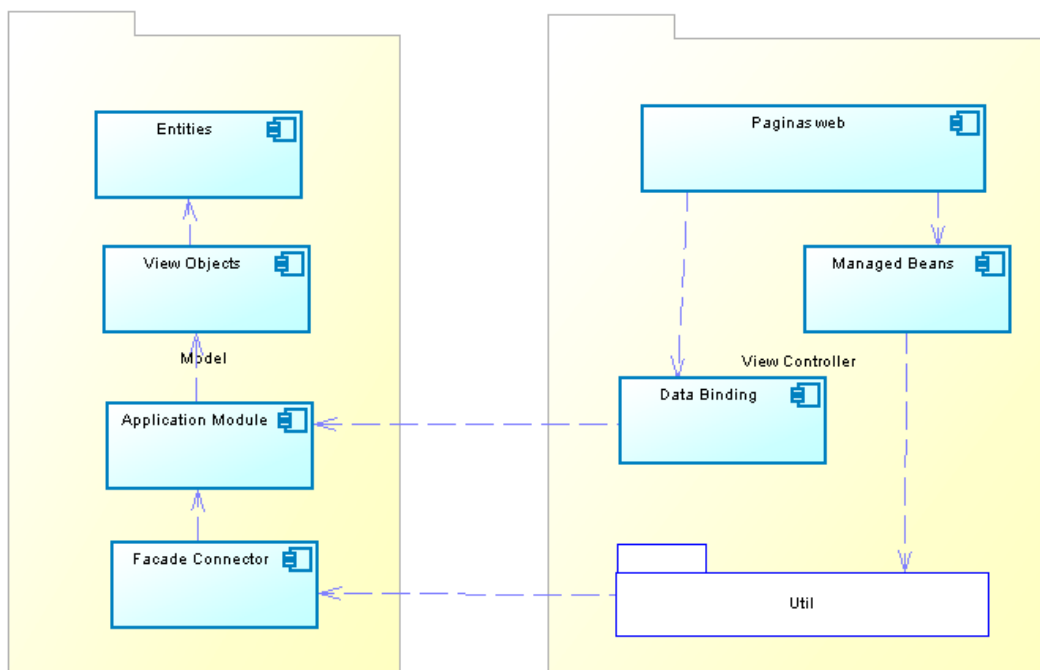
IdEscuela: Ingeniería Mecatronica

Escuela: Ingeniería Mecatronica

Facultad: Facultad de Ingeniería en Ciencias Aplicadas



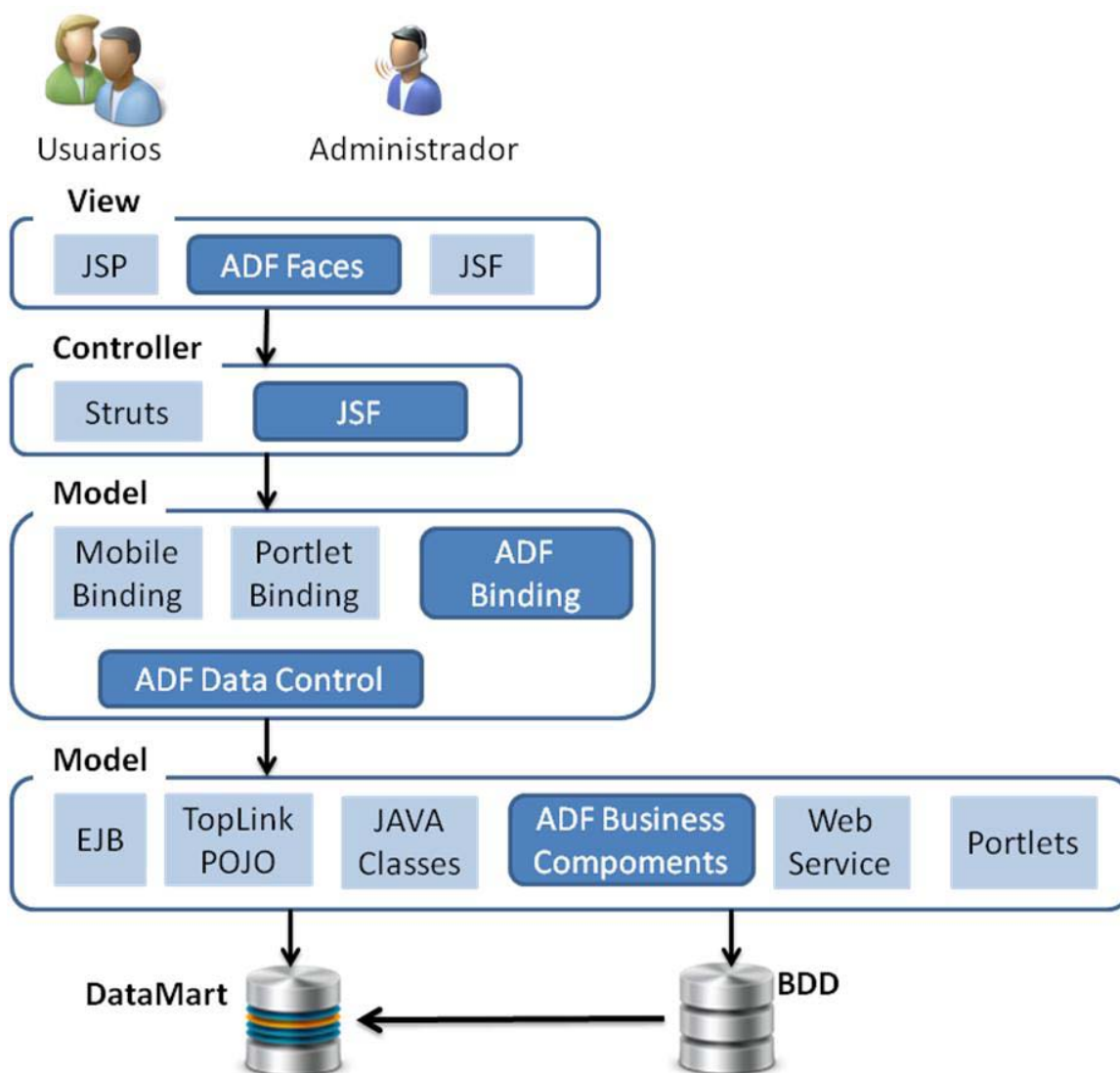
## DIAGRAMA DE COMPONENTES



9. Diagrama de Componentes del Proyecto



## DIAGRAMA DE DESPLIEGUE



10. Diagrama de Despliegue del Proyecto

### 1.7. PRUEBAS

A continuación se muestran las especificaciones de casos de prueba funcionales de los casos de uso incluidos en el proyecto de desarrollo de software.



## ESPECIFICACIÓN DEL CASO DE PRUEBA: “LOGIN - LOGOUT”

### INTRODUCCIÓN

En la presente sección se detallan las pruebas a realizarse para los distintos escenarios del caso de uso “*Login – Logout*”.

### PROPÓSITO

Probar que el caso de uso “*Login – Logout*” está correctamente implementado y que se cumplen las especificaciones funcionales y no funcionales.

### ALCANCE

Sólo se prueban los escenarios mencionados en el documento de Especificación de Caso de Uso: “*Login – Logout*”.

### DEFINICIONES, ACRÓNIMOS Y ABREVIACIONES

Para la presente sección tener en cuenta los siguientes términos:

**Autenticar:** Se refiere al hecho de haber ingresado sus datos para que la aplicación pueda identificar que la persona que intenta acceder a su contenido es quien dice ser. Esta acción se lo denomina técnicamente Login.

**Sesión:** Período de tiempo de actividad que un usuario pasa en el sistema, desde que hace el Login hasta que hace Logout o sale del sistema.

### ESCENARIOS DE PRUEBA

El presente documento contiene los distintos escenarios que se detallaron en la especificación del caso de uso “*Login – Logout*” y cada escenario tiene una breve descripción de lo que trata, el flujo



de actividades que contiene los pasos a realizarse para cumplir el escenario y los puntos de control que indican los pasos donde evaluar exhaustivamente.

