

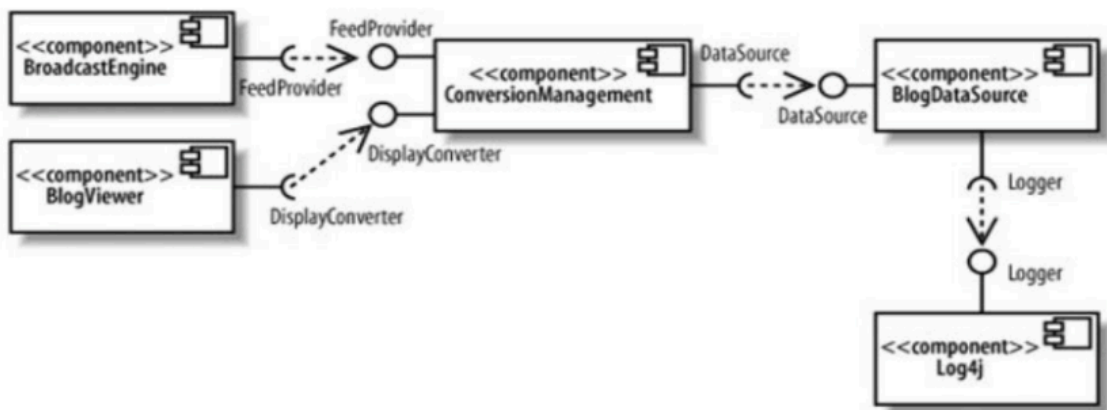
Tiago Almeida(113106), Tiago Costa(114629), Sofia Marrafa(114591), Pedro Costa(112682)

Grupo 3, v2024-03-20

RELATÓRIO

LAB 5: Vistas de arquitetura

Exercício 5.1




- a) No esquema apresentado temos 5 componentes: BroadcastEngine, BlogViewer, ConversionManagement, BlogDataSource.

Os componentes comunicam entre si da seguinte forma:

- O ConversionManagement oferece a interface FeedProvider e a interface DisplayConverter aos componentes BlogViewer e BroadcastEngine, respetivamente, que, por sua vez, as implementam. Para além disso, o componente ConversionManagemtn implementa ainda a interface DataSource, oferecida pelo BlogDataSource.
- O Componente BlogDataSource oferecea interface DataSource ao ConversionManagement, como já foi referido e, além disso, implementa a interface Logger oferecida pelo componente Log4j.

- b) O Apache Log4j é uma biblioteca da linguagem java desenvolvida pela empresa Apache Software Foundation. Esta serve para criar sistemas de *logging*.

Using Log4j in your Gradle build

To build with **Gradle** , add the dependencies listed below to your `build.gradle` file.

`build.gradle`

```
dependencies {  
    implementation 'org.apache.logging.log4j:log4j-api:2.23.1'  
    implementation 'org.apache.logging.log4j:log4j-core:2.23.1'  
}
```

Caso quisessem incluir Log4j em um projeto usando a “build tool” do Gradle, as coordenadas para o identificar são:

- 'org.apache.logging.log4j:log4j-api:2.23.1';
- 'org.apache.logging.log4j:log4j-core:2.23.1';

Exercício 5.2

a)

Para efeitos de exemplo usámos o sistema de compras online *eBay*.

Dentro da camada mais superficial (**UI layer**), encontra-se a interface do website, ou seja, o layout e o design que são apresentados ao cliente (**Frontend**);

Depois, na camada seguinte (**Application layer**), incluímos os serviços de compra, registo/login e procura de itens.

