**Trabajo Interciclo**

Carlos Castillo,Dario Molina, Edison Morocho & Andres Yascaribay.

Mayo 2019.

Universidad Politécnica Salesiana.

Ingeniería en Sistemas.

Aplicaciones Distribuidas.

**Abstract**

En el siguiente documento analizaremos la plataforma Java Enterprise Edition, las ventajas y desventajas de su uso y la relación que mantiene con la arquitectura dividida en capas.

**Tabla de Contenidos**

[Cap](#wrwpc1vh12do)**[í](#wrwpc1vh12do)**[tulo 1  
Introducci](#wrwpc1vh12do)**[ó](#wrwpc1vh12do)**[n e informaci](#wrwpc1vh12do)**[ó](#wrwpc1vh12do)**[n general](#wrwpc1vh12do) 5

[Introducción:](#j0zll) 5

[Especificaciones](#bookmark) 5

[Ventajas](#erhn88wgw5lu) 5

[Desventajas​​:](#pvocxt2jvo) 6

[Modelo:](#jd1pl5xlk1ks) 7

[Relación De La Arquitectura Jee Con La Arquitectura N-capas](#lhfmv4yampk3) 8

[Relación que existe entre Java Enterprise Edition y la arquitectura n-capas](#ksv4uv) 11

[Lista de referencias](#irxkq9cc4ge7) 12

**Lista de figuras**

[Cap](#wrwpc1vh12do)**[í](#wrwpc1vh12do)**[tulo 1  
Introducci](#wrwpc1vh12do)**[ó](#wrwpc1vh12do)**[n e informaci](#wrwpc1vh12do)**[ó](#wrwpc1vh12do)**[n general](#wrwpc1vh12do) 5

[Introducci](#j0zll)**[ó](#j0zll)**[n:](#j0zll) 5

[Especificaciones](#bookmark1) 5

[Ventajas](#erhn88wgw5lu) 5

[Desventajas](#pvocxt2jvo)**[​​](#pvocxt2jvo)**[:](#pvocxt2jvo) 6

[Modelo:](#jd1pl5xlk1ks) 7

[Relacio](#lhfmv4yampk3)**[́](#lhfmv4yampk3)**[n De La Arquitectura Jee Con La Arquitectura N-capas](#lhfmv4yampk3) 8

[Relacio](#ksv4uv)**[́](#ksv4uv)**[n que existe entre Java Enterprise Edition y la arquitectura n-capas](#ksv4uv) 11

[Lista de referencias](#irxkq9cc4ge7) 12

Capítulo 1  
Introducción e información general

Introducción:

Las universidades públicas en el Ecuador disponen de 125.000 cupos para bachilleres de los cuales 3.000 son para institutos privados, estos datos son obtenidos directamente del proceso de postulación realizado en el portal digital www.serbachiller.ec. Para este período se establecieron tres etapas, con dos fases de asignación, en las que los jóvenes deberán decidir si aceptan o no el cupo esto según datos obtenidos del Diario “El Telégrafo”. En base a estos datos y con la aspiración de granjear nuevos estudiantes de entre los aspirantes, se plantea crear una aplicación con información referente a la Universidad Politécnica Salesiana que ayude al aspirante a decidirse por estudiar en esta Universidad.

Objetivos

* Diseñar una aplicación informativa e interactiva referente a la Universidad Politécnica Salesiana sede Cuenca.
* Establecer los requerimientos de los Stakeholder.
* Diseñar el entornos gráfico de la aplicación.
* Diseñar la estructura lógica de la aplicación.
* Realizar las pruebas de la aplicación

Antecedentes

En los últimos años el ingreso de estudiante a la Universidad Politécnica Salesiana sede Cuenca ha visto un declive en sus matrículas comparado con el número de alumnos matriculados en las sedes de Guayaquil y Quito, en el período académico 2017-2017 de los 3.087 estudiantes nuevos en la UPS, la sede Cuenca obtuvo un total de 347 estudiantes, la sede Quito 1.227 estudiantes y la sede de Guayaquil 1.513 estudiantes. Pese a que en el periodo 2017-2018 la afluencia de estudiantes mejoró considerablemente con respecto al periodo anterior, no se pudo superar a la sede de Quito la cual en el periodo 2017-2018 obtuvo un total de 1.690 estudiantes nuevos mientras que la sede Cuenca obtuvo un total de 1.139 estudiantes.

