

Desenhando a arquitetura do software!



Isaac Felisberto de Souza
Engenheiro de Software

JOGO RÁPIDO!



RESPONDA APENAS

SE FOR

SIM

- Quem já desenha arquitetura do software?
- Quem conhece C4 model?
- Quem não gosta de desenhar? 😅

ROTEIRO DE HOJE!

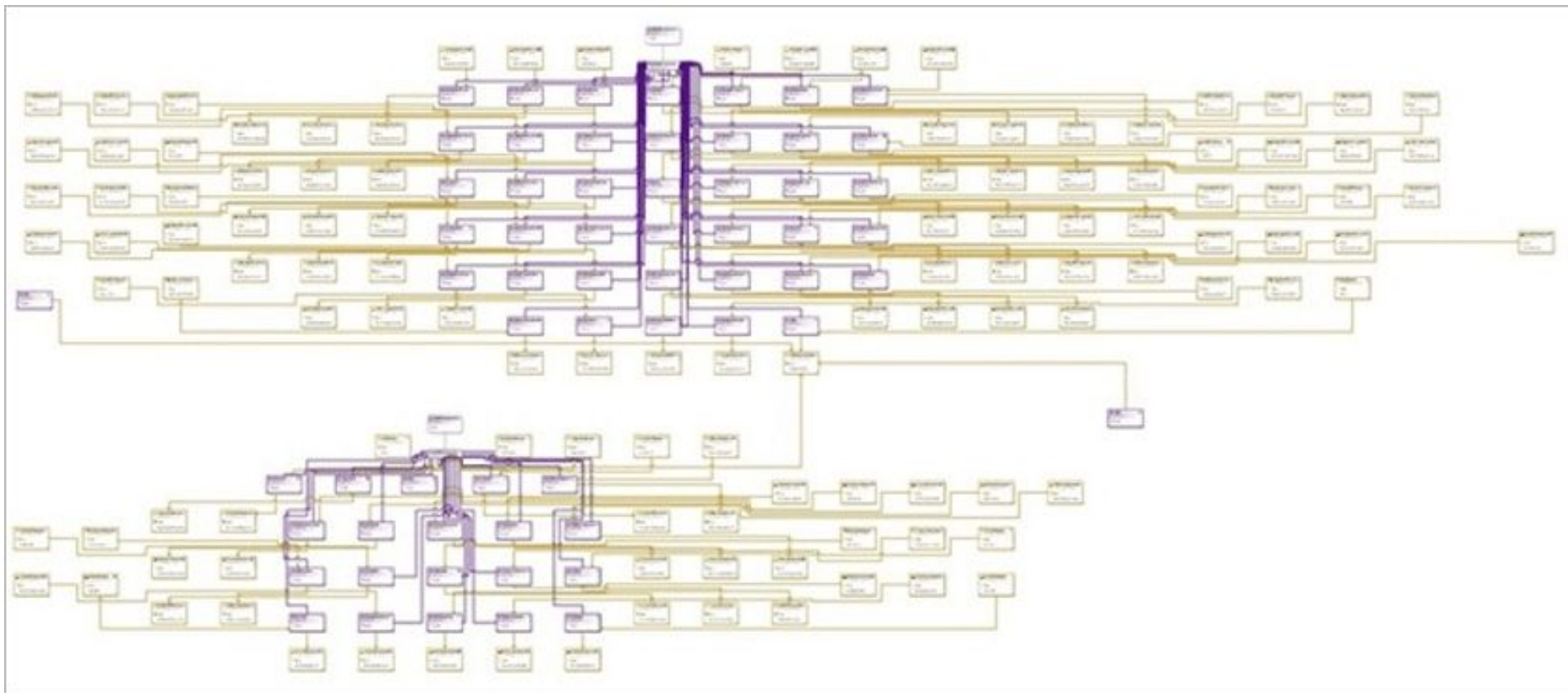
- 1 A IMPORTÂNCIA DO DESENHO.
- 2 DESENHO TÉCNICO E O C4 MODEL.
- 3 DESENHO +
ARQUITETURA DO SOFTWARE.

A IMPORTÂNCIA DO DESENHO



O que as imagens possuem em comum?

Quem fez, provavelmente não desenhou antes de “implementar”.



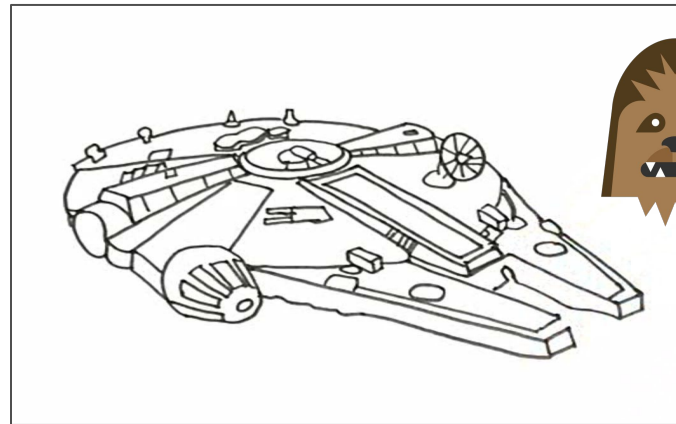
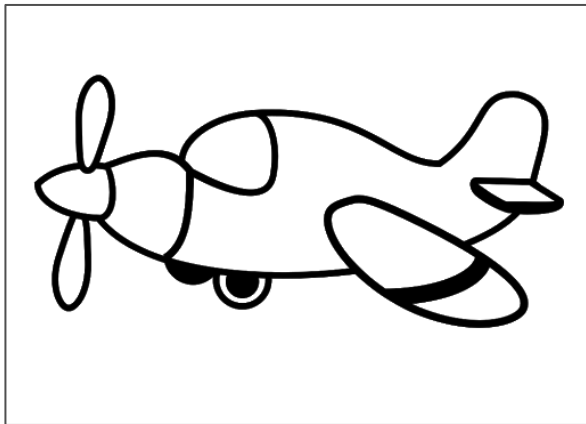
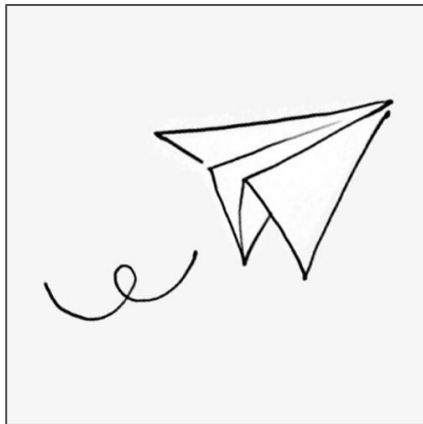


Porque em áreas técnicas, é importante desenhar?

