

## **1. Objetivo**

Capacitar o aluno a desenvolver um projeto, simulando a experiência profissional, utilizando técnicas, ferramentas, metodologias e boas práticas trabalhadas ao longo do curso de tecnologia.

## **2. Normas Básicas**

1. Números de integrantes por grupo: até **6 alunos**
2. Não será permitido o desenvolvimento individual do projeto.
3. Os grupos têm obrigação de aceitar novos componentes.
4. Os componentes dos grupos, através do líder, poderão indicar a não atribuição da nota aos componentes do grupo que não participaram das atividades relativas ao projeto.
5. Os grupos devem idealizar os projetos a serem desenvolvidos ao longo do ano letivo, dividido em entregas parciais durante o 1º. e 2º. semestre.
6. As entregas serão parciais e divididas entre as disciplinas trabalhadas ao longo do ano. As solicitações são individuais, ou seja, feitas por cada disciplina.
7. Será realizada uma entrega ao final de cada sprint, a entrega deve conter todos os pedidos de todas as disciplinas.

## **3. Responsabilidades**

### **Responsabilidades dos alunos**

- Ter ciência do documento da Challenge e cumprir as entregas conforme solicitação dos professores.
- Todos os componentes do grupo devem estar envolvidos em todas as disciplinas do projeto. Fica proibido e irá onerar a nota individual do aluno que desenvolver o conteúdo somente de uma das disciplinas, todos os alunos devem gerar artefatos de todas as disciplinas.
- Atentar-se a qualidade no desenvolvimento dos entregáveis.

### ***Responsabilidades dos Professores do Curso***

#### **Responsabilidades:**

- Responsáveis por orientar o desenvolvimento do projeto em suas disciplinas.
- A correção de cada entregável, é de responsabilidade do professor da disciplina em cada turma.

- Disponibilizar nota individual referente à disciplina.
- Manter informações no grupo de professores relativas a cada grupo (compartilhar avaliação dos grupos)
- Manter sua solicitação de entregável sempre atualizada para o grupo de professores e alunos.
- Explicar detalhadamente entregável da disciplina e critérios de avaliação junto aos alunos
- Disponibilizar justificativa de nota aplicada ao grupo/aluno, quando se aplicar.

#### **Relação de professores coordenadores da Challenge:**

- 1 TDS A, F, G, H, J, R, S, T e U

**Professores Allen Fernando e Renato Parducci**, coordenação referente a gestão, planejamento e acompanhamento dos projetos.

**Professores Andrey Masiero e Fernanda Caetano**, scrum máster, comunicação e orientação de entregas junto à B2W.

**Demais Professores do curso**, mentoria e acompanhamento técnico dos projetos.

#### **4. Exigências a serem cumpridas em cada disciplina**

Nesse momento serão realizadas duas entregas de cada disciplina que irão compor a entrega final da challenge.

##### **I. AI & Chatbot**

Primeira entrega:

Motivação dos serviços/técnicas escolhidos(as).

Em qual funcionalidade será aplicado e como será a aplicação.

Base de treinamento ou conjunto de variáveis que serão considerados para o modelo criado.

Segunda entrega:

O algoritmo/serviço funcionando e integrado a solução  
Relatório com a análise dos resultados

## II. Agile Software Design

Primeira entrega:

Diagramas e Descrições dos Casos de Uso do sistema projetado. (**Serão solicitados dois cenários, onde deverá ser escolhido um dos cenários para implementação com as demais disciplinas**)

Critérios de avaliação (10 pontos na 1a entrega, com peso 1 em cada item a seguir):

- a. Identificação e diagramação correta de Atores
- b. Identificação e diagramação correta de Casos de Uso
- c. Associação entre Atores (generalização/especialização)
- d. Associação entre os Casos de Uso (includes, extends, generalização)
- e. Associação entre Atores e Casos de Uso
- f. Descrição correta de Resumo de UC (Use Case/Caso de Uso)
- g. Descrição correta de Atores e UCs relacionados
- h. Descrição correta de Pré-condição e Pós-condição
- i. Descrição correta de Fluxo principal de UCs
- j. Descrição correta de Fluxos Alternativos e/ou Exceção

Segunda Entrega:

Primeira parte: Diagrama de Classes de Objetos

Critérios de avaliação (8 pontos , com peso 1 em cada item a seguir):