Los principales factores que pueden estar provocando esta situación son la poca publicidad e información suministrada a los nuevos estudiantes, especialmente información referente al ámbito de pago de matrículas puesto que muchos consideran que la universidad es demasiada costosa.

Alcance

En la aplicación se podrán encontrar la información disponible de las diferentes carreras de la Universidad Politécnica Salesiana entre lo atributos de cada carrera se encontrará:

* Su duración en ciclos
* Una descripción de la carrera
* Su campo laboral
* Título
* Modalidad
* Jornada de estudio

No se presentara datos relevantes del director de carrera por motivos de confidencialidad. Además que se dará a conocer como conseguir becas estudiantiles, información relacionada con las formas de pago de las matrículas en las cuales se encontrará:

* Contado
* Cuotas
* Dos cuotas
* Tres cuotas
* Cuatro cuotas
* Cinco cuotas

Información relacionada a los quintiles qué son y cómo estos son parte fundamental para el cálculo de su matrícula.

Es cierto que la Universidad Politécnica Salesiana es una universidad privada pero también, en conjunto con el gobierno, a través de la la institución IFTH, financian becas para facilitar el estudio a aspirantes con altos méritos académicos, de bajos recursos económicos o con capacidades especiales. Este tipo de información se mostrará en la aplicación, logrando con ello atraer a aspirantes de bajos recursos económicos, en este caso, que por desconocimiento del programa de becas y sabiendo que es una institución privada pensó que no podía postular a esta Universidad.

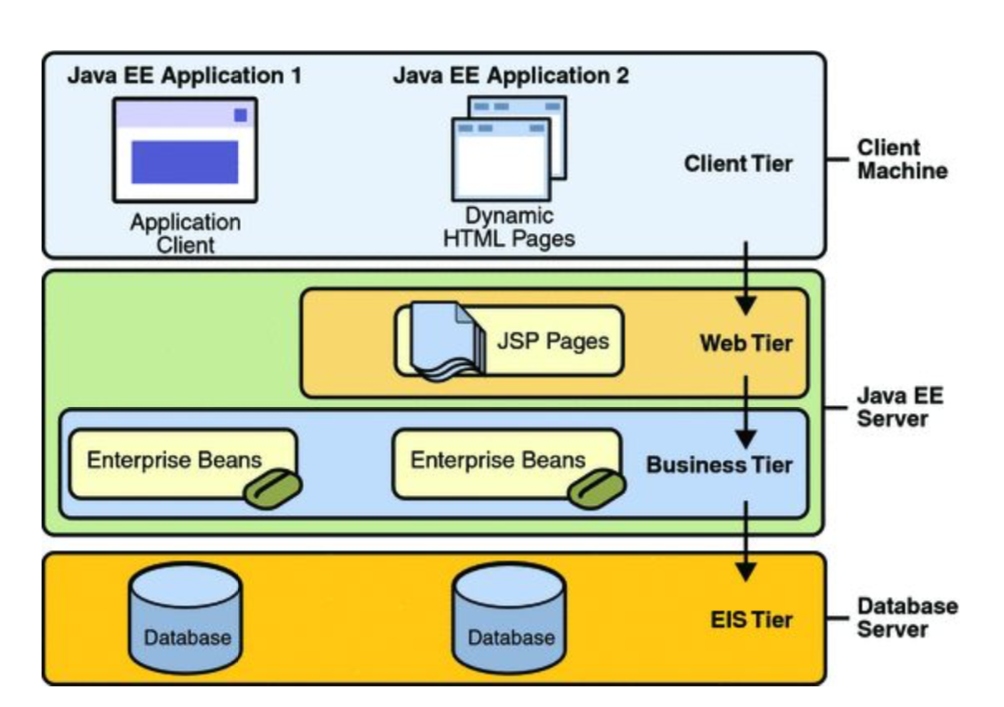
Modelo:

El modelo clásico de capas en la arquitectura de las aplicaciones Java EE se divide en las siguientes:

● Cliente: normalmente se trata de un navegador pero puede ser un teléfono inteligente o una computadora de escritorio, incluso otra aplicación. Es la que presenta la información al usuario.

