# DIAGRAMA DE ARQUITECTURA DE LA APLICACIÓN

### **Arquitectura Lógica**

Es el diseño conceptual de nuestra aplicación. Aquí se agruparía la arquitectura de la información. Debemos pensar en las necesidades del usuario y en cómo interactuará con nuestra aplicación.

Arquitectura Física

Se trata de la forma en que se distribuye nuestra aplicación a nuestros usuarios finales en él se denotan los actores y el medio a través del cual se hace llegar la aplicación al computador y/o dispositivo del usuario.

A continuación detallamos la arquitectura presentada para la solución.

Modelo de N-Capas

Este tipo de modelo se basa en la separación de la lógica de negocio de la lógica de diseño.

* Capa de la Vista (Capa de Presentación)

Es la que se encarga de que el sistema interactúe con el usuario y viceversa, muestra el sistema al usuario, le presenta la información y obtiene la información del usuario en un mínimo de proceso.

CARACTERISTICAS

* Amigable
* Fácil de usar
* Entendible

COMUNICACIÓN

* Capa de negocio

PATRON DESARROLLO

* MVC (Modelo Vista Controlador)
* Capa de Negocio (Reglas de reglas de Negocio)

Es donde residen las funciones que se ejecutan, se reciben las peticiones del usuario, se procesa la información y se envían las respuestas tras el proceso.

CARACTERISTICAS

* Implementa todas las reglas del negocio.

COMUNICACIÓN

* Capa de la Vista
* Capa de Datos
* Servicios externos (SOAP, REST)
* Capa de Datos (Capa de Base de Datos)

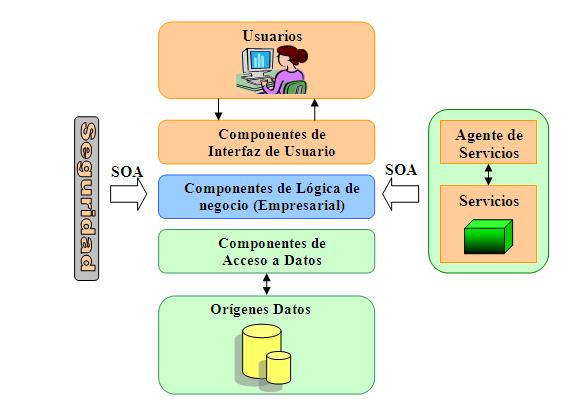
Es donde residen los datos y es la encargada de acceder a los mismos.

