

## 5.1

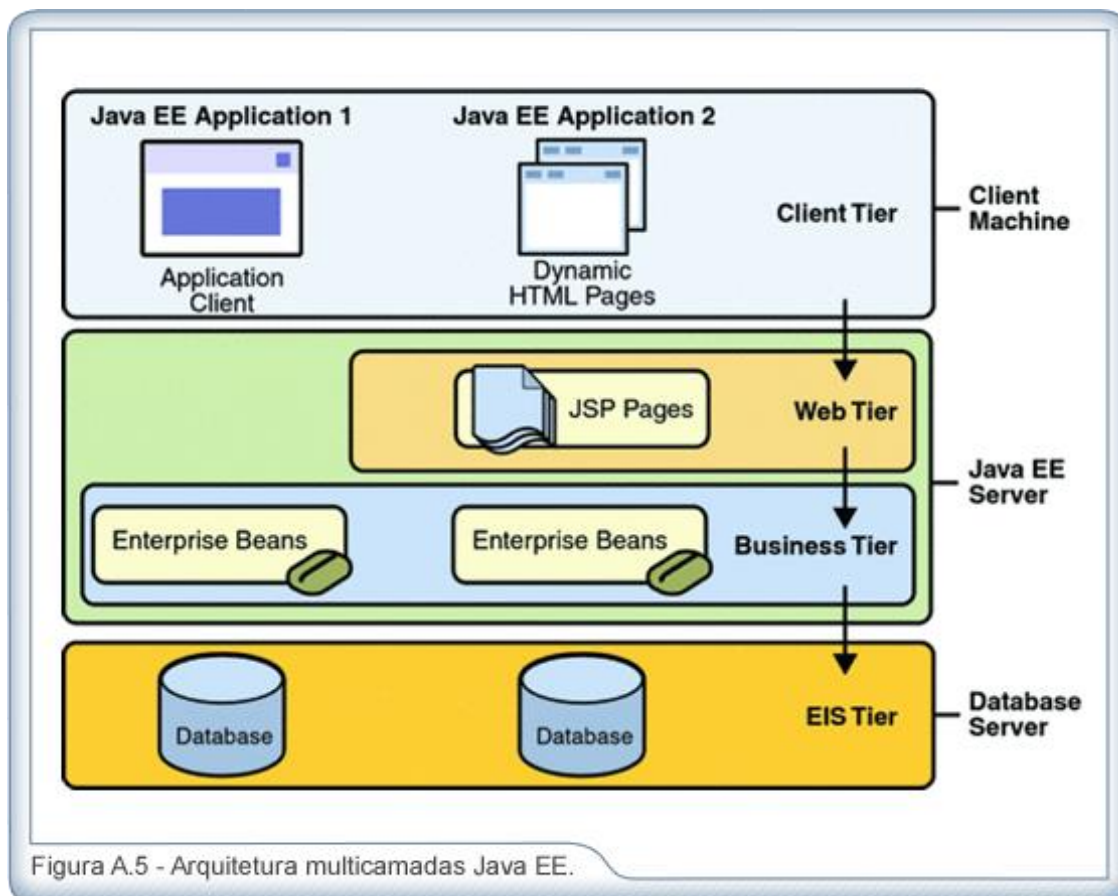
a) O diagrama acima representado é um diagrama de componentes. Neste, a componente `ConversionManagement` é fornecida às componentes `BroadcastEngine` e `BlogViewer`. Por outro lado, requer a componente `BlogDataSource`, que por sua vez, requer a componente `Log4j`.

b) A componente “Apache Log4j” serve para fazer “logging”, isto é, um processo que permite gravar registos, dados, informações e resultados para depois serem analisados.

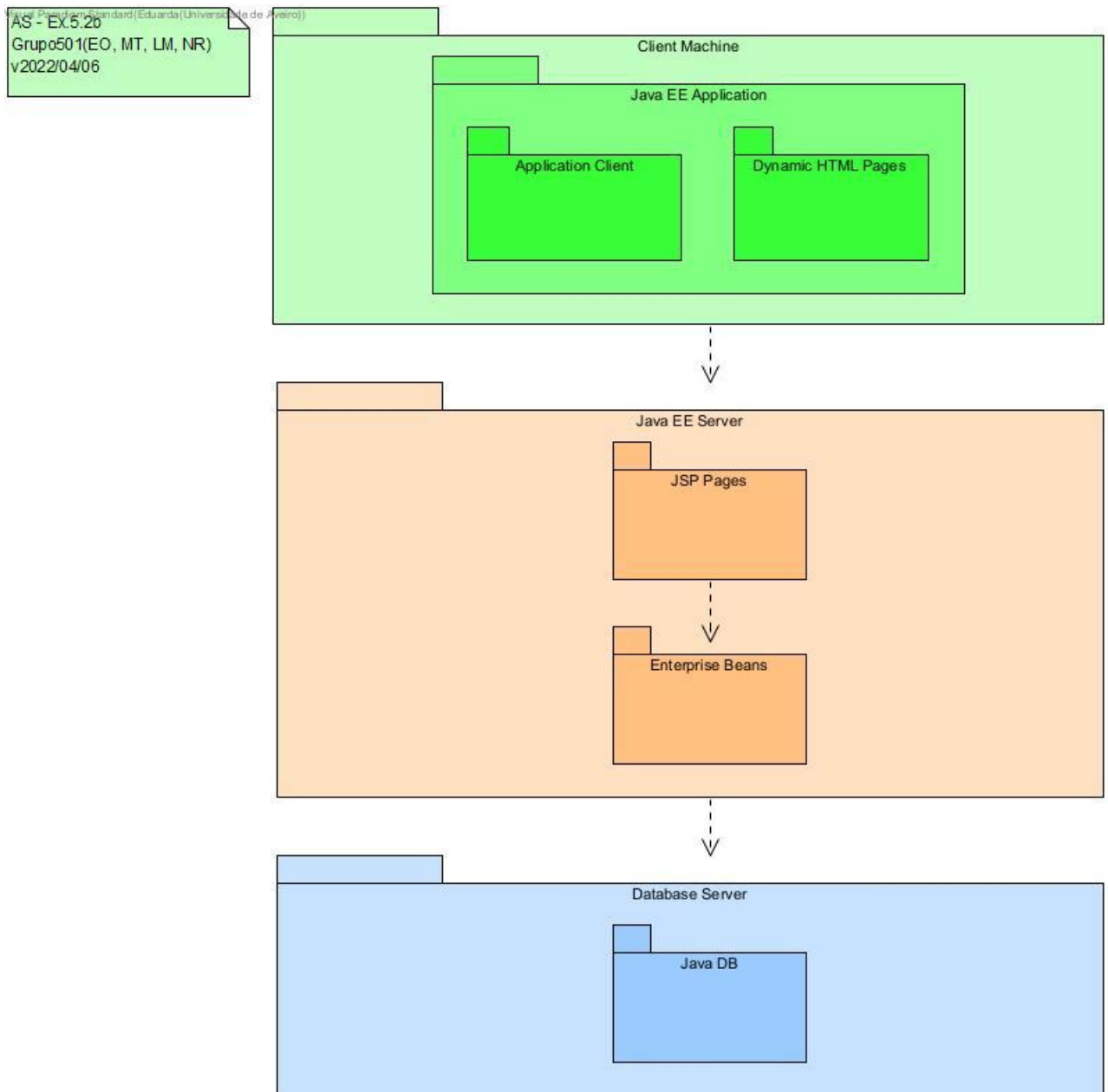
Tanto se pode referir a um comportamento de uma aplicação como todas as submissões e alterações feitas ao longo de um trabalho. Por fim é interessante realçar que é usada em aplicações Java que são muito populares no mundo.