### **ESCENARIO: FLUJO BÁSICO**

Validar que el usuario pueda iniciar y terminar su sesión de manera correcta.

**Descripción:** Es el escenario ideal del caso de uso, no deberían presentarse errores.

### **PRECONDICIONES**

El usuario no debe tener su sesión activa en el sistema. Se deben haber creado el usuario y sus datos indicando como dato de entrada.

### **DATOS DE ENTRADA**

Se accederá al sistema con el usuario “100” cuya contraseña es “*admin*”. Su perfil es Administrador.

### **ESCENARIO: ERROR DE CUENTA DE USUARIO**

Verificar el reconocimiento de las cuentas de usuario validando su existencia en la base de datos.

**Descripción:** Comprobar que una cuenta se encuentra o no registrada en la base de datos.

### **PRECONDICIONES**

La cuenta de usuario no ha sido registrada en la base de datos.

### **DATOS DE ENTRADA**

Un usuario cualquiera “*xxx*” con clave “*xxx*”.



## **FLUJO DE ACTIVIDADES**

### **Instrucción**

El usuario ingresa su cuenta de usuario.

### **Resultado esperado**

El sistema muestra mensaje de error “*Usuario no valido*” y retorna a la interfaz que lo solicito.

### **PUNTO DE CONTROLADOR**

Al final de dicha acción muestra el mensaje correspondiente.

### **PUNTOS DE REVISIÓN**

#### **Paso Punto de Control**

Muestra: mensaje de error.

#### **Validación a realizar**

Verificar que los datos sean los correctos, comprobar que el sistema muestre el mensaje de acuerdo al error



## 1.8. DOCUMENTACIÓN

### ORACLE ADF

ADF (Application Development Framework) es un entorno de trabajo, un framework de aplicaciones que se basa en la plataforma Java EE (Enterprise Edition).

Simplifica y acelera la implementación de aplicaciones orientadas a servicios y el desarrollo de aplicaciones empresariales que buscan visualizar, crear, modificar y validar datos.

Reduce la cantidad de código ya que ADF proporciona implementaciones de patrones de diseño hasta la implementación, con los datos de arrastrar y soltar vinculante, el diseño visual de interfaz de usuario.

Las aplicaciones desarrolladas con Oracle ADF pueden instalarse en cualquier servidor de aplicaciones Java EE compatible.

La única herramienta que trabaja con ADF es JDeveloper 10g. y 11g.

### ARQUITECTURA

#### ADF implementa el patrón MVC

Oracle ADF está basado en el patrón de diseño Modelo/Vista/Controlador (MVC). Este modelo separa las aplicaciones en tres capas por cuestiones de rendimiento,



productividad del desarrollador o escalabilidad entre otras, delegando la responsabilidad de funciones de la siguiente manera:

**1) Capa Modelo:** Implementa los objetos, encapsulando la lógica de negocio y es la responsable de la persistencia y la interacción con la base de datos, guardando el estado de la aplicación, está compuesta por dos tipos de componentes

- **ADF Data Controls:** Implementa una abstracción de los servicios de negocio para el cliente. Definen los datos devueltos por el servicio de negocio, estos son: atributos, colecciones y operaciones. Estos se crean arrastrando el servicio de negocio directamente al panel “data control palette” de JDeveloper. Los data control no muestran ningún detalle de la implementación, permitiendo el intercambio de implementaciones de manera rápida y limpia.
- **ADF Binding:** Define cómo trabajan los clientes con los data control. Los bindings son creados de manera automática cuando arrastramos un atributo del data control a la página.

**2) Capa Vista:** Implementa la interfaz de usuario. Pide datos al modelo a través del controlador. Envía eventos al controlador.

Pasa el control al Controlador para que decida la siguiente vista a mostrar (“enrutado”).

