



Documentação de Arquitetura

O que é um documento de arquitetura?

- É um mapa do software, usado para demonstrar de forma visual como o software está estruturado.
- Trata-se de uma ferramenta útil para comunicação tanto entre desenvolvedores quanto não-desenvolvedores.

Porque usar documentação de arquitetura?

A documentação tem três objetivos principais:

- **Compartilhamento de informação:** transferir informação entre pessoas trabalhando em diferentes áreas do projeto e entre novos participantes.
- **Comunicação:** passar as ideias do arquiteto para os desenvolvedores.
- **Análise:** é o ponto de partida para futuras revisões na arquitetura do projeto.

O Modelo C4

- C4 é um modelo de documentação que consiste de três diagramas e um diagrama opcional:

1. Diagrama de Contexto

2. Diagrama de Contêiner

3. Diagrama de Componente

4. Diagrama de Código (OPCIONAL)

Diagrama de Contexto

- Ele dá o contexto geral do seu sistema, sem entrar em detalhes técnicos.
- O importante a ser mostrado são os atores (usuários) e os sistemas do software que são parte da solução do problema.
- Esse seria o diagrama mostrado para um empresário em uma reunião para explicar o sistema sendo feito, o seu público-alvo, etc.

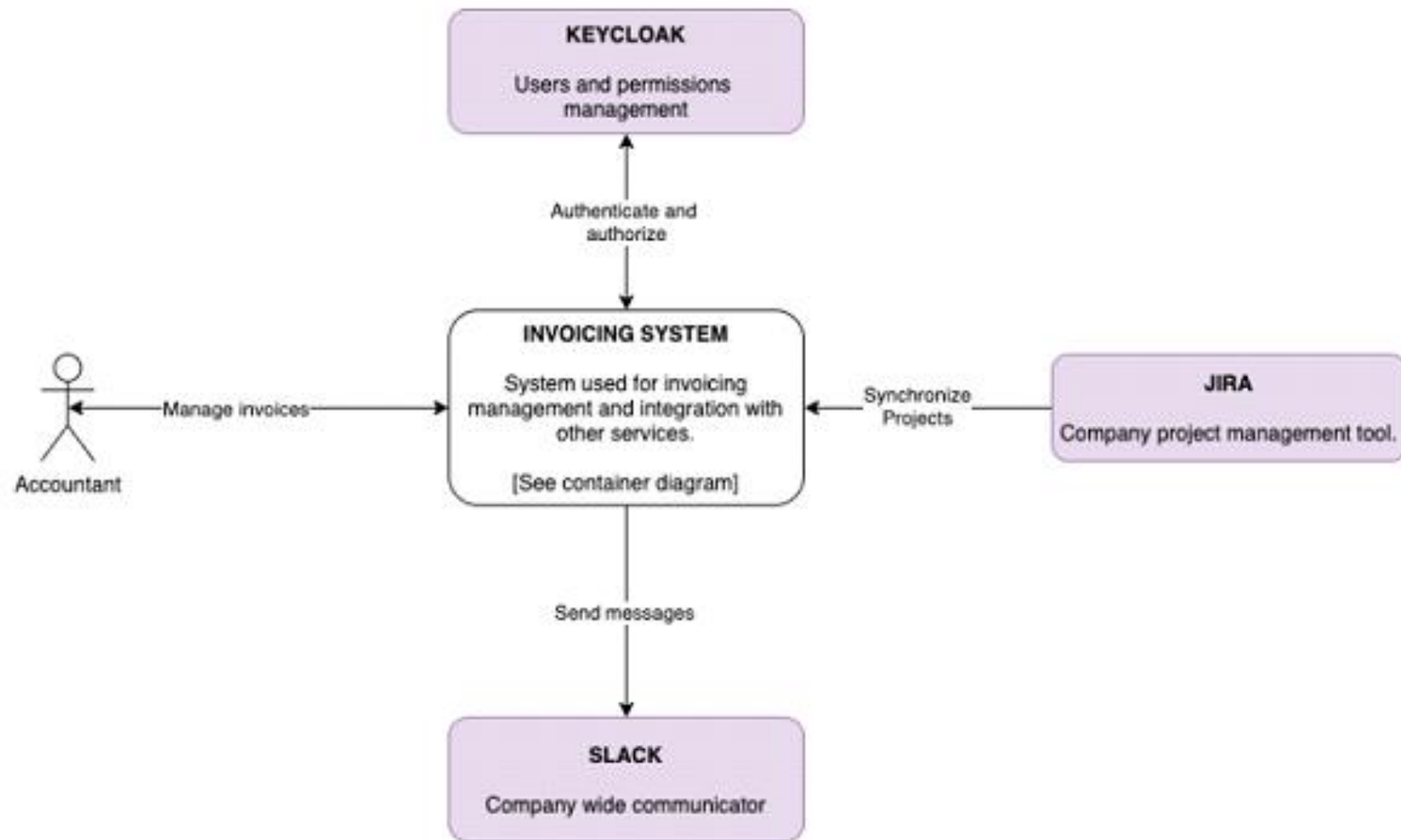


Diagrama de Contêiner

- O foco é em partes do sistema.
- Exemplos: serviços, banco de dados, front-end, armazenamento de arquivos.
- O importante é descrever a tecnologia utilizada em cada contêiner específico e a comunicação entre eles.

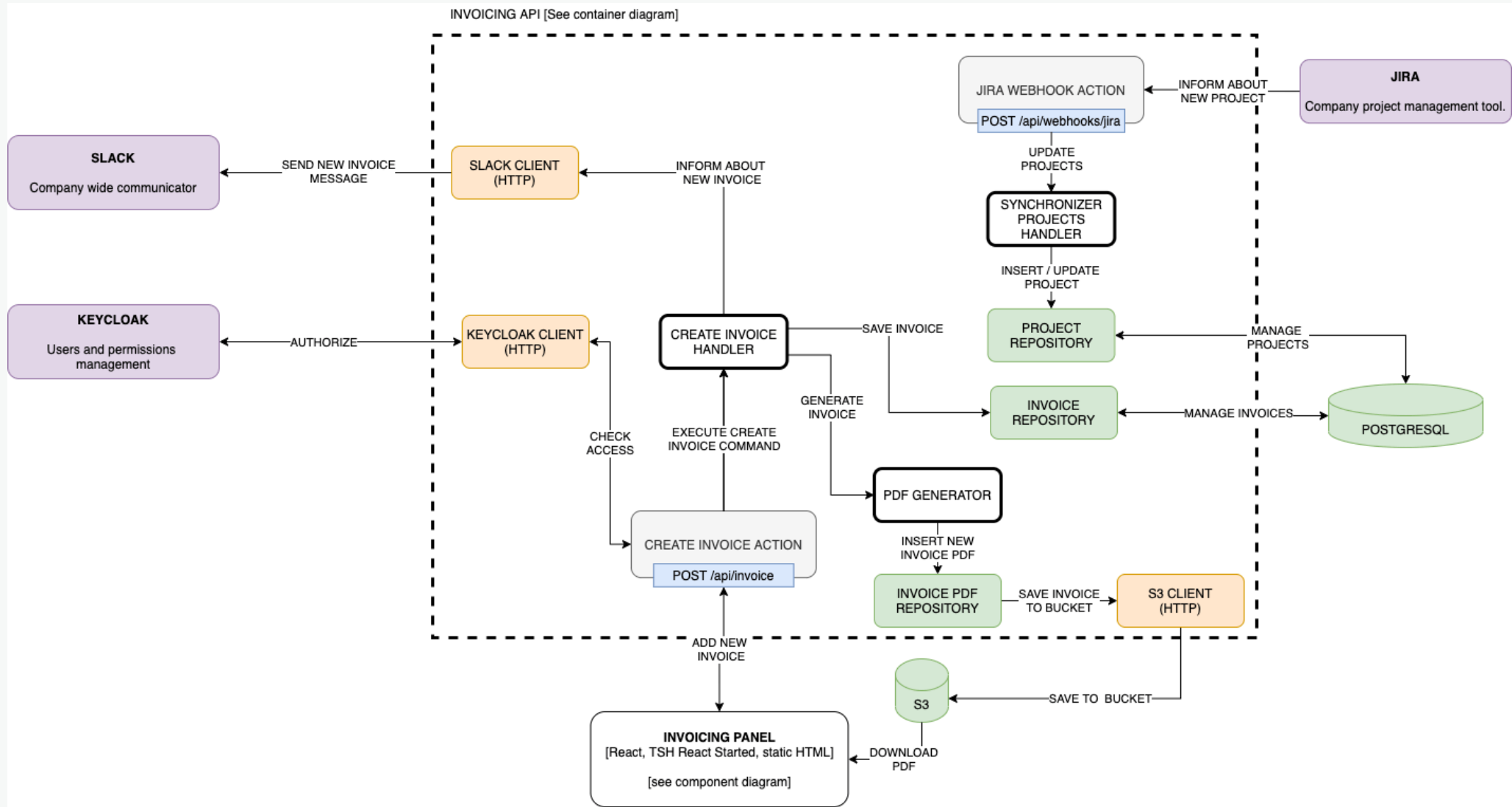
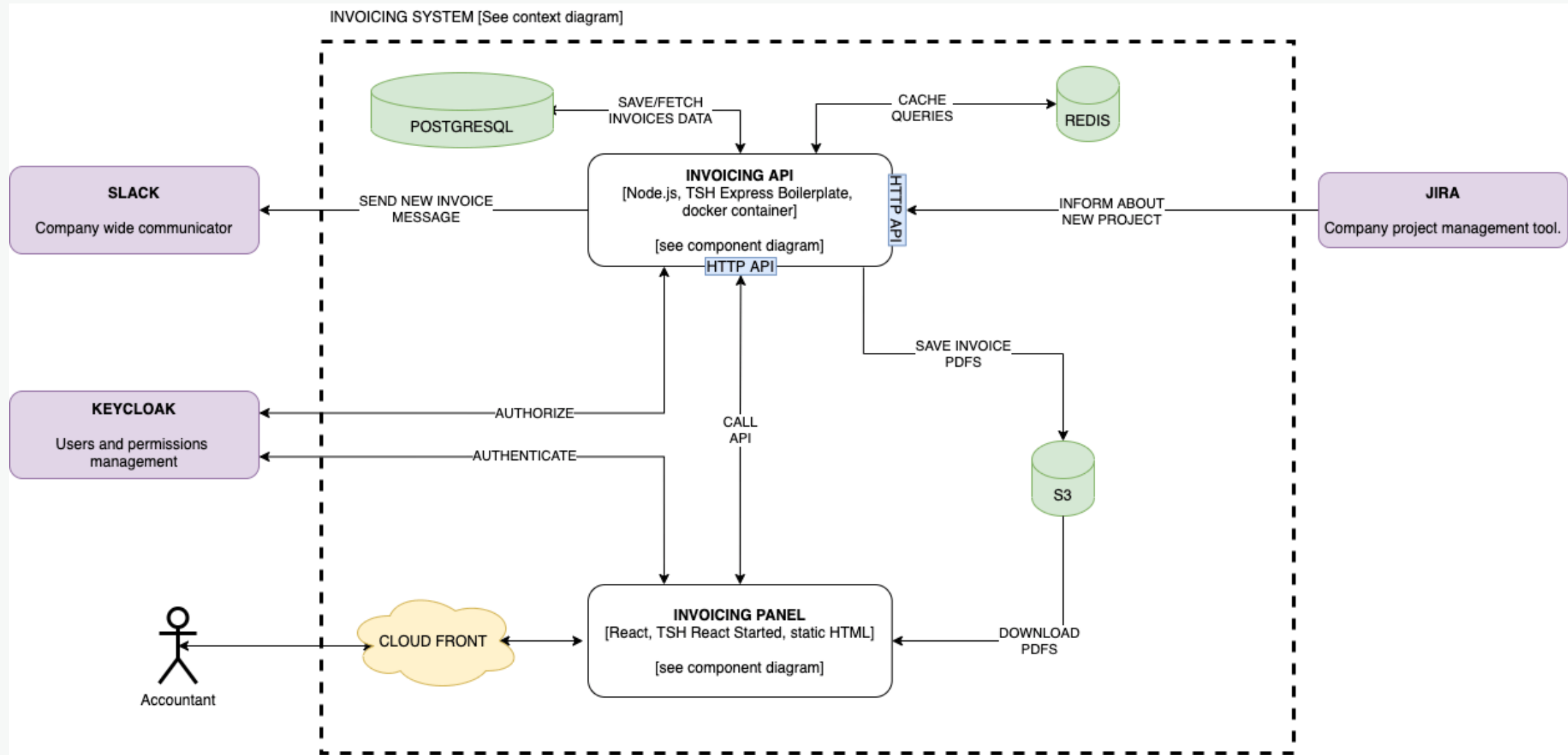


Diagrama de Componente

- Aqui, o foco é em dar mais detalhes da implementação de cada contêiner.
- Serviços, repositórios, tratadores de comando, tudo deve estar nesse diagrama.
- É importante não apenas mostrar as peças de cada componente, mas também a comunicação e relação entre elas.
- Embora não se deve exibir todos os detalhes da implementação, deve-se mostrar os mais importantes.



Dicas para a montagem dessa etapa:

- Toda seta precisa de descrição
- Todo nome usado deve ser o nome que será usado no código.
- Todo fim ou meio de comunicação deve ser visível
- Cada cor deve ter significado (geralmente com uma legenda para isso)

Diagrama de Código

- Novamente: é **opcional**.
- Descreve o código e como as partes se relacionam.
- Pode ser gerado pelo próprio código, gerando um UML.