c) Coordinates implementation group: ‘org.apache.logging.log4j’, name: ‘log4j-core’, version: ‘2.17.2’.

## 5.2 a) Exemplo concreto de uma arquitetura multicamada:



**5.2 b)** Transcrição do diagrama anterior para o Visual Paradigm, com as devidas adaptações:



**Explicação da arquitetura:**

Concluimos que o diagrama UML que melhor se adaptava a esta arquitetura seria um diagrama de pacotes. Posto isto:

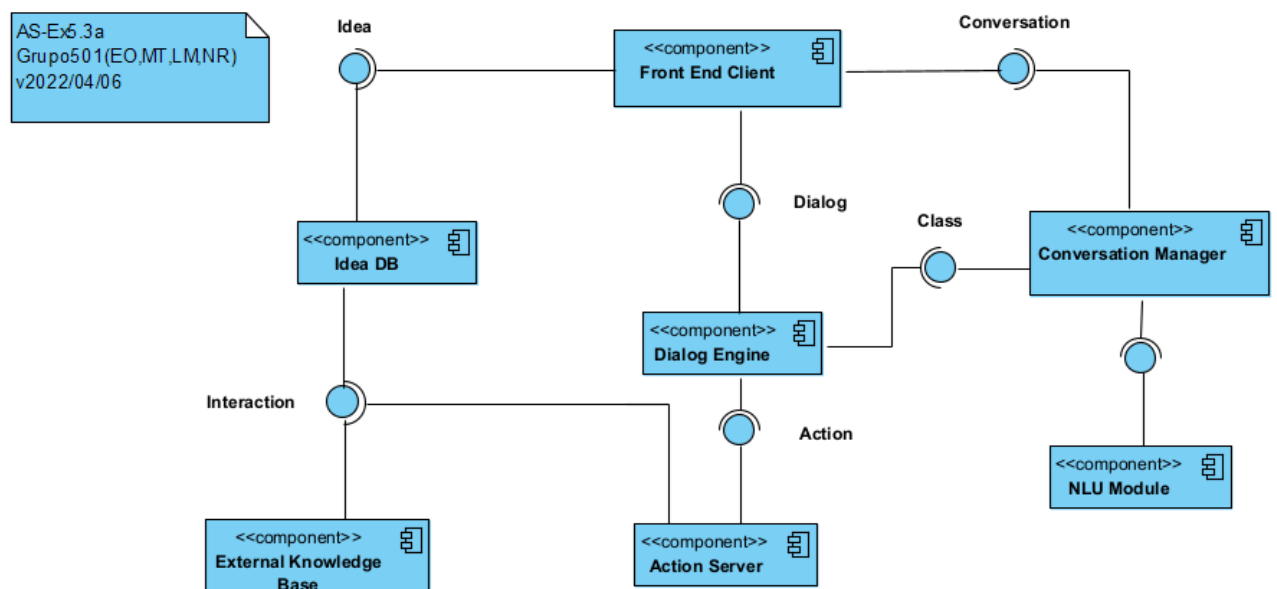
É na *Client Machine* que são executadas as aplicações Java EE, cujas componentes são a *Application Client* e *Dynamic HTML pages*.

A lógica é fornecida à *Client Machine* pelo *Java EE Server*, que engloba as *JSP pages*. Estas, por sua vez, dependem dos *Enterprise Beans*, cujo objetivo consiste em fornecer um desenvolvimento rápido e simplificado de aplicações Java. Tudo isto executado no servidor *Java EE*.

Por fim, chegamos à *Database Server* – um pacote muito importante visto que contem os sistemas de banco de dados Java. Assim, os pacotes anteriores irão depender deste.

### 5.3

a)



b)