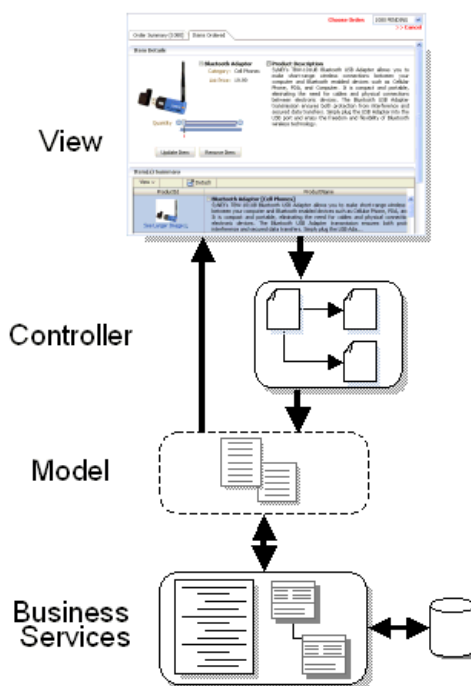




**3) Capa Control:** Es la responsable de controlar el flujo de navegación y de redireccionar las peticiones del usuario. Obviamente en una aplicación J2EE que siga el patrón MVC la comunicación entre capas es esencial. La capa control en este aspecto juega un papel fundamental puesto que es la encargada de recibir las peticiones de la interfaz de usuario y llamar al modelo para ejecutar reglas de negocio e interactuar con la base de datos.

[www.02]<sup>1</sup>

### ARQUITECTURA MVC



11. Arquitectura MVC separa limpiamente la interfaz de usuario, lógica de negocio y Navegación de página [IMAG.34]<sup>2</sup>

<sup>1</sup> <http://oracleadffaces.blogspot.com/2006/11/un-poco-de-teora-el-patrn-mvc-model.html>

<sup>2</sup> [http://download.oracle.com/docs/cd/E15051\\_01/web.1111/b31974/intro.htm](http://download.oracle.com/docs/cd/E15051_01/web.1111/b31974/intro.htm)



- La capa del modelo representa los valores de datos relacionados con la página actual
- La capa vista contiene las páginas de interfaz de usuario utiliza para ver o modificar los datos
- La capa de controlador de los procesos de entrada del usuario y determina navegación de la página
- La capa de servicios de negocio maneja acceso a datos y encapsula la lógica empresarial

### **ARQUITECTURA ADF**

La Arquitectura de ADF está basada en MVC. No obstante de las tres capas definidas en MVC, incorpora una capa denominada Business Service (“Servicios de Negocio”) que es el resultado de la división de la capa Modelo original.

La capa de Servicio de Negocios se encarga del acceso a las fuentes de datos es decir a la base de datos.

<b>MVC</b>	<b>ADF</b>
Modelo	Modelo
	Servicios de Negocio
Vista	Vista
Controlador	Controlador

**12. División de Capa de Modelo en un Servicio de Negocio [IMAG.35]<sup>3</sup>**

---

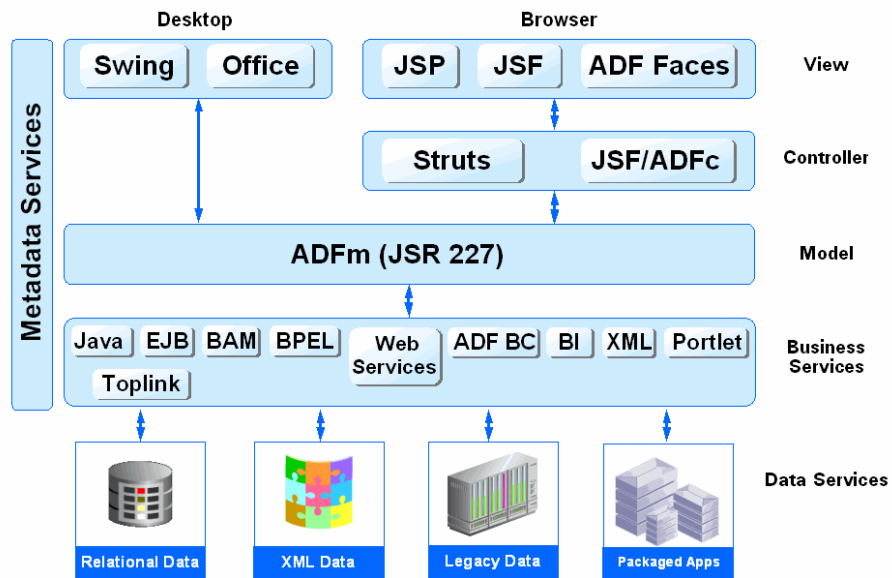
<sup>3</sup> [http://www.cuore.es/otros/seminarios/seminario\\_adf.pdf](http://www.cuore.es/otros/seminarios/seminario_adf.pdf)