● Capa web: se comunica con el cliente y la capa de negocio. Obtiene los datos y los transforman al formato adecuado al cliente generalmente HTML o JSON.

● Capa de negocio: proporciona y persiste los datos de la capa cliente y contiene la lógica de negocio de la aplicación. Se ejecuta en un servidor de aplicaciones o contenedor de servlets.Sistemas de información: donde se persisten los datos de la aplicación, puede ser una base de datos relacional como Oracle, MySQL o PostgreSQL o una base de datos NoSQL como Redis o MongoDB u otros sistemas como Elasticsearch.

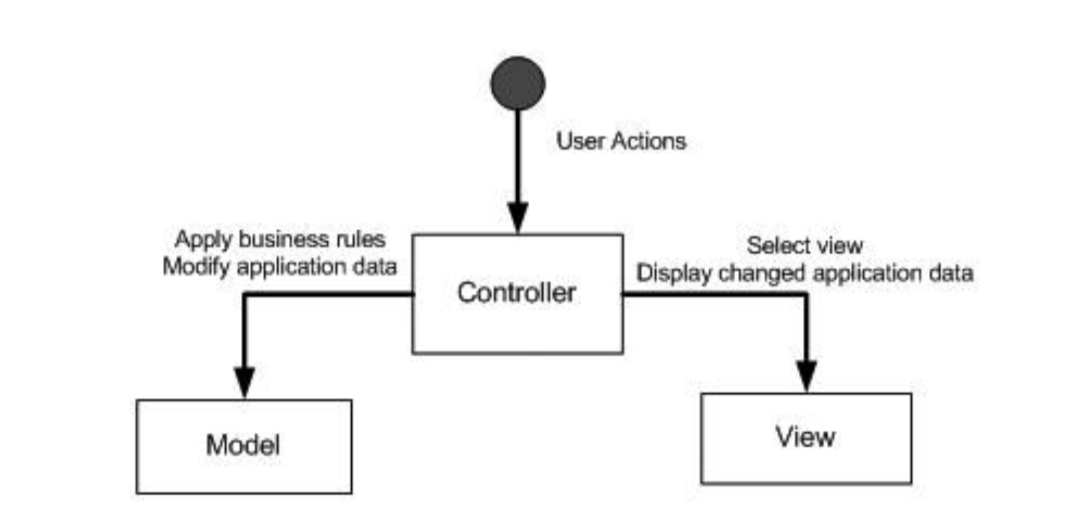


*“Figura 1”.* Java y su distribución en capas.

Relación De La Arquitectura Jee Con La Arquitectura N-capas

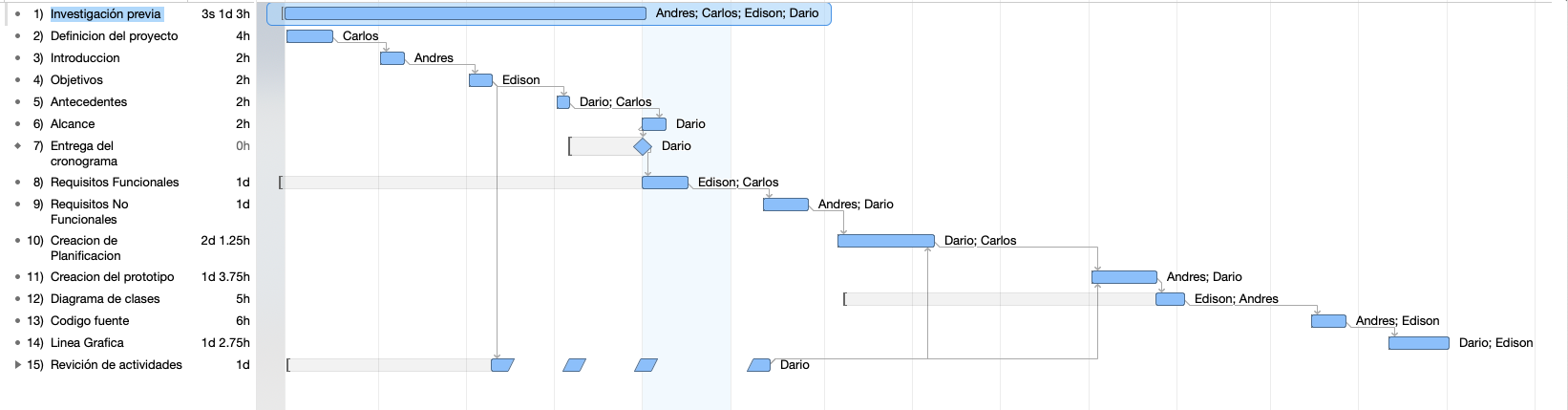
El desarrollo de aplicaciones distribuidas de n niveles es un trabajo complejo y desafiante. La distribución del procesamiento en niveles separados conduce a una mejor utilización de los recursos. También permite la asignación de tareas a expertos que están mejor capacitados para trabajar y desarrollar un nivel particular. Los diseñadores de páginas web, por ejemplo, están más equipados para trabajar con la capa de presentación en el servidor web. Los desarrolladores de bases de datos, por otro lado, pueden concentrarse en desarrollar procedimientos y funciones almacenados. Sin embargo, mantener estos niveles como silos aislados no tiene ningún propósito útil. Deben estar integrados para lograr un objetivo empresarial más grande. Es imperativo que esto se haga aprovechando el protocolo más eficiente; de lo contrario, esto conduce a una grave degradación del rendimiento.  
 Además de la integración, una aplicación distribuida requiere varios servicios. Debe poder crear, participar o administrar transacciones mientras interactúa con sistemas de información dispares. Esto es una necesidad absoluta para garantizar la concurrencia de los datos de la empresa. Dado que se accede a las aplicaciones de nivel n a través de Internet, es imperativo que estén respaldadas por servicios de seguridad sólidos para evitar el acceso malicioso.