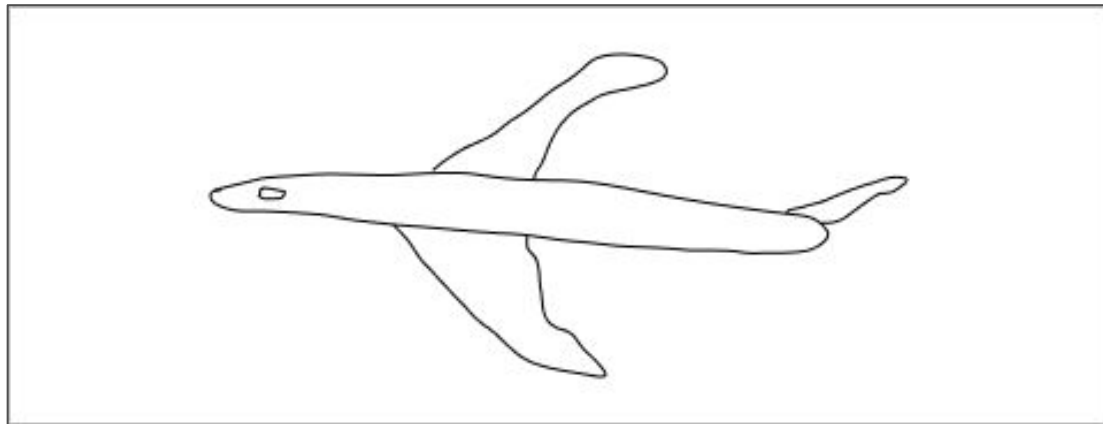
Em várias áreas, desenhar antes de implementar é uma regra!

É um modo de representar de forma visual algo que está no nível da imaginação.

Faça um avião!



O Avião esperado

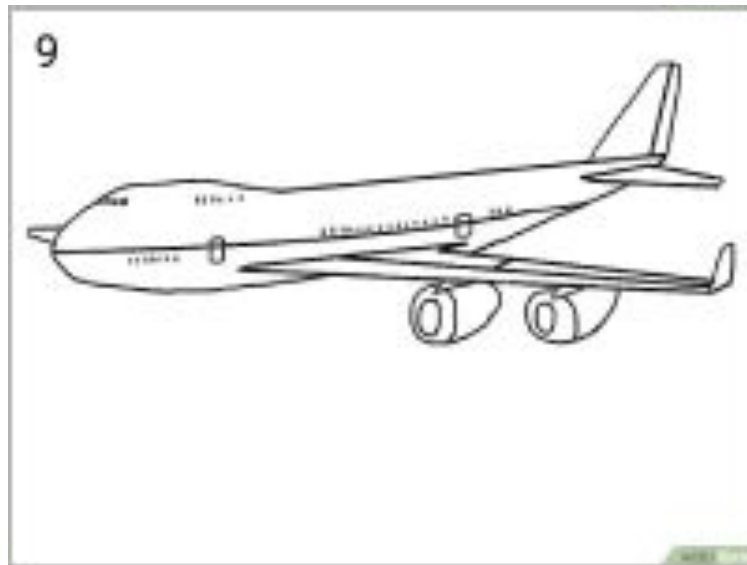


DESENHO TÉCNICO E O C4 MODEL

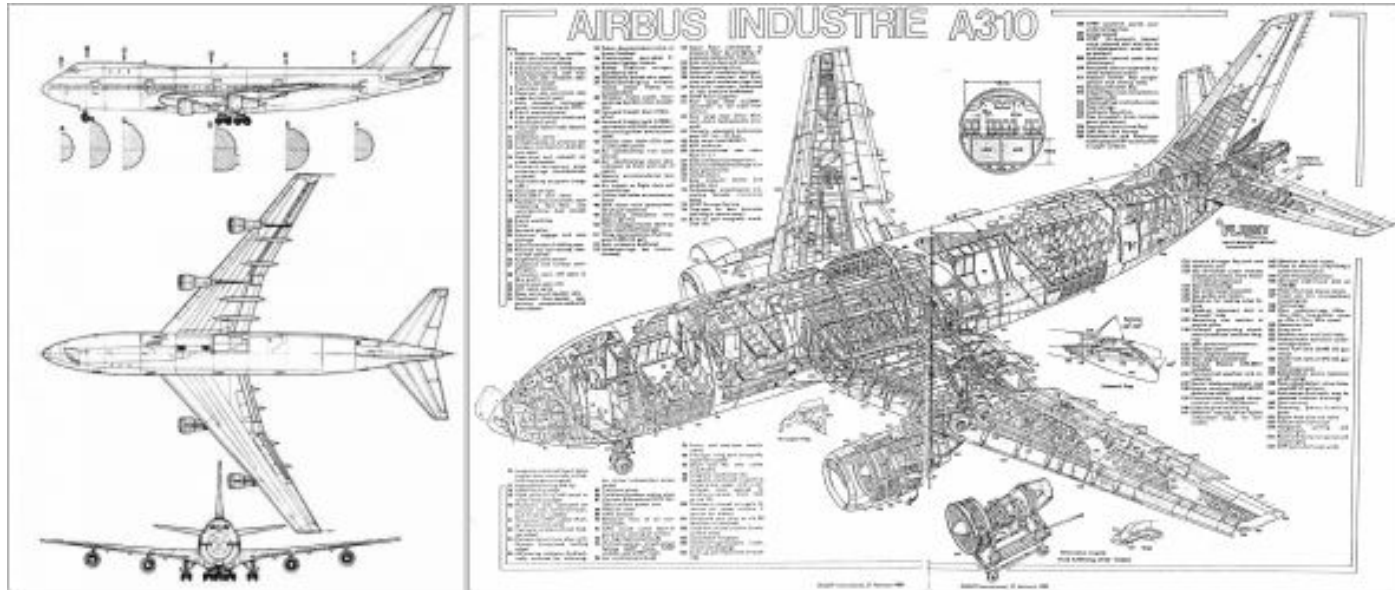
Vamos desenhar o mesmo avião!

Mas agora...

Aplicando uma técnica!

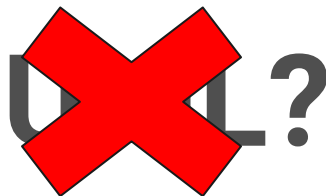
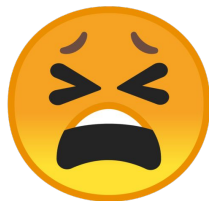


Para fabricar um avião,
é necessário um desenho com mais informações!



Desenho técnico no desenvolvimento de software

- Diagramas
- Notação



Diagramas

São um aliado para área de software,
pois permitem uma representação visual
estruturada e simplificada de um conceito ou ideia,

mesmo sendo às vezes
apenas um conjunto de “caixas e linhas”.

Notação

Determina a forma com a qual os elementos devem ser desenhados ou descritos.

Isso inclui formas geométricas, tipos de linhas, informações auxiliares, dentre outros.

