**Arquitetura de Software - Grupo 2**

|  |  |  |
| --- | --- | --- |
| Nome | RA | E-Mail |
| Elaine Pereira da Silva Alves | 1601555 | elainepsalves@gmail.com |
| Janaina Pereira dos Santos | 1700049 | janainasantos.j@gmail.com |
| Luciana Eloi de Andrade | 1201491 | lueloi000@gmail.com |
| Marcelo Augusto de Moraes | 1200638 | marcellometrix@gmail.com |
| Sara Pastor Hidalgo | 1601074 | sa.hidalgo@hotmail.com |
| Sergio Eduardo Di Santoro Bruzetti | 1700475 | sergiodisantoro@gmail.com |

**Exercício 3.4** - Pesquise frameworks (gratuitos e comerciais) que potencializam o uso de cada um dos estilos de arquitetura mencionados abaixo:

Arquitetura baseada em componentes (component-based)

Message Bus

Service-Oriented Architecture (SOA)

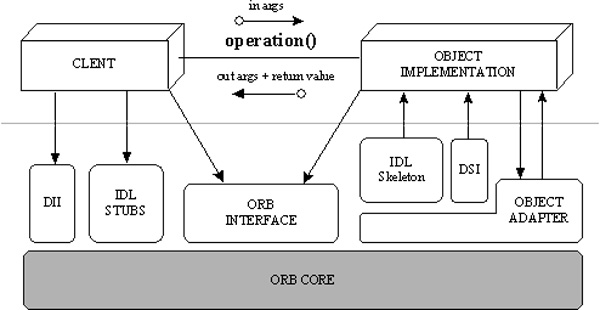
Identifique na descrição do produto as características mencionadas nesta apresentação.

**Arquitetura baseada em componentes:**

**CMM (CORBA Component Model) do OMG (Object Management Group).**

CORBA – Common Object Request Broker Architeture, é uma arquitetura aberta, independente de fornecedores, que aplicativos de computador usam para trabalhar juntos sobre redes. Usando um protocolo padrão IIOP, um programa baseado em CORBA de qualquer fornecedor, em quase todos os computadores, sistemas operacionais, linguagem de programação, e rede, podem interoperar com um programa baseado em CORBA do mesmo ou de um outro fornecedor, em quase qualquer outro computador, sistema operacional, linguagem de programação e rede.

Fonte: <http://www.corba.org/faq.htm>



**DCOM (Distributed Component Obejct)**

DCOM (acrônimo para Distributed Component Object Model) é uma tecnologia proprietária da Microsoft para criação de componentes de software distribuídos em computadores interligados em rede. O DCOM é uma extensão do COM (também da Microsoft) para a comunicação entre objetos em sistemas distribuídos. A tecnologia foi substituída, na plataforma de desenvolvimento .NET, pela API .NET Remoting e empacotada no WCF.

O DCOM pode ser utilizado na construção de aplicações em três camadas, de forma a centralizar as regras de negócio e processos, obter escalabilidade e facilitar a manutenção.

O DCOM funciona de forma transparente tanto para a aplicação cliente quanto para o servidor, que são codificados de acordo com o padrão COM.

Fonte: <https://pt.wikipedia.org/wiki/DCOM>

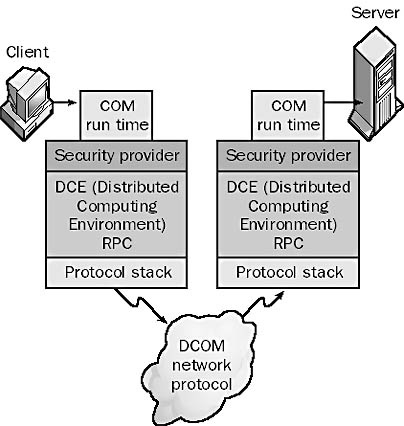
**Component Object Model (COM)** é uma plataforma da Microsoft para componentes de software lançada em 1993. Ela é usada para permitir a comunicação entre processos e a criação dinâmica de objetos em qualquer linguagem de programação que suporte a tecnologia.

**COM+**

Para fornecer aos desenvolvedores suporte a transações distribuídas, e melhor gerenciamento de memória e processamento, assim como para posicionar o Windows como uma alternativa a outros sistemas operacionais corporativos, a Microsoft introduziu a tecnologia Microsoft Transaction Server no Windows NT Service Pack 4.

No Windows 2000, tal extensão significativa à COM foi incorporada ao sistema operacional e renomeada COM+. Na mesma época a DCOM foi reconsiderada como uma entidade separada.

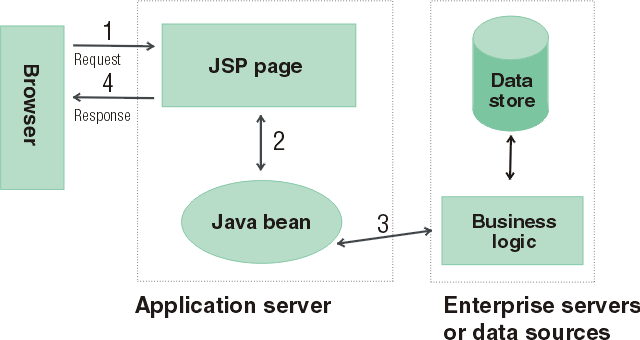
Fonte: <https://pt.wikipedia.org/wiki/Component_Object_Model>



**JavaBeans**

São componentes de software escritos na linguagem de programação Java. Segundo a especificação da Sun Microsystems os JavaBeans são "componentes reutilizáveis de software que podem ser manipulados visualmente com a ajuda de uma ferramenta de desenvolvimento". Um bean também pode ser definido como uma classe Java que expõe propriedades, seguindo uma convenção de nomenclatura simples para os métodos getter e setter. Praticamente são classes escritas de acordo com uma convenção em particular. São usados para encapsular muitos objetos em um único objeto (o bean), assim eles podem ser transmitidos como um único objeto em vez de vários objetos individuais. O JavaBean é um Objeto Java que é serializável, possui um construtor nulo (construtor vazio - construtor noarg) e permite acesso às suas propriedades através de métodos getter e setter.

