

## Aula 1

### Desenvolvimento Web Back End

Prof. Rafael Veiga de Moraes

1

### Conversa Inicial

2

### Análise de requisitos

- Requisitos funcionais
  - Definem o que a aplicação deve fazer
- Requisitos não funcionais
  - Definem como a aplicação será desenvolvida

3

- Como implementar os requisitos não funcionais?
  - Persistência de dados
  - Interpretação das requisições HTTP
  - Gerenciamento de sessão
  - Gerenciamento de *threads*
  - *Web services*
  - Envio de notificações por e-mail

4

### Arquitetura de Sistemas

5

### Arquitetura de sistemas

- Arquitetura física
- Arquitetura lógica

6

- ▀ **Arquitetura física**
  - **Cliente**
  - **Servidor**
  - **Sistemas de informação corporativos**

7

- ▀ **Arquitetura lógica**
  - **Modelo de uma camada**
  - **Modelo de duas camadas**
  - **Modelo multicamadas**

8

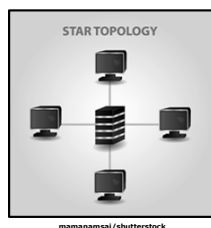
- ▀ **Modelo de uma camada**
  - **Do início dos anos 60 até meados dos anos 80**
    - ✓ **Processamento centralizado**
    - ✓ **Banco de dados rudimentares**
    - ✓ **Mainframes**

9

- ▀ **Modelo de duas camadas**
  - **Conhecido como modelo cliente-servidor**
  - **Predomina ao longo da década de 80**
    - ✓ **Surgimento das interfaces gráficas e dos Sistemas Gerenciadores de Banco de Dados (SGBDs)**
    - ✓ **Popularização dos *personal computers* (PCs)**
    - ✓ **Adoção das redes locais (Local Area Network – LAN)**

10

- ▀ **Modelo de duas camadas**

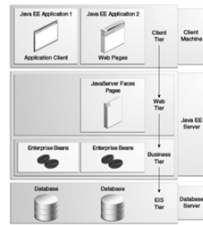


11

- ▀ **Modelo multicamadas**
  - **Camada de apresentação**
  - **Camada de domínio**
  - **Camada de fonte de dados**

12

### Modelo multicamadas



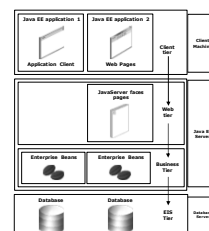
### Plataforma Java EE

- Desenvolvimento de aplicações corporativas de grande porte
- Fornecer um poderoso conjunto de Application Programming Interfaces (APIs)
  - Reduz a complexidade e o tempo de desenvolvimento do projeto
  - Otimizar o desempenho da aplicação
  - Permite desenvolver aplicações distribuídas, escaláveis, portáteis e transacionais

- Arquitetura física
  - Cliente (*client machine*)
  - Servidor (*Java EE server*)
  - Servidor de banco de dados (*database server*)

- Arquitetura lógica
  - Camada de apresentação
    - Camada do cliente (*client tier*)
  - Camada de domínio
    - Camada web (*web tier*)
    - Camada de negócios (*business tier*)
  - Camada de fonte de dados
    - Camada dos sistemas de informação corporativos (*EIS tier*)

### Arquitetura da plataforma Java EE



## Ambiente da Plataforma Java EE

19

- Camada do cliente
  - Máquina do cliente
- Camada da *web*/negócios
  - Servidor Java EE
- Camada EIS
  - Servidor de banco de dados

20

- Máquina do cliente
  - Cliente de aplicativo
  - Applet

21

- Servidor Java EE
  - Servidor de aplicação
    - ✓ Apache Geronimo, GlassFish, JBoss
  - Servlet Container
    - ✓ Tomcat, Jetty

22

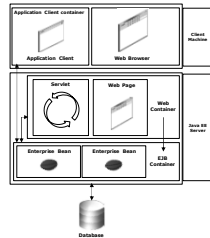
- Sistemas de informação corporativos
  - Servidor de banco de dados
    - ✓ SGBD
  - Servidores externos

23

## Componentes da Plataforma Java EE

24

- **Web Container**
  - JSP, JSF e Servlet
- **EJB Container**
  - EJB



25

- **Componentes web**
  - Servlet
  - JSP (JavaServer Pages)
  - JSF (JavaServer Faces)

26

- **Componentes de negócio**
  - **EJB (Enterprise Java Beans)**
    - ✓ *Session beans*
      - Stateless
      - Stateful
      - Singleton
    - ✓ *Message-driven bean*

27

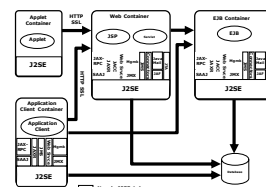
## Serviços e Bibliotecas

28

- JPA (Java Persistence API)
- JTA (Java Transaction API)
- JAX-RS (Java API for RESTful Web Services)
- JMS (Java Message Service)
- JNDI (Java Naming and Directory Interface)
- JAXP (Java API for XML Processing)
- JavaMail

29

- **Visão geral da plataforma Java EE**



30