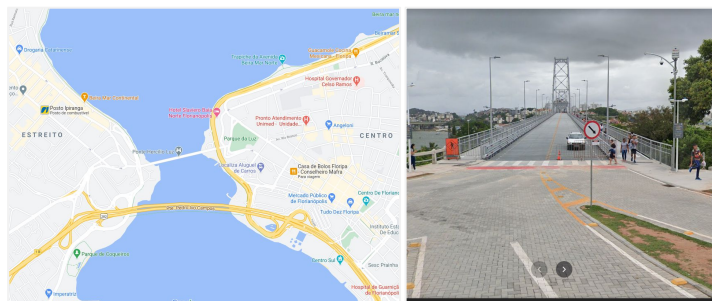
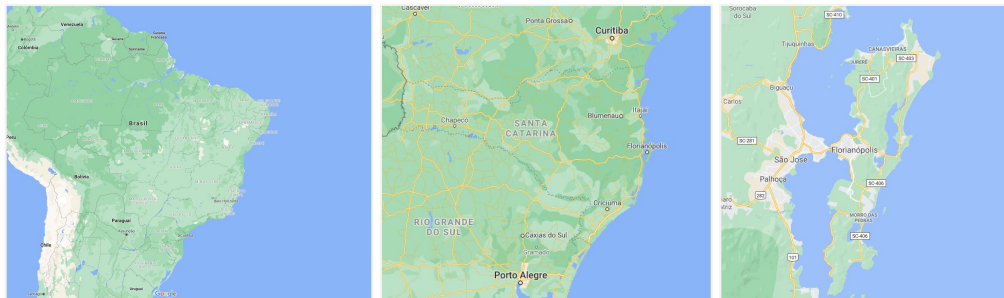
O C4 MODEL

C4 model

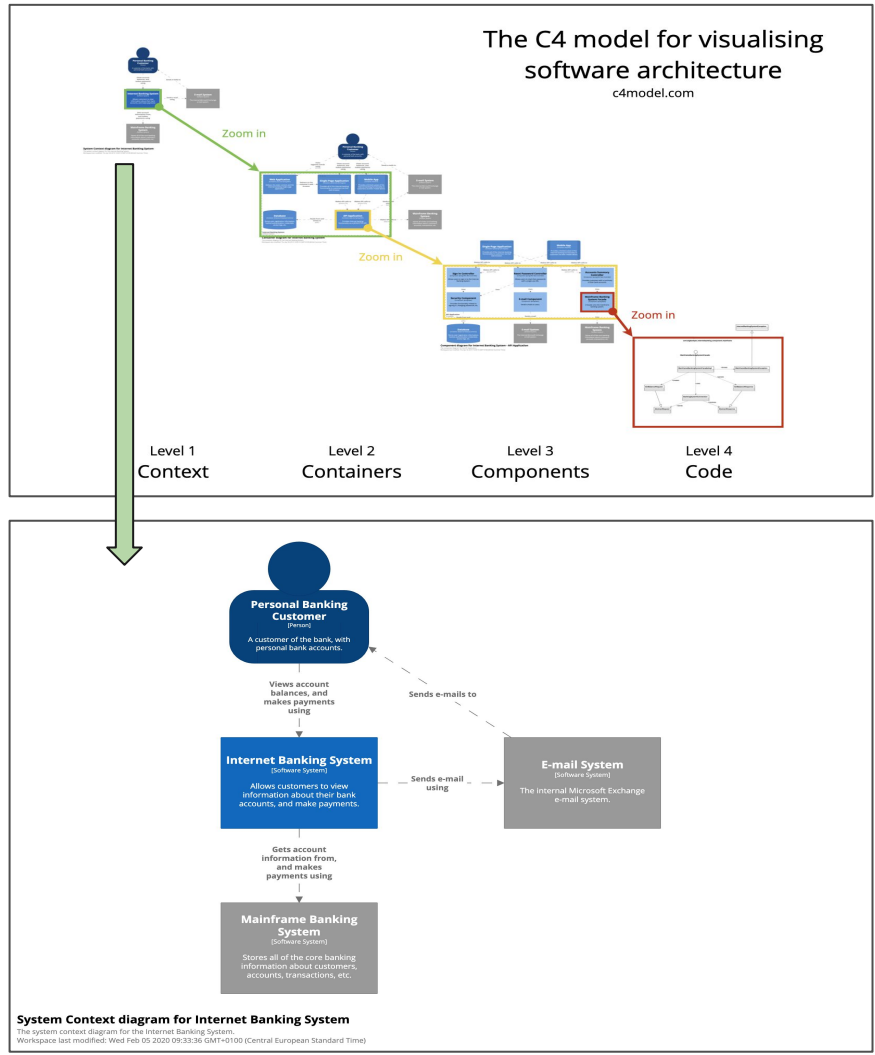
c4model.com

Uma notação simples
para modelar
arquitetura de software

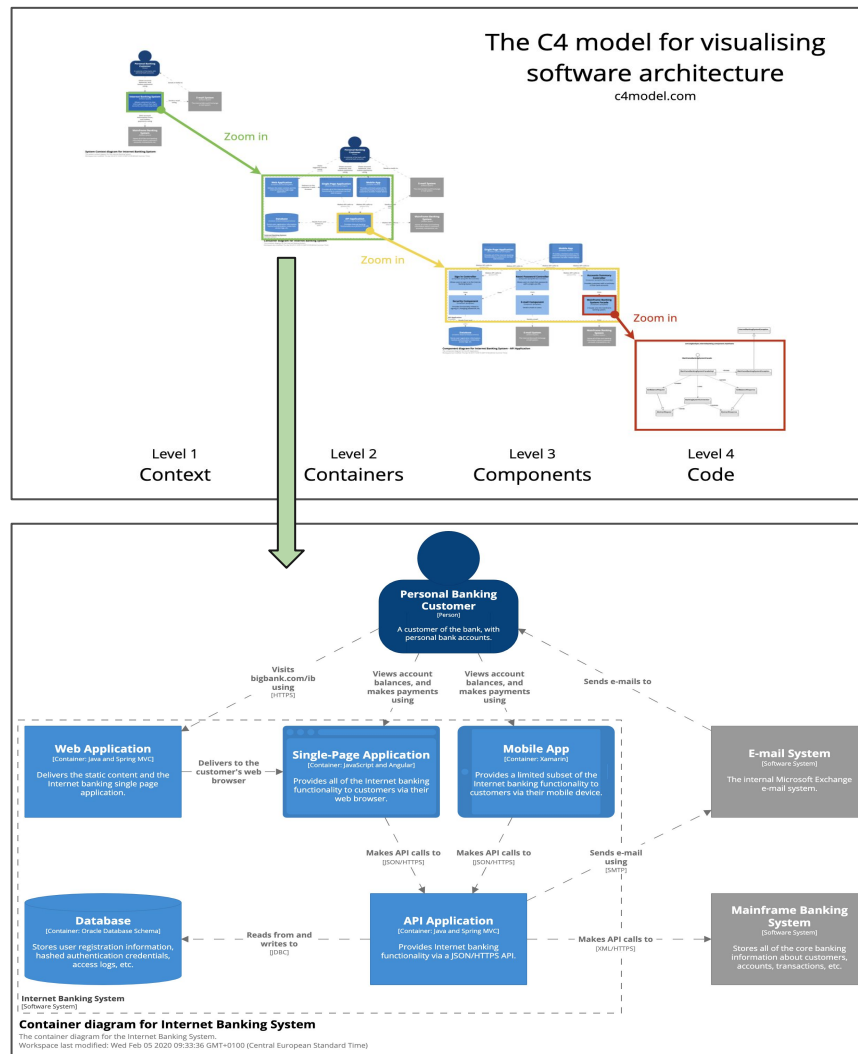
- Foca em uma forma livre de diagramar.
- Sugere boas práticas.
- Propõe a criação do “*Maps of your code*” com vários níveis de detalhes.