Segue-se a camada do *Domain* (**Domain layer**), onde estão os serviços de Pagamentos, o acesso e o Inventário;

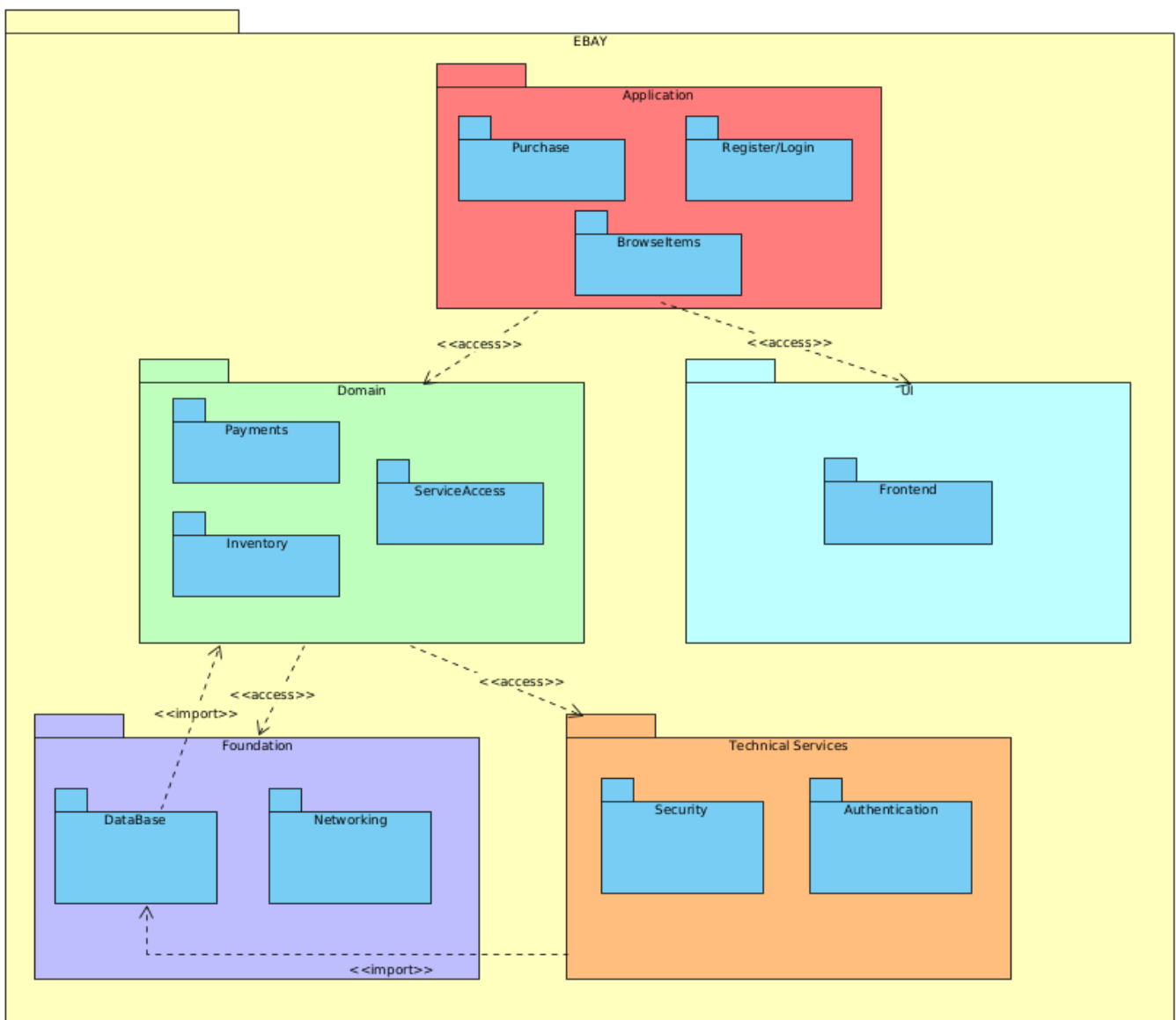
Temos ainda a camada *Technical Services* (**Technical Services layer**) onde se encontram os serviços de alto nível, como por exemplo os componentes para autenticação e segurança da aplicação;

Por fim, temos a camada mais profunda (**Foundation layer**) onde se encontram os serviços de baixo nível, nomeadamente as bases de dados e o sistema de networking.

