|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| SEGUIMIENTO DECISIONES DE DISEÑO DE SOFTWARE | | | | | | |
|  |  |  |  |  |  |  |
| **Problema/ Amenaza** | Mantener disponibles todos los recursos de la plataforma los 365 días del año. | | | | | |
|  |  |  |  |  |  |  |
| **Orientación de la decisión** | Para que el sistema cuente con un alto nivel de disponibilidad a lo largo del año, se debe contar con una arquitectura modular que permita el mantenimiento en zonas específicas del sistema, esto para no afectar otras áreas de este y por ende, su disponibilidad. | | | | | |
|  |  |  |  |  |  |  |
| **Supuestos** | * No se cuenta con un presupuesto en el proyecto. * Mantener la disponibilidad del sistema no debe afectar la seguridad ni el desempeño de este. | | | | | |
|  |  |  |  |  |  |  |
| **Restricciones** | * La arquitectura se implementa para una aplicación empresarial. * La arquitectura se implementa usando Java EE. * La implementación de la arquitectura no debe suponer ningún costo. * La seguridad prima por encima de cualquier componente o requerimiento no funcional. | | | | | |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **Alternativas consideradas** | Se considero emplear un patrón MVC para la separación entre la capa de negocio y la capa de presentación, pero JSP no nos permite hacer una separación completa, ya que se debe mezclar código Java con tags html. | | | | | |
|  |  |  |  |  |  |  |
| **Solución encontrada** | La arquitectura implementada será arquitectura multicapa, la cual se dividirá en:   * Capa de presentación. * Capa de Negocio. * Capa de persistencia.   La capa de presentación será la encargada de recibir peticiones de los usuarios y realizará llamados a la capa de negocio.  La capa de negocio se encargará de las acciones relacionadas con las peticiones realizadas por los usuarios de la aplicación.  La capa de persistencia almacena la información de usuarios, puntos de venta, productos y demás componentes de la aplicación. | | | | | |
|  |  |  |  |  |  |  |
| **Decisiones relacionadas** | * Se implementará el patrón Data Access Objects (DAO) para crear un nivel de abstracción en la capa de negocio para acceder a la capa de persistencia. * Para la capa de presentación se utilizará JSP (Java Server Pages). * Para la capa de negocio se utilizará EJB (Enterprise JavaBeans). * Para la capa de persistencia se utilizará JPA (Java Persistence API). | | | | | |
|  |  |  |  |  |  |  |

< control de cambios>

|  |  |  |
| --- | --- | --- |
| **CONTROL DE CAMBIOS** | | |
| **Fecha** | **Descripción** | **Autor(es)** |
| 05/03/2018 | Creación de documentar decisiones de diseño | Juan Felipe García Martínez. |