La capa de Oracle ADF modelo es la principal, debido a que implementa la especificación JSR-227, esta provee una API para acceder a datos de metadatos. Esta capa permite un enfoque de obligar a cualquier interfaz de usuario, sin la necesidad de escribir código.

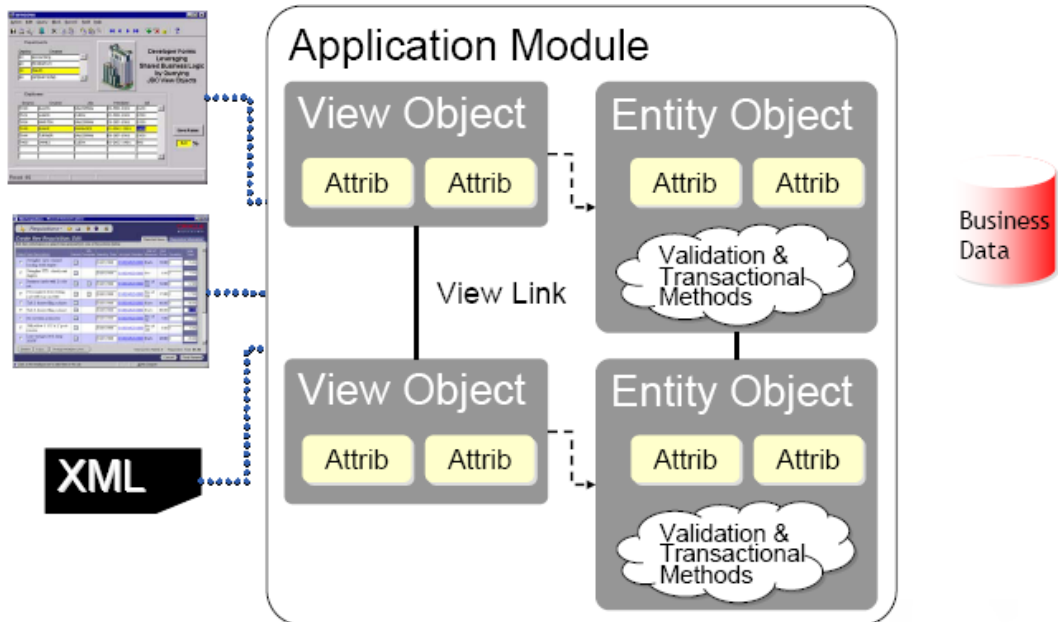
Los otros módulos de tecnología de aplicaciones son:

- Oracle ADF Business Components, que simplifica la creación de servicios de negocios.
- Oracle ADF Faces, que ofrece una amplia biblioteca de componentes AJAX interfaz de usuario para aplicaciones web construidas con JavaServer Faces (JSF).
- Oracle ADF Contralor, que integra JSF con Oracle ADF Modelo. El Contralor ADF extiende su estándar de controlador de JSF, proporcionando una funcionalidad adicional, como los flujos de trabajo reutilizables que pasan a controlar no sólo entre las páginas JSF, sino también entre otras actividades, para las llamadas de método de instancia o de otros flujos de tareas.



