

# DOMAIN DRIVEN DESIGN

## Objetivos:

Representar todo o conhecimento absorvido e colocar em pratica todo o aprendizado nos dois semestres da disciplina de *Domain Driven Design*, desenvolver um projeto que esteja de acordo com o tema proposto, deve conter a documentação e diagramas do projeto, a entrega deve ser feita na linguagem de programação Java, seguindo as boas práticas e entregando os tópicos abaixo:

## Documentação:

- (0,50 ponto) – Capa contendo o nome do projeto, RM dos integrantes e nome completo, na segunda página um índice (sumário) com os tópicos de acordo com as informações sobre o projeto, terceira e quarta página a explicação e a justificativa do porquê do projeto.
- (0,50 ponto) - Conter o link do Github do repositório do projeto.
- (10 pontos) – Diagrama de classes que reflita o código Java, ou seja, todas as classes que estão no código, tem que estar nesse diagrama, com no mínimo 10 classes, sendo que os prints devem estar no .doc que irá entregar.
- (0,50 ponto) Print do protótipo das telas envolvidas com o *front-end* e print das tabelas de banco de dados do projeto.

**OBS: A DOCUMENTAÇÃO DEVE SER ENTREGUE EM UM ÚNICO DOCUMENTO, NO FORMATO PDF, SERÁ AVALIADA A QUALIDADE E A ORGANIZAÇÃO, ESTAR EM BOA RESOLUÇÃO E LEGÍVEL. CASO CONTRÁRIO HAVERÁ DESCONTO DA PONTUAÇÃO DE ACORDO COM OS TÓPICOS.**

# DOMAIN DRIVEN DESIGN

## Projeto Java:

- (10 pontos) Tratamento de exceções, contendo um pacote para as classes de exceções.
- (10 pontos) Camada de conexão com o Banco de Dados, deixando o usuário e senha na classe de conexão e CRUD com os métodos Inserir, Deletar, Alterar e Selecionar os dados;
- (30 pontos) Camada de Integração do *Back-End* com o *Front-End*, mínimo de quatro páginas no *Front-end* que deve estar de acordo com o *Back-End*, este por sua vez deve estar alinhado e conectado funcionando com o Banco de Dados;
- (10 pontos) Camada BO contendo os métodos intermediários da API.
- (0,50 ponto) Implementação de padrões de projeto vistos em aula.
- (10 pontos) No mínimo duas APIs externas, por exemplo uma com o viacep e outra a sua escolha, mas se quiser, podem ser duas APIs externas que você escolher.

# DOMAIN DRIVEN DESIGN

## Entrega:

- O líder irá criar uma pasta e incluir as informações abaixo:
  - O .doc que tem a parte escrita que foi solicitada e os prints dos diagramas de classe, pagina do front-end e a modelagem relacional do banco de dados.
  - O projeto Java que você irá exportar da IDE (Eclipse ou IntelliJ) e o usuário e senha terão que estar na entrega, disponíveis na classe de conexão, para que o professor consiga testar o projeto.
  - Um vídeo de 2 min explicando o seu desenvolvimento e mostrando a integração na prática do back com o front

**OBS: A ENTREGA DEVERÁ SER FEITA EM UMA PASTA COMPACTADA (.ZIP) CONTENDO O PROJETO JAVA E A DOCUMENTAÇÃO NO FORMATO PDF através do Portal da FIAP.**

**Instruções:** Para efetuar essa entrega você terá que pensar que o Java – *back-end* tem que estar pronto para conversar com o *front-end* e o banco de dados, para que esteja integrado, então vamos lá, essa matéria também está totalmente ligada a matéria de Software Design & Total Experience, pois as classes apresentadas no Diagrama de Classes e até mesmo no projeto Java tem que refletir a ideia do seu projeto e não se esqueça que também está integrado com Python, pois se receber os dados eles irão ficar armazenados, então tem a oportunidade de processamentos e ou análises com Python, então siga as instruções abaixo para fazer a entrega: