1. Apresentacao do ATAM

Método arquitetura análise tradeoff, é uma tecnica utilizada para a reducao de riscos, normalmente definido no inicio do desenvolvimento do software. O objetivo principal, é encontrar a melhor arquitetura do sistema de softaware com base nos pontos de sensibilidade e trade-offs.

Trade-offs(perder-ganhar) são decisões que as empresas tomam em abrir mão de uma qualidade ou outros aspetos do sistema, para ter ganhos noutras qualidades do sistema ou aspetos. Normalmente ao perder uma qualidade, a intensão é ganhar pelo menos duas qualidades. Basicamento trade-offs são medidas estrategicas usadas onde os resultados não são imediatas mas ao fim de longo prazo são compensadores.

1. Apresentacao dos objetivos de negocio. (Driver)

O objetivo do projeto é desenvolver uma aplicacao onde os utilizadores podem gerir as funcionalidades de uma sandes bem como, avaliar, comentar, encomendar e doar uma sandes. No caso de um cliente, apenas é possivel se registar no sistema, avaliar uma sandes, fazer um comentario a uma sandes e por fim poder encomendar uma sandes. No caso de um voluntario, so é permetido fazer o levantamento das sandes para as doacoes. O adminstrador tem o acesso total, tais como as sandes, utlizador, avaliacao, comentario e a encomenda, assim podendo criar, atualizar e eliminar se possivel os elementos referidos.

*Os drivers estao no readme da parte 2*

1. Apresentar Arquitetura

?Apresentar os diagramas?

A arquitetura utilizada é o Model-View-Controller(MVC) é um padrao de software que visa separa os conceitos em 3 camadas conectadas. Esta arquitetura é usada muitas das vezes nas aplicacoes web onde a View é responsavel pela representacao dos dados ao utlizador, o controller serve para tratar dos dados na entrada ou na saida, e por fim o Model que o sustento do software é a parte da modelacao dos dados.

1. Métodos arquitetónicos

Foi utilizado um ficheiro de configuração para atribuir valores estáticos como por exemplo definir o valor da nota minima e maxima na avaliação de uma sandes.

Foram criados classes utils para ajudar no desenvolvimento, como por exemplo a classe ObjectMapper que serve para a transformação de um objeto em outro tipo de objeto. A classe DateValidation serve para a verificação a data e hora.

Decidiu-se usar o padrão Builder na construção dos objetos, assim mantendo uma maior integridade dos dados.

1. Arvore de utilidade

|  |  |  |
| --- | --- | --- |
| Desempenho | Latência | Nos metodos de busca na BD não há ciclos de For nem while, assim favorecendo o tempo de resposta por parte da BD |
| Taxa de transferencia |
| Modificabilidade | Organização do codigo | As classes têm uma estrutura padrao, assim ajudando na manutenção do mesmo. |
| Manutenção |
| Disponibilidade | Falha de hardware | - |
| Falha de serviços |
| Segurança | Confidencialidade dos dados | Não são disponibilizados dados que podem compromoter a seguranca dos dados, como por exemplo o id dos objetos. |
| Integridade dos dados | Usado o padrão Builder para garantir que todos os dados criados estão certos. |