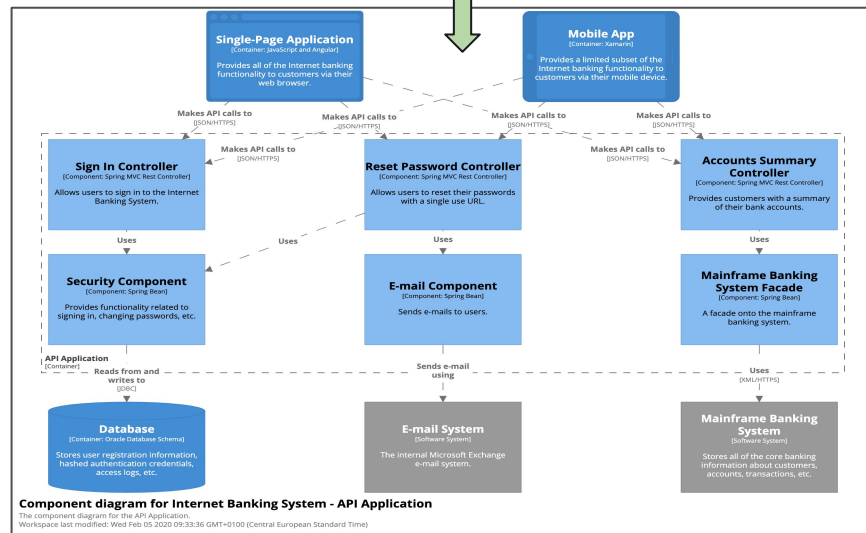
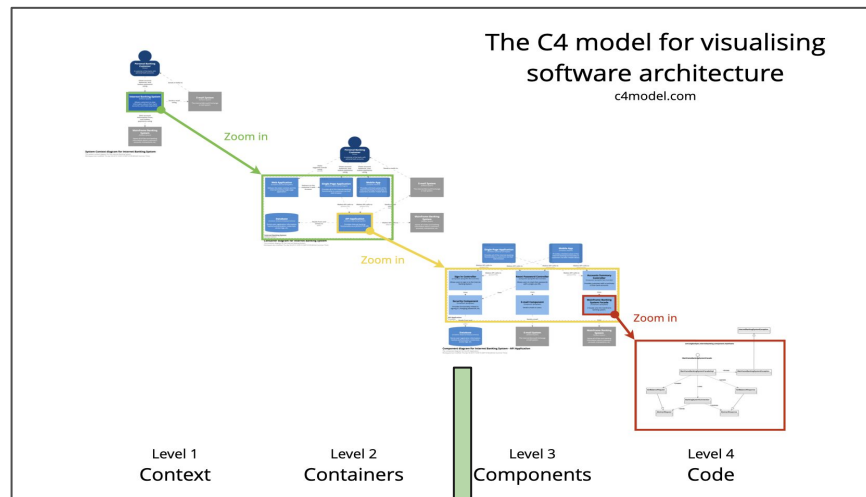
Nível 1: Contexto



Nível 2: Containers



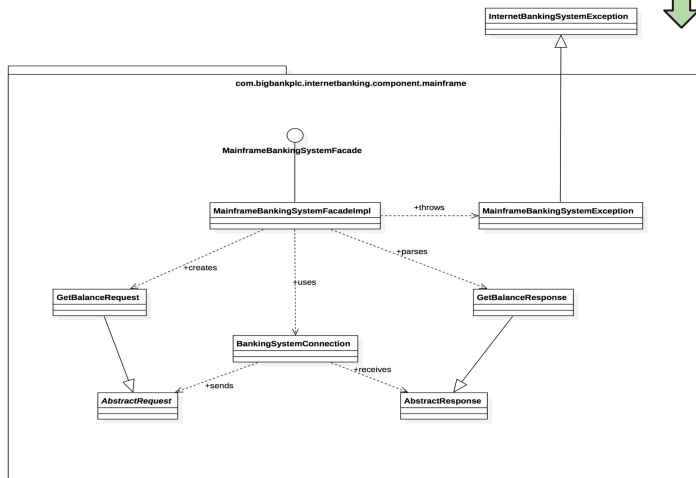
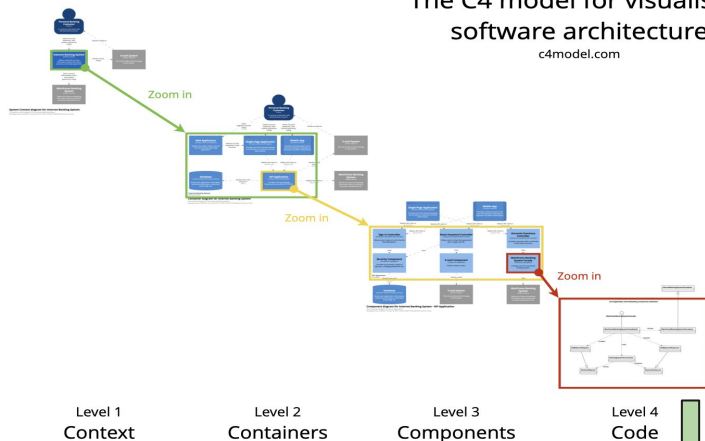
Nível 3: Components



Nível 4: Code

The C4 model for visualising software architecture

c4model.com





Diagramas complementares

- System Landscape
- Dynamic
- Deployment

DESENHO + ARQUITETURA DO SOFTWARE.

BENEFÍCIOS EM DESENHAR A ARQUITETURA DO SOFTWARE



Ajuda a compor
e detalhar ideias.

Permite que outras
pessoas visualizem
melhor a ideia proposta.

Ajuda a validar se a ideia
de fato é boa e viável.

Implementação

Permite um melhor
acompanhamento sobre
a implementação

Pessoas podem verificar
se a implementação está
seguindo o esperado

Esclarecimento

Permite lembrar
o que e como foi feito

Acelera a compreensão
de novas pessoas no time

Ajuda no planejamento de
novas implementações

O QUE UM DESENHO DE ARQUITETURA DO SOFTWARE DEVE TER?

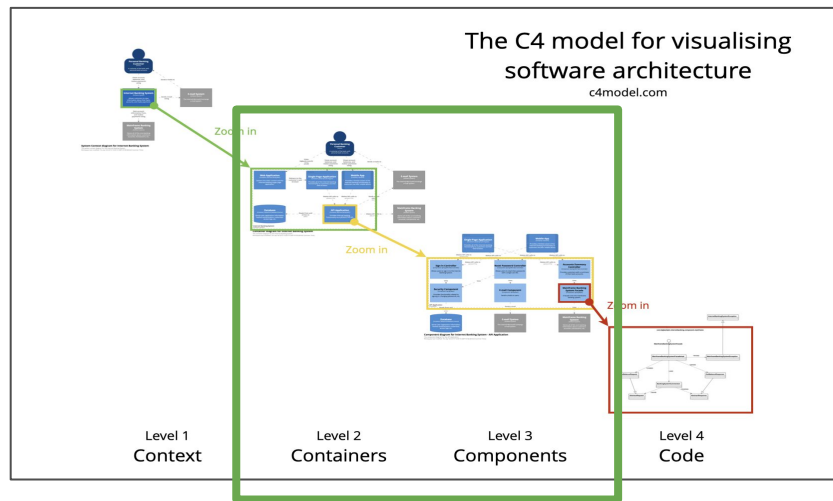
Elementos

Descrições básicas

Relações

Hierarquia ou fluxo

Vamos utilizar o notação C4 model, nos níveis de container e component.



