

Projeto Integrador I

IES011 – Engenharia de Software I

Prof. Esp. Vickybert Freire

1 OBJETIVO

Utilize o documento [Projeto Integrador I - Geral.pdf](#) como base para o desenvolvimento do Projeto Integrador, cujo objetivo é guiar e padronizar o trabalho a ser desenvolvido pelas três disciplinas envolvidas (Design Digital, Desenvolvimento Web I e Engenharia de Software I).

O **objetivo deste documento** é definir o escopo do trabalho a ser desenvolvido e entregue com **item específico** da disciplina **Engenharia de Software I**.

Em conformidade com a ementa da disciplina, essa atividade tem a pretensão de conduzi-los no aprendizado de: Identificar as características de Sistemas de Informação, seus tipos, viabilidade técnica; explicar as características de um sistema, seus componentes e relacionamentos.

2 ITENS ESPECÍFICOS - ENGENHARIA DE SOFTWARE I

2.1 APRESENTAR AS FERRAMENTAS, LINGUAGENS E SOFTWARES QUE SERÃO UTILIZADOS

A escolha das linguagens, bancos de dados, frameworks e ferramentas são decisões importantes para um time de desenvolvimento ao planejar a construção de algum sistema. Por exemplo, um time pode optar pelo uso de bancos relacionais caso exista grande necessidade de consistência de dados e normalização, ou por um banco não-relacional caso seja esperado um volume de consultas muito elevado.

Documente quais foram as decisões tomadas pelo time, justificando, para cada linguagem, framework, etc.

- a. Linguagem de back-end;
- b. Framework de back-end;
- c. Framework de front-end;
- d. Sistema gerenciador de banco de dados;
- e. Versionador de código;
- f. Ferramenta de diagramação UML;
- g. Outras ferramentas.

Exemplos de ferramentas: Confluence, GitHub, Figma, VSCode, Diagrams.net, Java, Spring, Kafka

Exemplos de justificativas:

“The early architecture of Uber consisted of a monolithic backend application written in Python that used Postgres for data persistence. Since that time, the architecture of Uber has changed significantly, to a model of microservices and new data platforms. Specifically, in many of the cases where we previously used Postgres, we now use Schemaless, a novel database sharding layer built on top of MySQL.” (fonte:

<https://eng.uber.com/postgres-to-mysql-migration/>)

“O nosso planejador (...) é baseado na biblioteca para IO assíncrono Vert.x, para JVM. Vert.x é uma excelente escolha para implementar serviços com IO assíncrono de alto rendimento com muitas conexões de diferentes clientes em paralelo provendo baixa latência.” (fonte: <https://medium.com/olxbrasiltech/low-code-flerta-com-busca-e-recomendação-7f9098201969>)

“After gathering evidence, the team decided to adopt Flutter, thanks to its excellent testing capabilities, such as built-in testing infrastructure for Unit, Integration, and End-to-End tests without the need for rendering to the screen. Overall, — compared to other options — they also found the Flutter development experience to be superior, with better hot reload capabilities, robust official documentation, and a more stable API.” (fonte:

<https://building.nubank.com.br/scaling-with-flutter/>)

2.2 RESOLUÇÃO DOS DESAFIOS TÉCNICOS

Conforme especificado no documento Geral, cada projeto necessitou em sua concepção de um desafio técnico, como alta volumetria de acessos, alta volumetria de dados tratados, comunicação real-time, identificação de padrões, etc (requisitos não-funcionais).

Nesta seção é esperado que vocês descrevam como planejam resolver esse desafio técnico.

Destaque qual é o desafio, como será a resolução, fontes de consulta/estudo. Utilize de diagramas para explicitar a funcionalidade.

Exemplos:

<https://netflixtechblog.com/engineering-trade-offs-and-the-netflix-api-re-architecture-64f122b277dd>

<https://netflixtechblog.com/system-architectures-for-personalization-and-recommendation-e081aa94b5d8>

<https://medium.com/olxbrasiltech/low-code-flerta-com-busca-e-recomendação-7f9098201969>

2.3 DESENHO DA ARQUITETURA DE COMPONENTES VISÃO GERAL

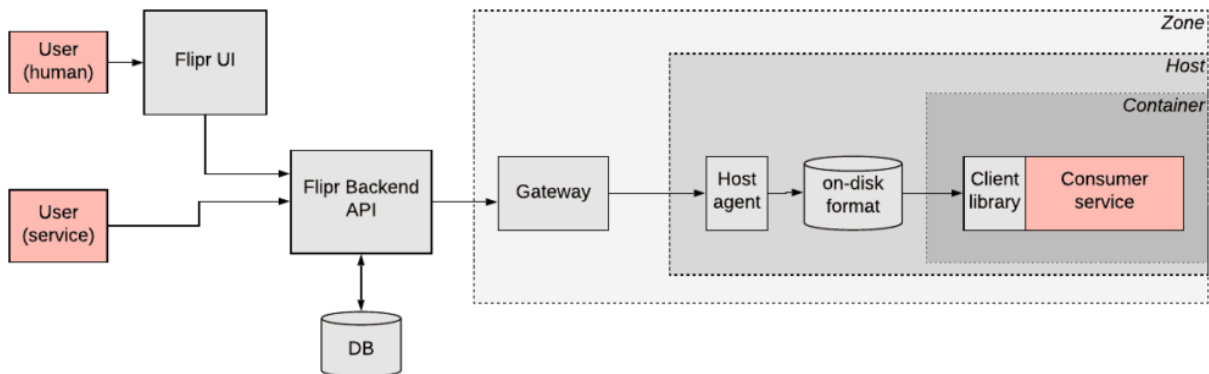


Figura 1. Exemplo de arquitetura de componentes apresentando uma visão geral do sistema

2.4 MODELAGEM DO BANCO DE DADOS (MODELO ENTIDADE RELACIONAMENTO)

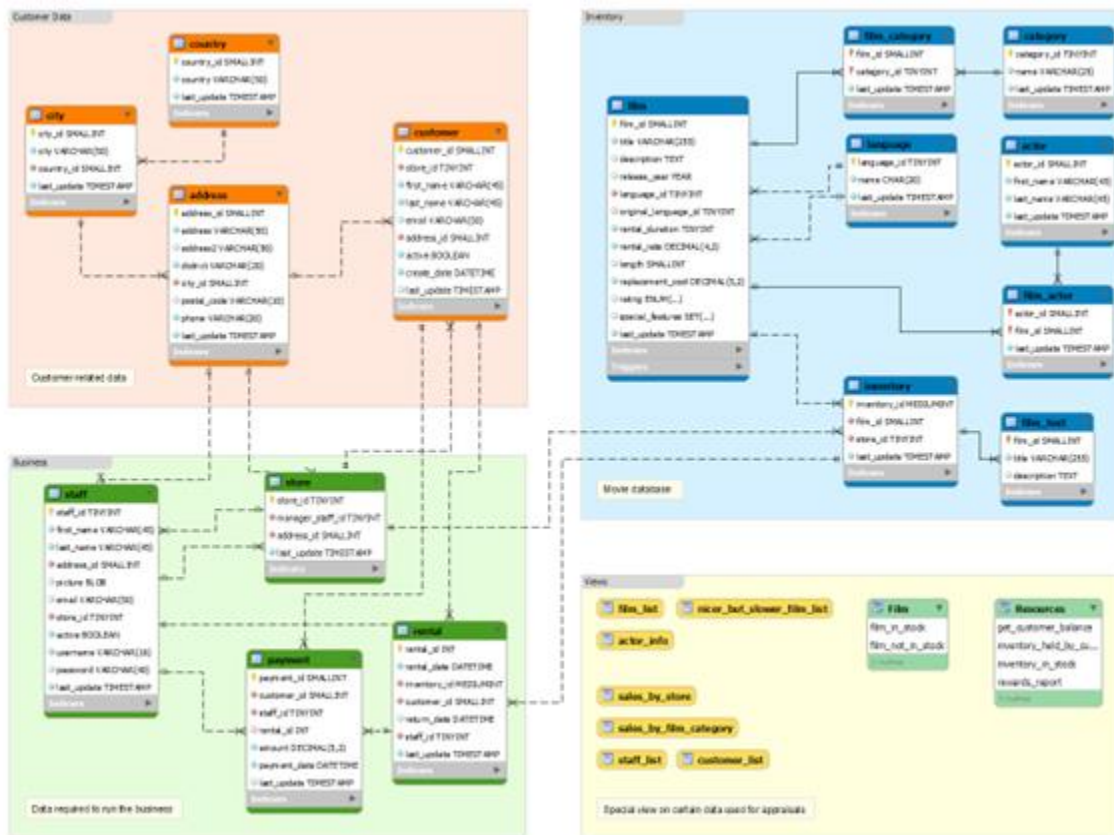


Figura 2. Exemplo de modelo entidade relacionamento

