|  |  |
| --- | --- |
| La arquitectura de software de un sistema es una descripción de alto nivel de la estructura del sistema, los componentes que lo integran y la manera en que estos componentes se comunicaran entre sí para satisfacer los requerimientos y restricciones del sistema. **NIVELES DE ABSTRACCIÓN** Clases, estructuras - Módulos, paquetes, librerías-Servicios (procesos, grupos de procesos) **SOFTWARE** Programas de cómputo que en su ejecución proporcionan las caracteristicas, funciones y desempeño de un producto. Mediante el uso de estructuras de datos permiten que se manipule de manera adecuada la información  **CARACTERÍSTICAS DESEABLES DE UN PRODUCTO DE SOFTWARE** ¡ Mantenibilidad ¡ Confiabilidad ¡ Eficiencia ¡ Usabilidad **Requerimientos**: Descripción formal de lo que necesitamos construir | Hay 3 tipos de restricciones - Técnicas - De negocio - Legales o regulatorias **DESARROLLO BASADO EN COMPONENTES** Define las estructuras de datos, algoritmos, características de la interfaz y mecanismos de comunicación asignados a cada componente del software **INTERFAZ Y EVENTOS** Interfaz de un componente - Determina las operaciones que implementa un componente. Usualmente son los atributos y métodos publicos que el componente define mas los eventos que emite -  Eventos -  Esepecifican la forma que el componente notifica al exterior una respuesta a un estímulo externo o un cambio en alguna condición extern |

|  |
| --- |
| **DISEÑO DE COMPONENTES** Define las estructuras de datos, algoritmos, características de la interfaz y mecanismos de comunicación asignados a cada componente del software ¡ Esta tarea la desarrollan los ingenieros de software o programadores con conocimientos firmes en estructuras de datos y objetos ¡ El diseño a nivel componente permite revisar los detalles del diseño y garantiza su consistencia con otras representaciones del diseño **CAPTURA DE REQUERIMIENTOS** Identificar a todos los actores en nuestro sistema  ¡ Capturar y describir todos los posibles caso y escenarios  ¡ Flujo de usuario, expander cada caso de uso a través de flujo de eventos  ¡ Cada evento contiene  ¡ Acciones  ¡ Datos |