Los eventos desencadenados por cualquier acción del usuario son interceptados por el controlador. Dependiendo de la acción, el controlador invoca el modelo para aplicar reglas comerciales adecuadas que modifican los datos de la aplicación. El controlador selecciona un componente de vista para presentar los datos de la aplicación modificados al usuario final. Por lo tanto, verá que MVC proporciona pautas para una separación clara de responsabilidades en una aplicación. Debido a esta separación, varias vistas y controladores pueden trabajar con el mismo.

*“Figura 2”.* Descripción del funcionamiento del controlador cuando el usuario genera una acción.

**Capítulo 2**  
**Cronograma.**

Las diferentes tareas han sido propuestas para que cada integrante pueda tener un control de

****

*“Figura 3”. Arquitectura* JAVA EE.

Relación que existe entre Java Enterprise Edition y la arquitectura n-capas

Según lo que se ha investigado para el presente trabajo se ha visto una clara relación que existe entre Java EE y la arquitectura N-capas ya que Java EE para permitir que se tenga las ventajas antes mencionadas, divide en 4 capas sus componentes, estas capas son cliente, web , negocio y datos las cuales ya se explicaron con anterioridad en este documento, la arquitectura vista en clase de N-capas divide a la aplicación así mismo en componentes pero a diferencia de Java EE esta divida en 3 componentes usuario, negocios y datos. Pero como podemos observar cada una de estas capas tienen la misma finalidad tanto en Java EE como en la arquitectura N-capas.

Tabla 1

*Relacion N-Capas y JEE*

|  |  |
| --- | --- |
| **N-Capas** | **Java EE** |
| **Cliente** | **Usuario** |
| **Web** | **Negocio** |
| **Negocio** |
| **Datos** | **Datos** |

Podemos observar cada una de estas capas tienen la misma finalidad tanto en Java EE como en la arquitectura N-capas

Lista de referencias

Méndez,J (2016) Java™ EE at a Glance. California, E.E.U.U.

[Viklund](http://andreasviklund.com/), A. (2006). Java y su distribución en capas. [Figura 1]. Recuperado de <http://www.jtech.ua.es/j2ee/2006-2007/jee.html>

Kayal,D.(2009). Descripción del funcionamiento del controlador cuando el usuario genera una acción. [Figura 2]. Recuperado de <https://www.developer.com/design/article.php/10925_3808106_4/Introducing-Enterprise-Java-Application-Architecture-and-Design.htm>

Barrios.J (2003).Arquitectura JAVA EE.. [Figura 3].Recuperado de <https://users.dcc.uchile.cl/~jbarrios/J2EE/node14.html>