

CHALLENGE 2024



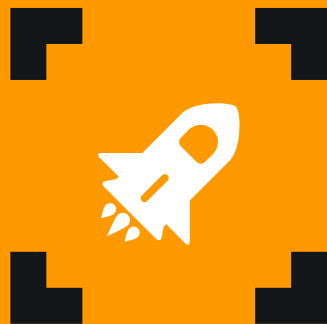
CHALLENGE 2024

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - 2º ANO

EMPRESA PARCEIRA

- **Plusoft** – www.plusoft.com.br





O Desafio

O DESAFIO

"Desvendando o Potencial dos Dados: Explorando a AI Generativa e o Deep Analytics para Impulsionar o Crescimento Empresarial"



O DESAFIO

Cada vez mais as empresas estão em busca de **entender como os dados** gerados nas interações de encantamento, compra e atendimento podem ajudar na condução do negócio. Leads, prospects e clientes geram dados que podem ser transformados em insights e ações. Análise, predição e prescrição são elementos fundamentais nessa jornada. **Como a AI generativa e o deep analytics podem ajudar nesse processo?** Empresas que entregarem essa proposta de valor podem crescer muito rápido em um mercado avido por informações.

ALGUNS EXEMPLOS



Modelo de Previsão de Comportamento do Cliente: Desenvolvimento de um modelo preditivo que utiliza técnicas de machine learning para prever o comportamento futuro dos clientes com base em seus dados históricos de interação, permitindo que a empresa antecipe suas necessidades e personalize suas estratégias de engajamento.

Sistema de Recomendação Personalizada: Construção de um sistema de recomendação inteligente que utiliza algoritmos de AI generativa para analisar os padrões de comportamento dos clientes e sugerir produtos ou serviços relevantes de acordo com seus interesses e preferências individuais.

Análise de Sentimento e Feedback Automatizado: Implementação de uma solução de análise de sentimento e feedback automatizado que utiliza técnicas avançadas de processamento de linguagem natural (NLP) para extrair insights úteis a partir das interações dos clientes em diferentes canais de comunicação, como e-mails, mídias sociais e chats online.

Otimização de Campanhas de Marketing: Desenvolvimento de um sistema de otimização de campanhas de marketing que utiliza análises profundas de dados para identificar os canais e estratégias mais eficazes para atrair e converter leads em clientes, maximizando o retorno sobre o investimento (ROI) em marketing.

Assistente Virtual Inteligente para Atendimento ao Cliente: Criação de um assistente virtual baseado em AI generativa que seja capaz de fornecer suporte e assistência personalizada aos clientes, respondendo às suas perguntas, resolvendo problemas comuns e encaminhando consultas mais complexas para os canais de atendimento humano quando necessário.



Normas Básicas

GRUPOS

- ▣ Máximo de 5 **integrantes**, sem exceções;
- ▣ **Não é permitido** desenvolver o projeto individualmente ou em grupos com 6 integrantes
- ▣ É **recomendado que os grupos sejam compostos preferencialmente por alunos da mesma turma**. No entanto, é permitido formar grupos com alunos de turmas diferentes, desde que estejam **cientes e aceitem os possíveis inconvenientes**, como horários de apresentação distintos (manhã/noite), diferentes professores, entre outros.
- ▣ **Não** é permitido formar grupos com **alunos de outros cursos ou anos** (TDS ON é outro curso).

ENTREGAS

- ▣ Os grupos devem **idealizar os projetos** a serem desenvolvidos ao longo do ano letivo, dividido em **entregas parciais durante o 1º. e 2º. semestre;**
- ▣ Haverá um total de 4 sprints (4 entregas ao longo do ano), sendo duas no primeiro semestre e duas no segundo semestre;
- ▣ **A entrega** ocorrerá ao final de cada Sprint e será para **todas as disciplinas;**
- ▣ **Cada sprint** terá os seus entregáveis de **cada disciplina**, especificados pelos respectivos professores no **portal e neste documento;**

ENTREGAS

- Entrega de cada **sprint** será realizada **por disciplina**;
- Cada disciplina disponibilizou os **requisitos na documentação**;
- Entrega de todas as disciplinas serão realizadas através do **portal FIAP**.

1º Entrega
15/04/2024

2º Entrega
13/05/2024



Responsabilidades

ALUNOS

- ▣ Ter **ciência do documento da Challenge** e cumprir as entregas conforme solicitação dos professores.
- ▣ **Organização e gerenciamento do grupo**, como o planejamento e prazos das entregas internas.
- ▣ Atentar-se a qualidade no **desenvolvimento dos entregáveis**.

PROFESSORES

- **Responsáveis por orientar o desenvolvimento** do projeto em suas disciplinas;
- 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;
- **A correção de cada entregável**, é de responsabilidade do professor da disciplina em cada turma;
- Disponibilizar **nota individual** referente à disciplina;
- Disponibilizar **justificativa de nota aplicada** ao grupo/aluno, quando se aplicar;

SCRUM MASTER

- ▣ **Comunicação entre os alunos/professores e a(s) empresa(s) parceira(s);**
- ▣ **Manter os dados dos grupos de alunos atualizada** (a gestão e formação dos grupos é de responsabilidade dos alunos);
- ▣ Manter o **documento da challenge** atualizada;
- ▣ **Agendar os eventos** com a empresa parceira/professores, como apresentações, reuniões e treinamentos;

Scrum Master 2024

Professores: Karina Costa, Luis Carlos Silva, Thiago Keller e Thiago Yamamoto.