Notas:

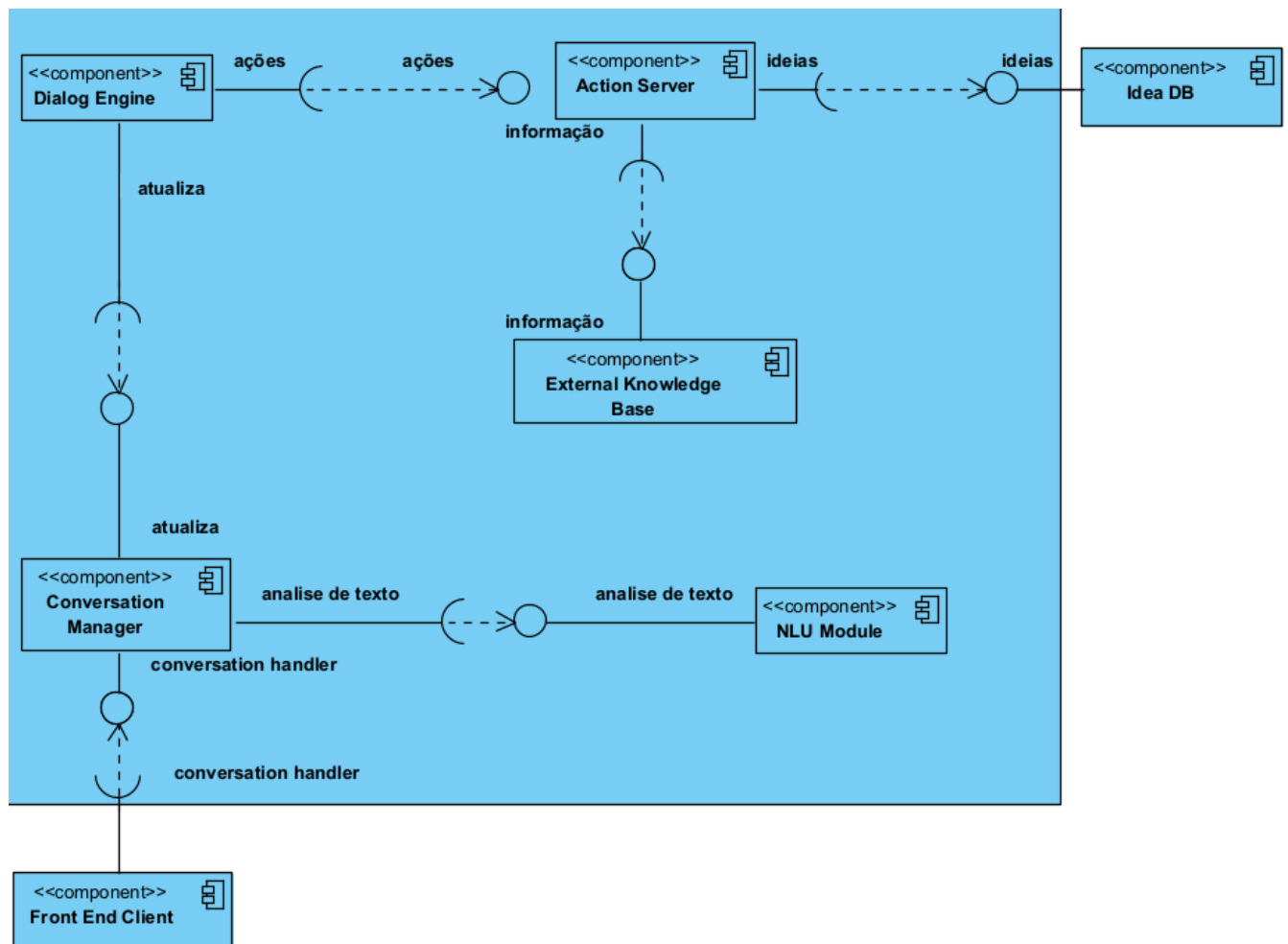
- A camada *Application* tem acesso direto às camadas *UI* e *Domain* de forma a que as suas funcionalidades atuem;
- A camada *Domain* tem acesso direto às camadas *Foundation* e *Technical Services* de forma a que as suas funcionalidades atuem.
- A camada *Domain* e *Technical Services* tem ainda que importar os dados da base de dados que se encontra na camada *Foundation*.

b)



Exercício 5.3

a)



b)