CARACTERISTICAS

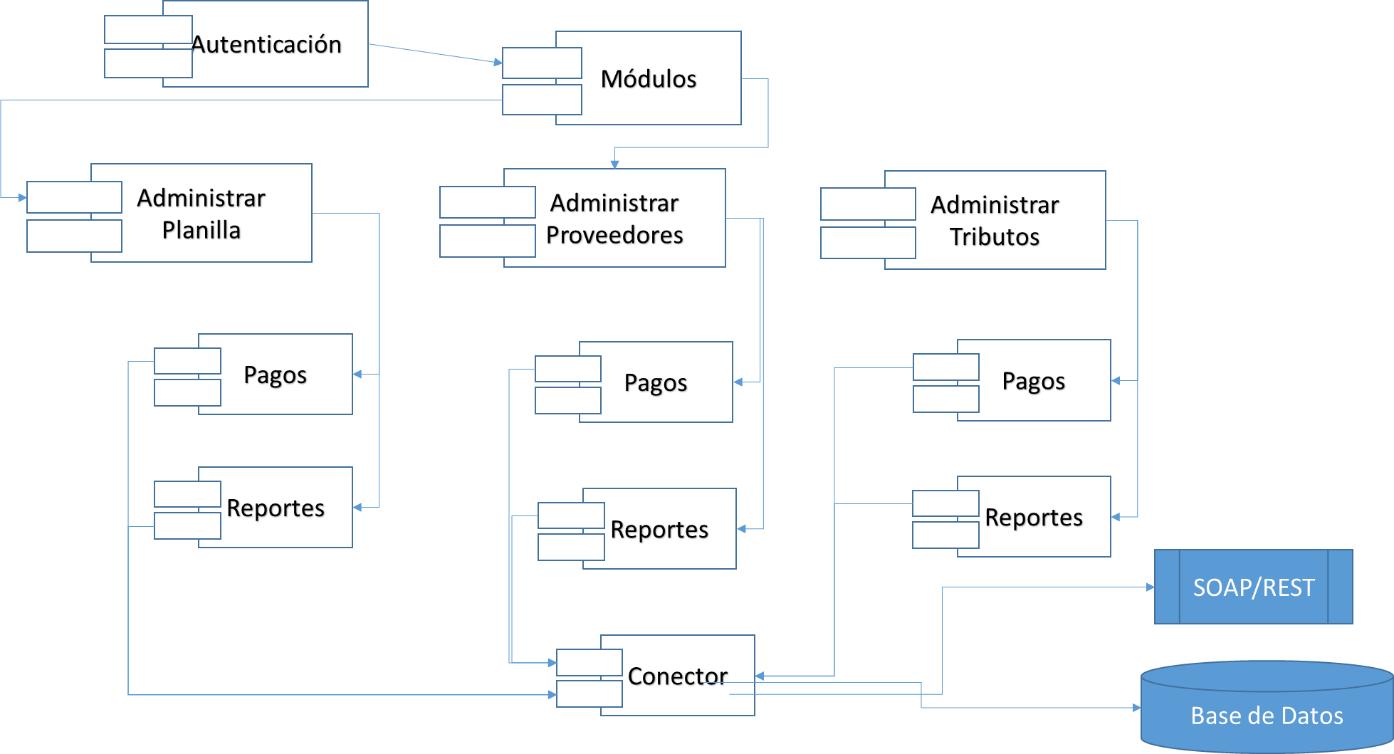
* Acceso a distintos SGBD

COMUNICACIÓN

* Capa de Negocio



# COMPONENTES LOGICOS DE LA APLICACIÓN



# tecnologias

Herramientas de Gestión

Garantizara el cumplimiento de las metas propuestas en cada entregable (Sptrint) programado para su liberación.

Complementara el registro y control de todos los incidentes reportados por el equipo de QA.

Permite identificar mediante indicadores la capacidad/rendimiento de los recursos y de esta manera balancear el equipo como practica de mejora continua.

* Jira / TFS Online

Repositorio de fuentes

Su función principal es la de centralizar el código fuente de la aplicación, así como la de gestionar su versiona miento, brancheo, etc.

* Git / TFS

Construcción de la Solución

Sofware base necesaria para iniciar el desarrollo de la aplicación, se debe cumplir con estos requisitos.

Jenkins, programador de despliegues automáticos para los ambientes de desarrollo, certificación y producción, se requiere el uso de esta tecnología como practica de integración continua.

* Visual Studio cummunity 2015
* Framework 4.5
* MVC 4
* Jenkins

Capa Vista

La capa presentación debe garantizar la usabilidad, fácil de uso, amigable, para ello utilizamos tecnología del lado del cliente que no facilita un buen diseño del front-end.

* Bottstrap 4.5 (Completo)
* JQuery 2.1.1 (Completo)
* Motor de vistas Razor

Capa de Negocio

En ocasiones las aplicaciones interactúan con recursos (pdf, archivos office) para ellos incluimos como tecnología los siguiente componentes, que son de libre uso.

* Itextsharp
* SDK OpenXML 2.0

Capa de Acceso Datos

Para el acceso a datos se recomienda el uso de Entity Framework/ Hibernate for .Net, estas ORM aliviaran el esfuerzo de trabajar directamente en la BD, haciendo el trabajo por nosotros.

* Entity Framework 6 (Code-Firts) / Hibernate
* ADO .NET (Sql Server)
* ODP Net (Oracle)

# DIAGRAMA INTEGRACIÓN CONTINUA

El proceso se ejecuta de forma periódica, y debe ser de la siguiente forma:

* Se descarga del control de versiones las últimas fuentes con los cambios implementados a un directorio de trabajo.
* Se realiza la compilación de estas fuentes.
* Se lanza el análisis estático y de cobertura de las fuentes.
* Se instala en el entorno (compartido) de Desarrollo.
* Se lanzan las pruebas automáticas.
* Se generan los informes.

Si por algún motivo uno de estos pasos falla se detiene el proceso y se notifica a los responsables.

Ventajas

* Identificar fallas, errores de manera temprana
* Entrega y despliegue continuo.