Uma solução fictícia, chamada Guia Hóspede, o domínio é: RESERVA DE HOTÉIS.

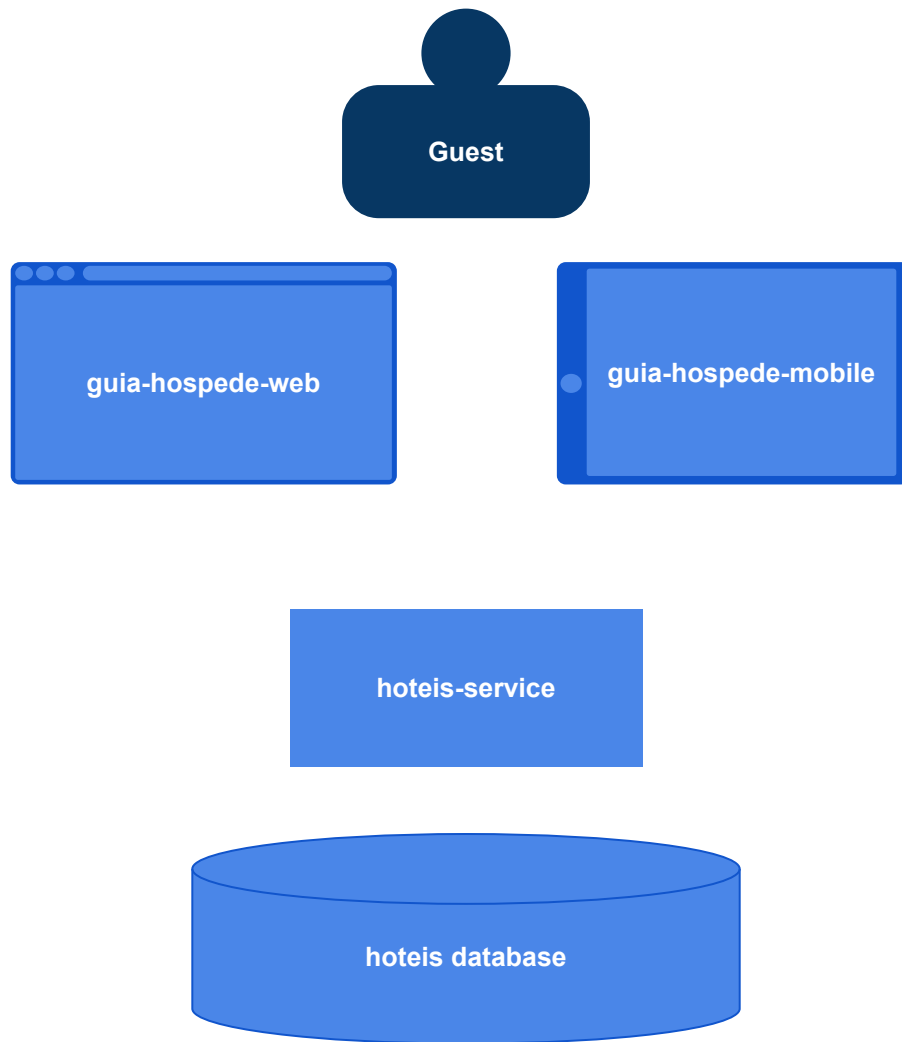
Diagrama de Container

Elementos

Descrições básicas

Relações

Hierarquia ou fluxo



Indexed Data

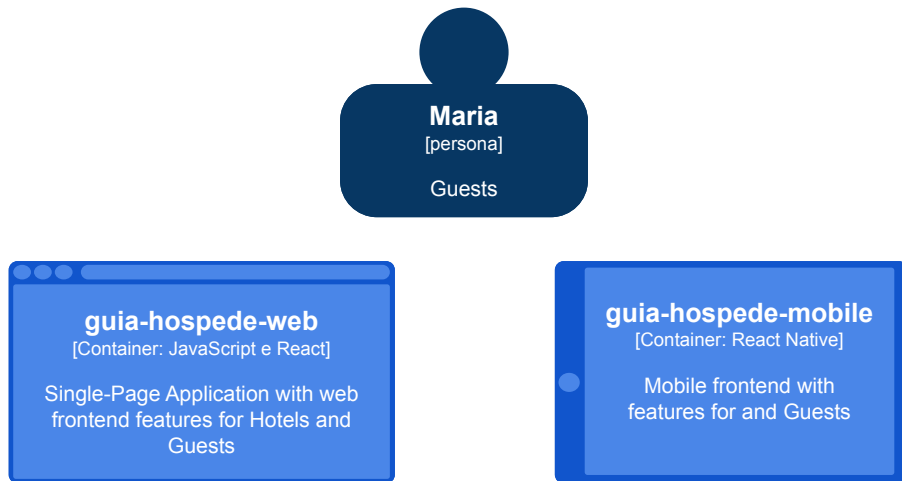
Event Bus

Elementos

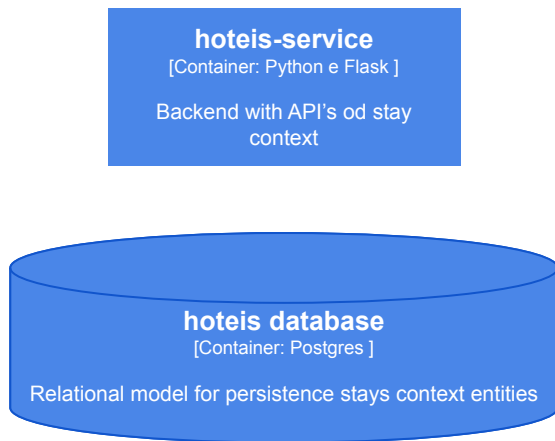
Descrições básicas

Relações

Hierarquia ou fluxo



- Nome
- Estereótipo ou tecnologia utilizada
- Descrição curta



Indexed Data
[Container: ElasticSearch]

Indexed data of hosting domain model

Event Bus
[Container: Kafka]