- a. Identificação das Classes de objetos
- b. Atribuição do estereótipo de Classes
- c. Associação entre as Classes (generalização/especialização, composição, agregação)
- d. Identificação de cardinalidades nas associações de classes
- e. Documentação de Atributos de Classes com tipificação
- f. Documentação de Métodos de Classes com detalhamento de parâmetros do contrato do método
- g. Marcação de visibilidade de atributos
- h. Marcação de visibilidade de métodos

Segunda parte: link para vídeo que demonstra o uso/operação do software para executar um cenário de uso "core" (representativo) dos objetivos do projeto (peso 2)

**Obs.: Os vídeos não devem exceder 5 minutos.**

### III. Computational Thinking using Python

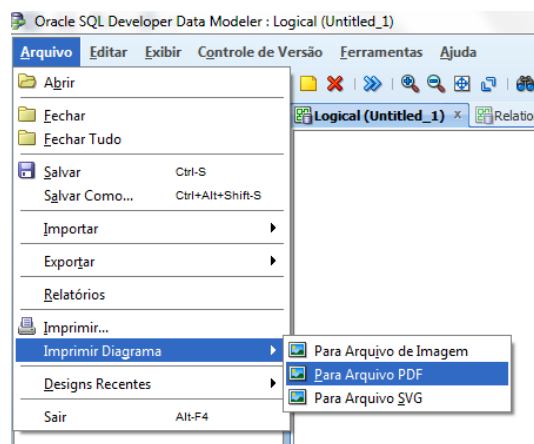
Junto com Domain Driven Design.

### IV. Database Modeling - SQL

Primeira entrega:

#### Modelo Físico de Dados ou Modelo Relacional atualizado

1. Criar o modelo físico de dados a partir do modelo lógico de dados.
2. Utilizar a ferramenta SQL Developer Data Modeler para construir o MER relacional.
3. Manter legenda contendo o nome dos componentes do grupo e respectivos RMs.
4. Gerar arquivo PDF referente ao MER relacional desenvolvido.



#### 5. Salvar como:

1TDS<X>\_2020\_<NomeGrupo>\_MODELO\_RELACIONAL.PDF

Onde: X = turma onde o grupo está inserido.

Segunda Entrega:

#### SCRIPT DDL PARA IMPLEMENTAÇÃO DO BD – VALOR: 5,0 PONTOS

1. SCRIPT DDL referente à implementação da estrutura do banco de dados.

2. Gerar o código através da ferramenta SQL Developer Data Modeler.
3. Arquivo no formato SQL.
4. Inserir no início do arquivo, em forma de comentário, o nome e RM de cada componente do grupo **(OBRIGATÓRIO)**.
5. **Nome do arquivo a ser entregue:**  
**1TDS<X>\_2020\_<NomeGrupo>\_SCRIPT\_DDL.SQL**  
**Onde: X = turma onde o grupo está inserido**

**SCRIPT DML PARA POPULAR AS TABELAS PARA OS TESTES DA APLICAÇÃO – Valor: 5,0 pontos**

1. SCRIPT DML referente a massa de dados para testes.
2. Arquivo no formato SQL.
3. **Nome do arquivo a ser entregue:**  
**1TDS<X>\_2020\_<NomeGrupo>\_SCRIPT\_DML.SQL**  
**Onde: X = turma onde o grupo está inserido.**
4. Indicações para criação da massa de dados online (podem ser utilizados outros similares):  
<http://www.generatedata.com>  
<http://mockaroo.com>  
<http://freedatagenerator.com/sql-data-generator>  
<http://databasetestdata.com>

**4. A massa de dados, deve ser composta por dados válidos, ou seja, não devem ser inseridos: xxxx,11111, teste, ou similar. Trabalhar com dados fictícios e válidos.**

**V. Domain Driven Design**

Primeira entrega:

A Entrega 1, será composta pelos seguintes entregáveis:

- Documento contendo (4,0 pts) :
  - capa com o nome dos integrantes, nome da equipe e nome da solução (0,5 pts)
  - Índice **(0,5 pts)**
  - Capítulo 1 – Objetivos e justificativa da solução **(1 pt)**