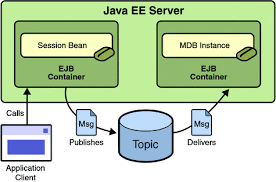
Fonte: <https://pt.wikipedia.org/wiki/JavaBeans>



**Enterprise JavaBeans (EJB)**

É um componente da plataforma JEE que roda em um container de um servidor de aplicação. Seu principal objetivo consiste em fornecer um desenvolvimento rápido e simplificado de aplicações Java, com base em componentes distribuídos, transacionais, seguros e portáveis. Atualmente, na versão 3.2 (liberada em 28/05/2015), o EJB tem seu futuro definido conjuntamente entre grandes empresas como IBM, Oracle e HP, como também por uma vasta comunidade de programadores numa rede mundial de colaboração sob o portal do JCP.

<https://pt.wikipedia.org/wiki/Enterprise_JavaBeans>



**Message Bus**

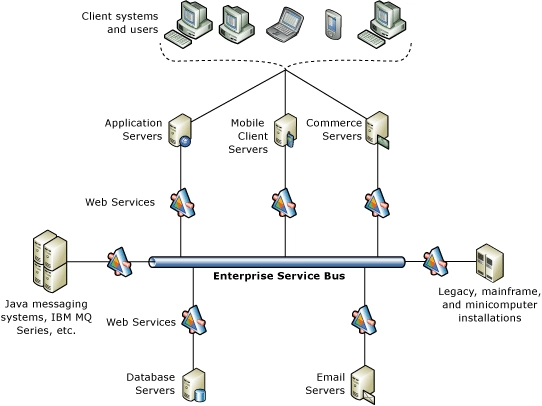
Modelo de arquitetura que prescreve o uso de um software que pode receber e enviar mensagens utilizando um ou mais canais de comunicação, para que as aplicações possam interagir sem precisar conhecer detalhes específicos sobre cada outra.

Fonte: <https://marcobaccaro.wordpress.com/2010/10/21/o-que-um-estilo-de-arquitetura/>

**Enterprise Service Bus (ESB)**

É uma implementação de infraestrutura para habilitar e disponibilizar softwares orientados a serviços (SOA). No entanto, experiências do mundo real de soluções SOA tem demonstrado que ESB é somente um de muitos componentes necessários para construir uma infraestrutura orientada a serviço (SOI). O termo ESB tem gerado algumas direções, no qual sua definição tem várias interpretações sobre plataforma de integração com requerimentos iniciais e particulares de SOA.

Infraestrutura ESB:



Baseado em muitos sucessos de infraestrutura para serviços, fornecedores de tecnologia consideram Enterprise Service Bus com uma coleção de padrões arquiteturais do modelo tradicional Enterprise Service Integration (EAI), com um middleware orientado a mensagem, web services, nteroperabilidade entre plataformas, serviços e repositórios. Na plataforma Microsoft, o produto para ESB ou EAI é o Biztalk Server.

Fonte: <https://marcobaccaro.wordpress.com/2011/02/21/enterprise-service-bus/>

**SOA**

O termo "Service-Oriented Architecture" (SOA) ou Arquitetura Orientada a Serviços expressa um conceito no qual aplicativos ou rotinas são disponibilizadas como serviços em uma rede de computadores (Internet ou Intranets) de forma independente e se comunicando através de padrões abertos. A maior parte das implementações de SOA se utilizam de Web services (SOAP , REST e WSDL). Entretanto, uma implementação de SOA pode se utilizar de qualquer tecnologia padronizada baseada em web.

<https://pt.wikipedia.org/wiki/Service-oriented_architecture>

SOAP (Simple Object Access Protocol, em português Protocolo Simples de Acesso a Objetos) é um protocolo para troca de informações estruturadas em uma plataforma descentralizada e distribuída.

Fonte: <https://pt.wikipedia.org/wiki/SOAP>

A Representational State Transfer (REST), em português Transferência de Estado Representacional, é uma abstração da arquitetura da World Wide Web (Web), um estilo arquitetural que consiste de um conjunto coordenado de restrições arquiteturais aplicadas a componentes, conectores e elementos de dados dentro de um sistema de hipermídia distribuído.

<https://pt.wikipedia.org/wiki/REST>

A Web Services Description Language (WSDL) é uma linguagem baseada em XML utilizada para descrever Web Services funcionando como um contrato do serviço. Trata-se de um documento escrito em XML que além de descrever o serviço, especifica como acessá-lo e quais as operações ou métodos disponíveis.

<https://pt.wikipedia.org/wiki/Web_Services_Description_Language>