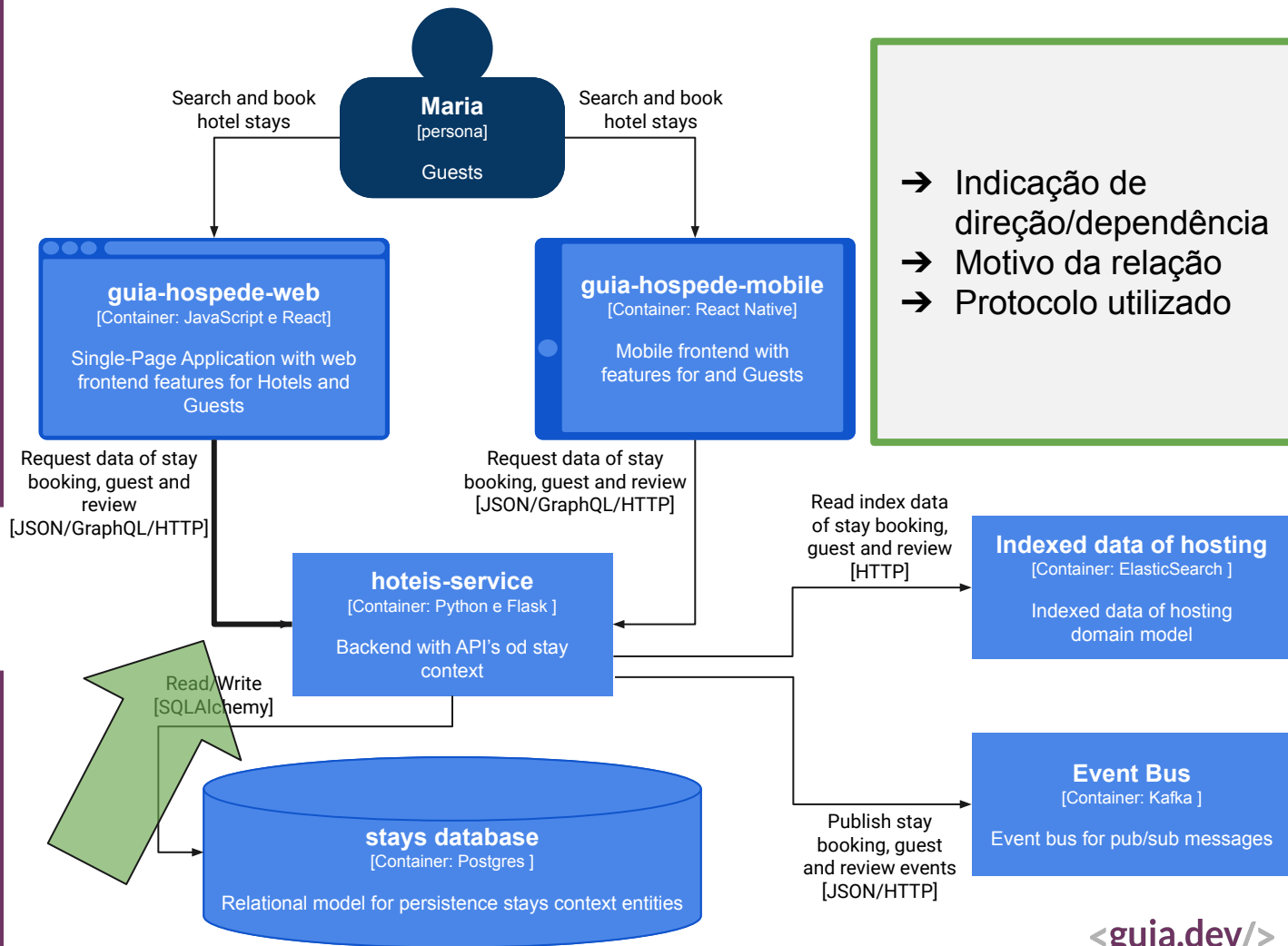
Event bus for pub/sub messages

Elementos

Descrições básicas

Relações

Hierarquia ou fluxo



Elementos

Descrições
básicas

Relações

Hierarquia ou
fluxo

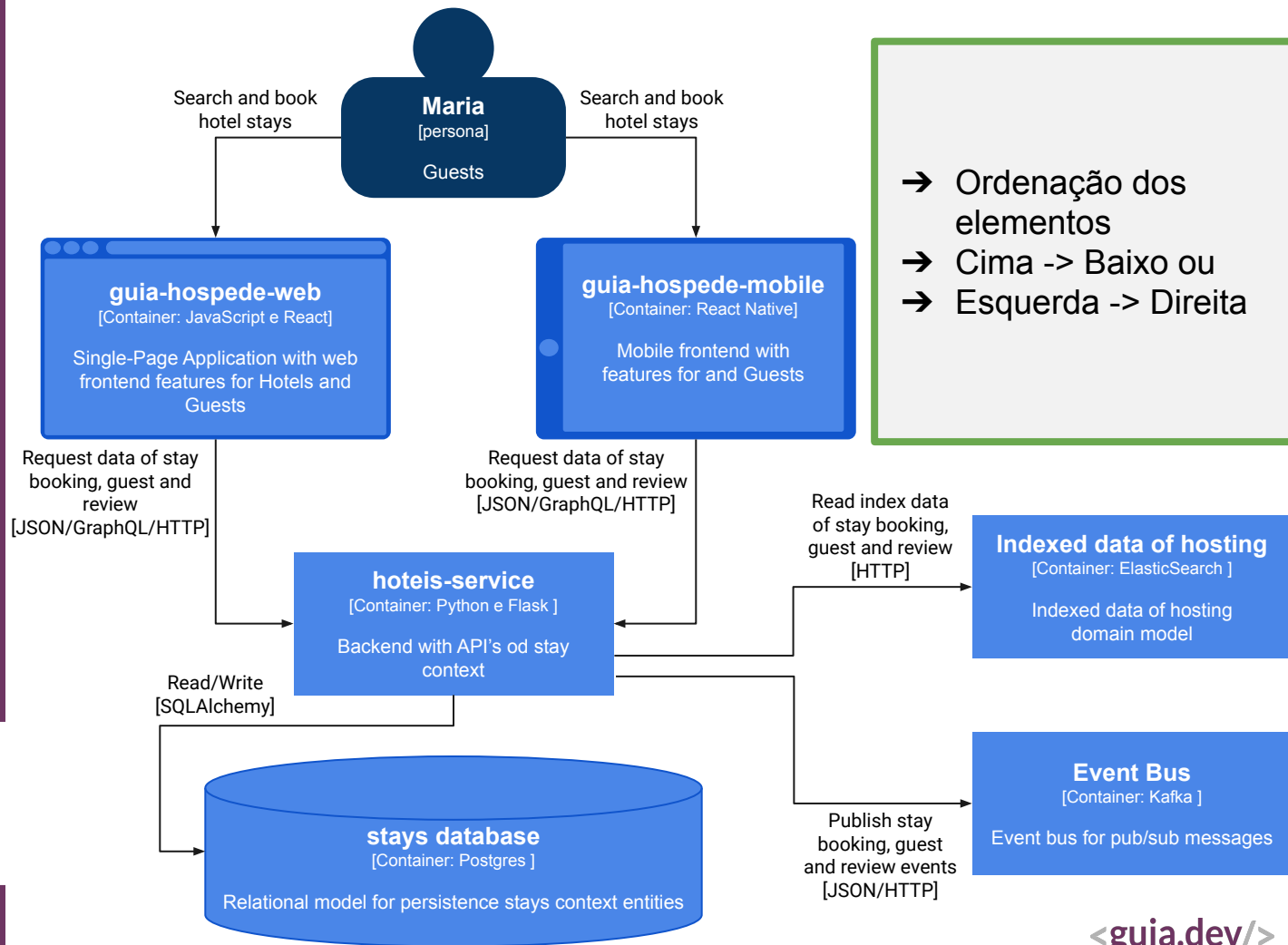
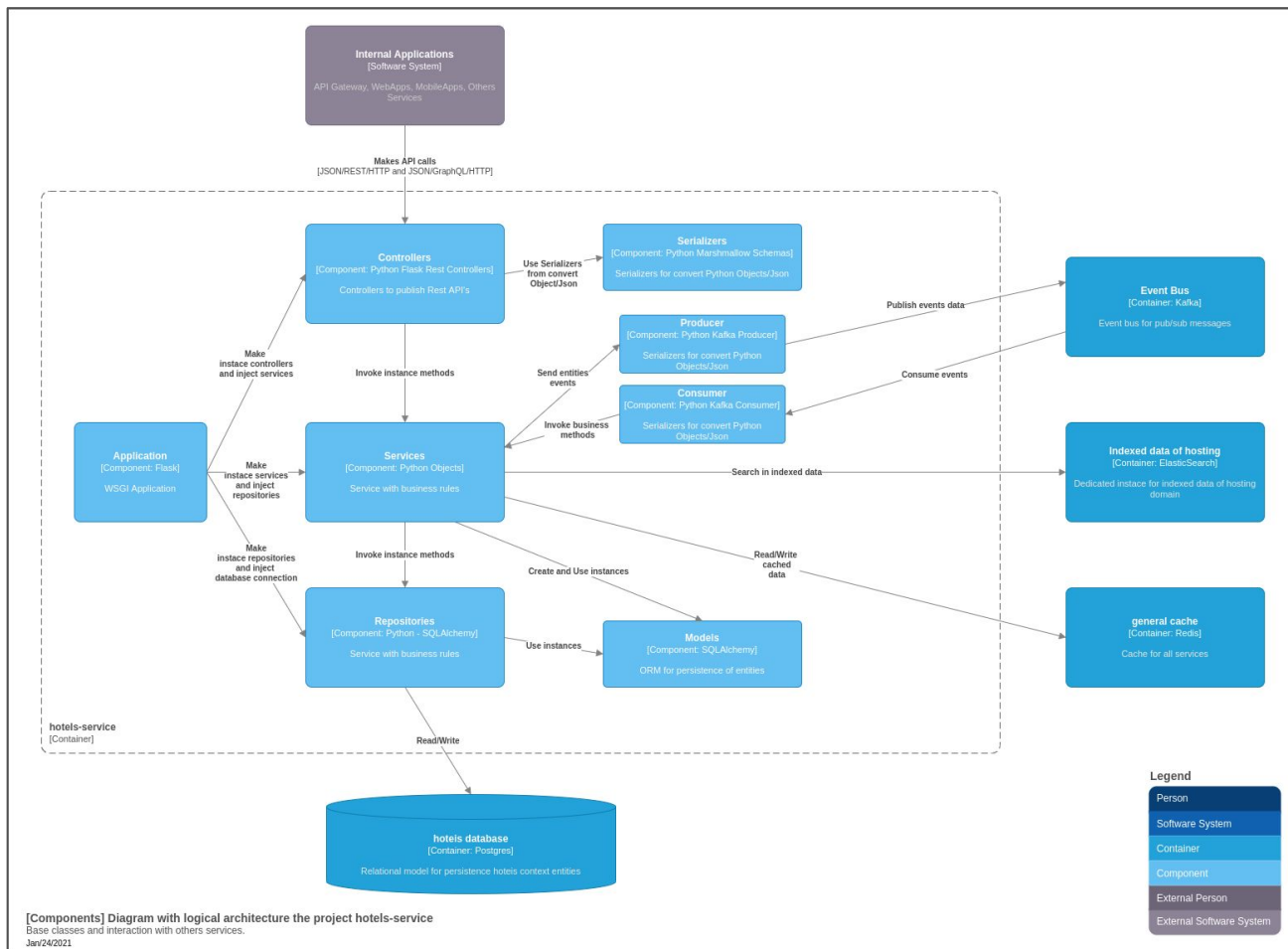
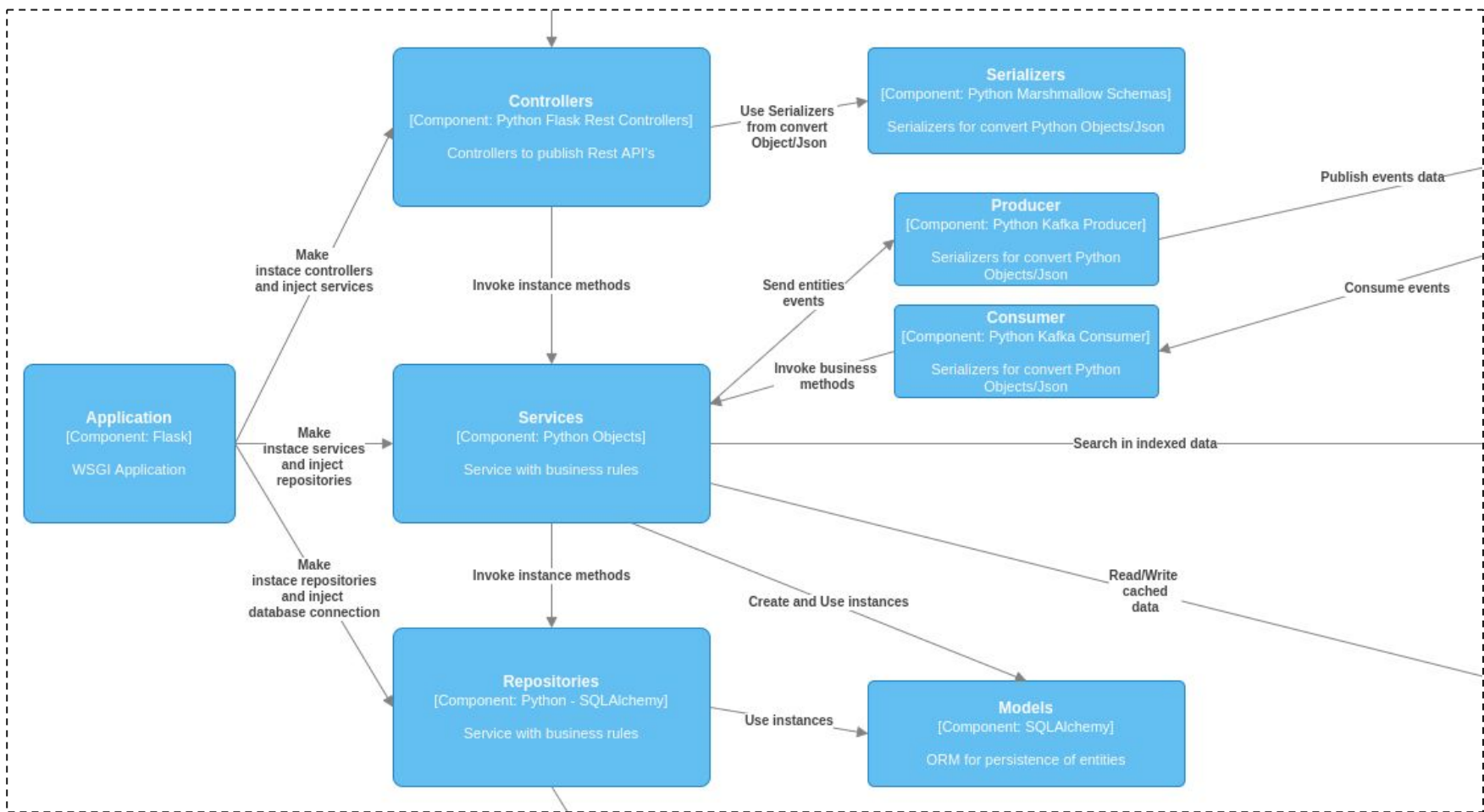


Diagrama de Component







Informações que são apresentadas nestes desenhos

Container:

- Os projetos existentes.
- Informações como (nome, tecnologias, descrição básica).
- Relação de dependência entre eles.
- Descrição básica da relação.
- Protocolo de comunicação.
- Estilos/Padrões arquiteturais.
- Outros serviços como banco de dados, Cache, dados indexados.