13. Arquitectura ADF [IMAG.36]<sup>4</sup>

### ADF BUSINESS COMPONENTS (BC)



14. ADF Business Components [IMAG.37]<sup>5</sup>

<sup>4</sup> [http://download.oracle.com/docs/cd/E15051\\_01/web.1111/b31974/intro.htm](http://download.oracle.com/docs/cd/E15051_01/web.1111/b31974/intro.htm)

<sup>5</sup> [http://www.bbr.cat/presentaciones/PDF/Noticias\\_EventosBbr/BBR+Forms+Java\\_v1.pdf](http://www.bbr.cat/presentaciones/PDF/Noticias_EventosBbr/BBR+Forms+Java_v1.pdf)



Surge la necesidad de tener un framework que abstraiga la complejidad de la plataforma J2EE.

Oracle ADF Business Components es un framework de desarrollo que provee de un conjunto de librerías con funcionalidades que permiten crear componentes de negocios, ayudándonos a mantener la plataforma J2EE, convirtiéndose en un puente entre la base de datos y la aplicación encargándose así de la función de validar los datos.

ADF incrementa la productividad de los desarrolladores ya que implementan componentes reusables.

Sus componentes cubren los siguientes aspectos:

- Persistencia
- Reglas de negocio
- Seguridad
- Validaciones
- Transacciones

ADF BC está compuesta por los siguientes componentes:

- Componentes del dominio del negocio.
- Componentes del modelo de datos.



## COMPONENTES DEL DOMINIO DEL NEGOCIO

Representan características asociados a los objetos de la base de datos, y estos son: Entity Objects y Associations

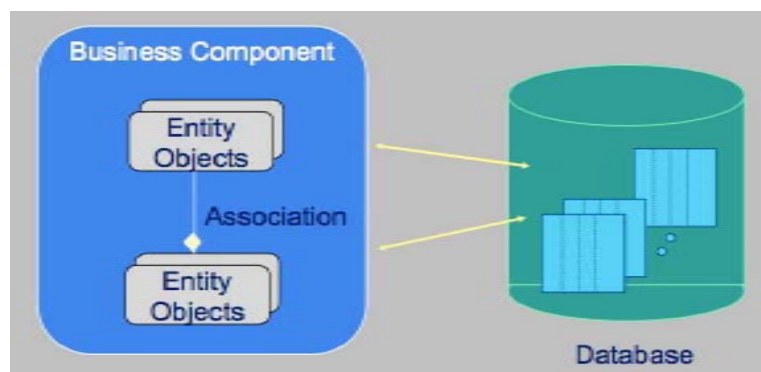
- **Entity Object**

Son los primeros componentes que se debemos crear con Jdeveloper.

Un Entity Object representa un objeto o entidad, cada de ellos mapean una fuente de datos, generalmente una tabla o una vista de una base de datos. Estos contienen atributos, validaciones e información sobre la persistencia. Manejan las interacciones con la base de datos, es decir cada cambio es visto en la entidad y posteriormente se debe hacer un “commit” en la base de datos para que dichos cambios se persistan.

- **Associations**

Está compuesto de una entidad origen (master) y una entidad destino (detail). Son los mecanismo para relacionar dos Entity Objects. Su implementación es mediante XML que es generado por JDeveloper automáticamente.



15. Componentes del dominio del negocio [IMAG.38]<sup>6</sup>

## COMPONENTES DEL MODELO DE DATOS

Recogen los datos de los componentes del dominio del negocio para mostrarlos al resto de las capas (vista y controlador) por medio del modelo, estos son: View Objects y View Links.

- **View Objects**

Generalmente son usados para hacer joins, filtrados y ordenaciones de los datos. Los View Objects representan una vista de los datos, estos pueden construirse directamente o usando cualquier número de Entity Objects mediante sentencias SQL.

---

<sup>6</sup> [http://www.cuore.es/otros/seminarios/seminario\\_adf.pdf](http://www.cuore.es/otros/seminarios/seminario_adf.pdf)



### **PROCESO DE DESARROLLO CON ADF EN JDEVELOPER**

1. Crear el workspace de la aplicación
2. Crear las capas de Servicios de Negocio (Business Services) y Modelo.
3. Crear las capas Controlador y Vista.
4. Testear y depurar.
5. Desplegar en un servidor de aplicaciones.