- Capítulo 2 – Requisito 1 (input) descritivo, apontamento das tabelas e classes envolvidas e print do protótipo da(s) tela(s) envolvidas com esta funcionalidade. (1 pt)
- Capítulo 3 – Requisito 2 (output) descritivo, apontamento das tabelas e classes envolvidas e print do protótipo da(s) tela(s) envolvidas com esta funcionalidade. (1 pt)
- Projeto (6,0 pts) exportado como Archive File (zip) pelo Eclipse, contendo:
  - Camada de conexão, já deve estar “setada” com um usuário (um dos integrantes do grupo) com a conta do banco Oracle da FIAP e obviamente com a devida senha preenchida. (0,5 pts)
  - Camada Java Beans/Domain (1,5 pts), com os devidos padrões e regras trabalhados em sala de aula.
  - DAO => com um CRUD (relacionado ao input e/ou output citado no documento) (2,0 pts – o grau de dificuldade será levado em consideração para a correção, além da utilização dos padrões e regras utilizados e aplicados em sala de aula)
  - BO/Validation => sobre o CRUD que foi desenvolvido. (2,0 pts – o grau de dificuldade será levado em consideração para a correção, além da utilização dos padrões e regras utilizados e aplicados em sala de aula).

#### Segunda Entrega:

- Projeto (10,0 pts) exportado como Archive File (zip) pelo Eclipse, contendo:
  - Camada de conexão, já deve estar “setada” com um usuário (um dos integrantes do grupo) com a conta do banco Oracle da FIAP e obviamente com a devida senha preenchida. (0,5 pts)
  - Camada Java Beans/Domain (1,5 pts), com os devidos padrões e regras trabalhados em sala de aula.
  - DAO e BO/Validation para a funcionalidade de INPUT (2,0 pts), – o grau de dificuldade será levado em consideração para a correção, além da utilização dos padrões e regras utilizados e aplicados em sala de aula.
  - DAO e BO/Validation para a funcionalidade de OUTPUT (3,0 pts), – o grau de dificuldade será levado em consideração para a correção, além da utilização dos padrões e regras utilizados e aplicados em sala de aula.
  - JAVADOC (1,5 pts) para as classes DAO e BO entregues (anotações na classe – não é preciso enviar o JAVADOC gerado), ATENÇÃO este critério será avaliado como “executado” ou “não executado”, ou seja, “1,5 pts” ou “0 pts”, por isso ele deve estar COMPLETAMENTE feito.
  - EXCEPTIONS (0,5 pts) para o devido tratamento das exceções que podem ser disparadas no projeto.
  - Entrega (1,0 pt) para que as regras abaixo, sejam devidamente seguidas:

- Desenvolver o seu projeto em uma nova workspace. Este projeto deverá ter o nome do grupo;
- Dentro deste projeto deverão criar três pastas: lib, funcionalidades e banco.
- Dentro da pasta lib, os arquivos utilizados para conexão com banco, via JDBC.
- Dentro da pasta banco, um arquivo chamado banco.txt, que deverá conter em ordem correta e lógica, os comandos DDL e DML para criação do banco e população das tabelas.
- Dentro da pasta funcionalidades, deverão criar um arquivo chamado funcionalidades.txt, onde deverá estar descrito as duas funcionalidades implementadas.
- Todo o projeto deverá ser realizado na IDE Eclipse/IntelliJ.
- Não serão concedidos prazos adicionais para análise e correções de possíveis problemas na execução, após a entrega ao professor.

#### **VI. Responsive Web Development**

Deverão desenvolver um sistema web que deverá contemplar:

Primeira Entrega:

- Telas/interfaces para atender a solução proposta à empresa parceira;

Segunda Entrega:

- CRUD os usuários do sistema persistirem e manipularem as informações;
- Sistema de login para usuários e administrador, diferenciando os níveis de acesso;
- Se houver utilização de Chatbot na solução, integrar ao projeto;

#### **Pontos obrigatórios:**

O sistema deverá ser replicável em qualquer computador;

Todas as telas deverão ser responsivas;

Só serão aceitas as seguintes tecnologias:

HTML5;

CSS3;

JS;

Bootstrap;

jQuery;

JSP;

Servlet;

JSTL;

El;

Node-Red (Chatbot).

JSON (Chatbot)

## 5. Entregas

Todas as entregas serão realizadas via **Portal do Aluno**.

A **primeira entrega** deverá ser realizada até o **dia 20 de setembro de 2020**.

A **segunda entrega**:

- A semana de **27 a 30 de outubro** será dedicada ao trabalho dos grupos na challenge.
- As apresentações dos Pitches ocorrerão na semana de **03 a 06 de novembro**.
- Dia **02 de novembro** é o prazo final para entrega do projeto, de acordo com este documento.