Component:

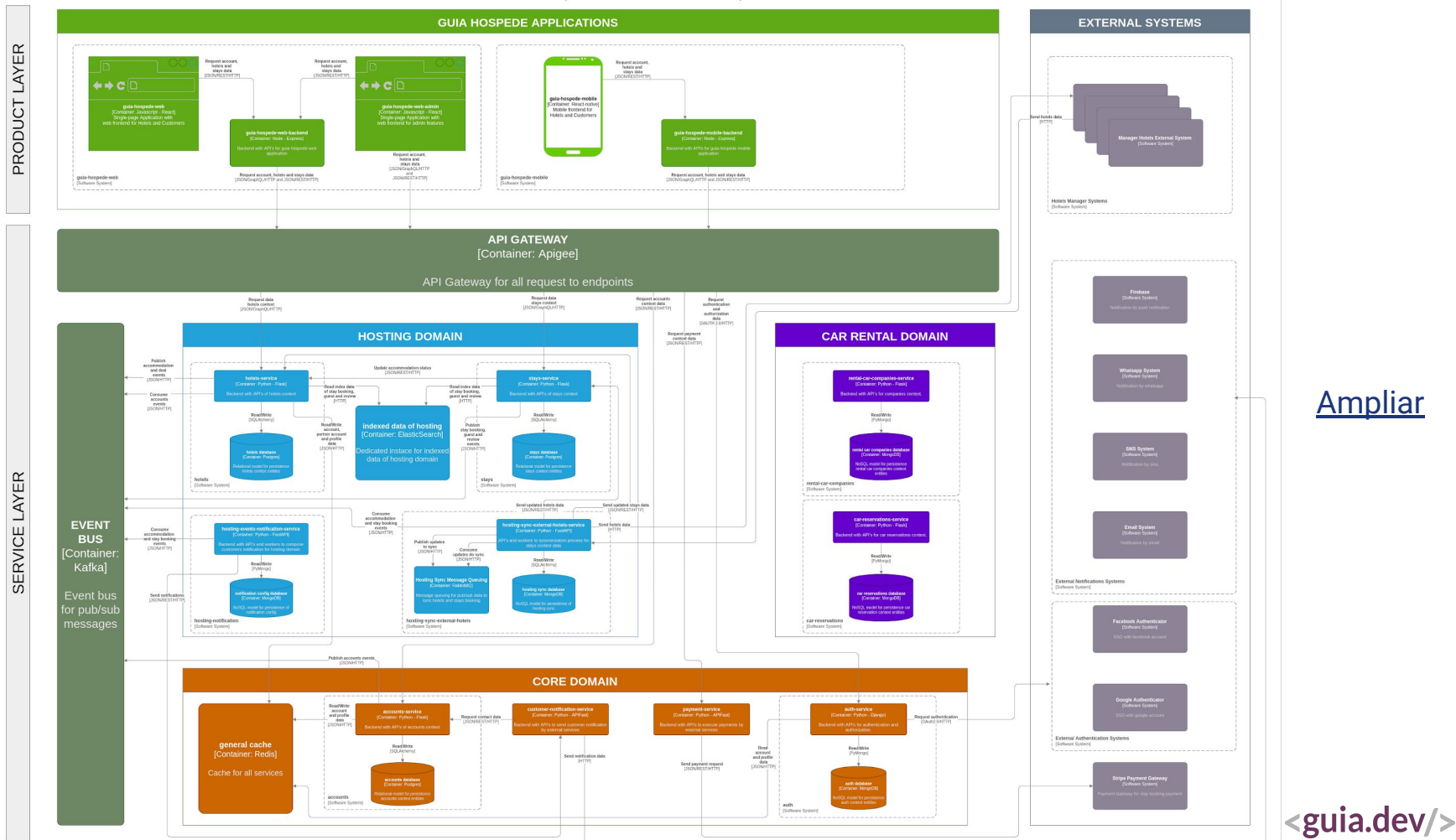
- Componentes/Classes base do projeto.

E PARA SOLUÇÕES
MAIORES

COMO
ARQUITETURA DE
MICROSERVICES?

Diagrama: System Landscape
+
Diagrama: Component

- Elementos
- Descrições Básicas
- Relações
- Hierarquia / Fluxo
- **Agrupamentos**





**Informações que são
apresentadas neste
desenho**

System Landscape + Component :

- Visão dos domínios
- Organização da arquitetura
- Outros padrões arquiteturais
- Os serviços externos utilizados

Mas desenhar isso é chato!

O que fazer então quando:

- O time não lembra destas informações.
- Novos(as) integrantes entram no time.
- Outros times desejam entender.

- O tempo das pessoas será utilizado para explicar.
- Várias e repetidas vezes.
- Pense um time com 5, 10, 100, 1000 pessoas! Quanto esse tempo custa?

E o que acontece quando quem detém as informações troca de time ou empresa?

Pessoa arquiteta de software,
boa,
é a que bota a mão na massa!

SIM!

Deve botar a mão na massa!

E...

precisa também saber expressar e apresentar as ideias!

precisa documentar e não apenas reter informação na memória.

**Desenhar a arquitetura
é uma forma de fazer isso!**

Tornando o desenho interativo

- Publicar o desenho através de um URL.
- Utilizar em reuniões.
- Ter links para:
 - Código fonte
 - Documentação de API's
 - Acesso aos ambientes publicados
 - Outros detalhes

O que não desenhar!

- Coisas que sofram muitas mudanças:
 - Diagrama completo de classes.
 - Diagrama de entidades-relacionamento
- Não incluir no desenho:
 - Detalhes de API's
 - Versão de tecnologias utilizadas
 - Descrições detalhadas

Ferramentas

- Durante planejamento, Quadro Branco!
- Diagrams.net.
- Lucid Chart.
- Microsoft Visio.
- PlantUML.

Outras em c4model.com

Em resumo, considerando:



As informações que um desenho pode apresentar.

A ajuda no planejamento e execução.

A economia de tempo das pessoas que explicam sempre as mesmas coisas.

O conhecimento compartilhado com os demais.

Desenhar

a arquitetura do software

se torna um sinal de

maturidade, responsabilidade

e profissionalismo.

OBRIGADO!



Isaac Felisberto de Souza
Engenheiro de Software

isaacsouza@gmail.com

linkedin.com/in/isaacfsouza



<guia.dev/>

**Um guia para auxiliar o
Desenvolvimento de Software.**

Através de conteúdo para desenvolvedores(as).



www.guia.dev



Guia Dev (linkedin.com/company/guiadev)



@guia_dev

Siga! e compartilhe ;-)

<guia.dev/>