Premiação

PREMIAÇÃO

- Premiação em **dinheiro**;
- **Medalha, shape** e **camiseta** exclusivas;



Imagens ilustrativas



Cronograma

CRONOGRAMA – 1º Semestre



DATA	EVENTO	STEAKHOLDER
19/03/2024	Abertura do Challenge com a Plusoft	PLUSOFT
A definir	Primeira mentoria (online)	PLUSOFT
Até 15/04	ENTREGA DA SPRINT 1	ALUNO
Até 23/04	Feedback das entregas SPRINT 1	PROFESSORES
A definir	Segunda mentoria presencial (FIAP)	PLUSOFT
Até 13/05	ENTREGA DA SPRINT 2	ALUNO
Até 21/05	Feedback das entregas SPRINT 2	PROFESSORES

MENTORIAS

- ❑ **Caso o grupo não participe de uma mentoria**, o grupo será penalizado em 10 pontos em todas as disciplinas (pontuação de 0 a 100);
- ❑ Neste semestre haverá **duas sessões de mentoria**. A não participação do grupo em ambas resultará em um **desconto de 20 pontos** em cada disciplina. Caso o grupo falte a apenas uma, o **desconto será de 10 pontos**.
- ❑ **Não é obrigatória a participação de todos os integrantes** do grupo para a mentoria.
- ❑ **O desconto será para todos do grupo**, não é individual, ou seja, se pelo menos um integrante participar da mentoria, o grupo não terá a nota descontada.
- ❑ Caso a apresentação para a empresa **esteja significativamente abaixo das expectativas**, indicando falta de preparação adequada por parte do grupo, poderá haver um desconto na nota



1º Entregas

COMPLIANCE & QUALITY ASSURANCE



Desenvolver a declaração de visão e escopo do projeto em formato PITCH, contendo:

- ☐ Descrição do problema a resolver (peso 20%)
- ☐ Descrição dos objetivos da solução idealizada (peso 20%)
- ☐ Definição do público alvo que comprará (cliente/pagante) e/ou usará a solução (consumidor) (peso 20%)
- ☐ Estudo de produtos semelhantes já existentes no mercado (que solucionam o mesmo problema, mesmo que de forma diferente, parcial ou totalmente) (peso 20%)
- ☐ Avaliação do potencial de mercado (fatia de mercado a conquistar) (peso 20%)

Gere um PDF com o material desenvolvido e suba a sua resposta da challenge sprint - entregas fora do formato perderão 10%

MASTERING RELATIONAL AND NON-RELATIONAL DATABASE

Caso o grupo opte **por um banco de dados relacional** a entrega será:

- ❑ Entregar o Projeto de Banco de Dados Relacional, compondo **(50 pontos)**:
 - Os Diagramas que deverão ser construídos no Oracle Data Modeler, cumprindo a notação DER-(Logical Model) e o MER-(Physical Model).
 - Deverá ser utilizado obrigatoriamente a notação de Barker para o DER-(Logical Model) e no mínimo está na 3ª Forma Normal (3FN)
- ❑ Gerar o Modelo Físico e criar os objetos/esquema no banco de dados. **(10 pontos)**
- ❑ Para cada Tabela preencher no mínimo 5 registros de acordo com a especificação do projeto. **(20 pontos)**
- ❑ Criar dois blocos anônimos para mostrar os dados inseridos, com pelo menos 3 consultas de junções (Joins) utilizando agrupamento (group by) e ordenação (order by). **(20 pontos)**
- ❑ **Entregável:** arquivo zipado contendo o pdf. Projeto de Banco de Dados Relacional e arquivo .sql com os scripts separados por tabela e pdf do modelo logico e físico.
- ❑ **Observação:** A boa organização do conteúdo dos arquivos é de suma importância para apontamento da nota, caso haja falha nesta organização a correção pode vir a ser comprometida acarretando diminuição da nota.

MASTERING RELATIONAL AND NON-RELATIONAL DATABASE

Caso o grupo opte **por um banco de dados não relacional** a entrega será:

- ❑ Entregar o Projeto de Banco de Dados Não Relacional, compondo (até **40 pontos**):
 - ❑ Documento com a descrição do projeto e a justificativa em utilizar um Banco de Dados NoSQL.
 - ❑ Qual tipo de banco de dados será utilizado e porquê ele se encaixaria no seu projeto.
- ❑ Construção de 5 arquivos de dados de acordo com o tipo de banco de dados escolhido (até **60 pontos**)
 - ❑ Caso seja MongoDB: Realizar 5 cinco documentos em .JSON (ou BSON), com no mínimo 10 atributos preenchidos.
 - ❑ Caso seja Redis: Realizar 5 chaves com valores de exemplos preenchidos.
- ❑ **Entregável:** arquivo zipado contendo o **pdf**. Projeto de Banco de Dados e os arquivos de *script* referente ao banco de dados escolhido
- ❑ **Observação:** A boa organização do conteúdo dos arquivos é de suma importância para apontamento da nota, caso haja falha nesta organização a correção pode vir a ser comprometida acarretando diminuição da nota.

DEVOPS TOOLS E CLOUD COMPUTING



A partir da solução identificada pelo seu grupo, justifique como a técnica de Virtualização pode contribuir na entrega desse importante projeto. (até 100 pontos)

O que será avaliado:

01) A organização e à estrutura do material gerado. (até 90 pontos)

Se a dissertação demonstra os conhecimentos adquiridos em aula e se o texto é coerente e organizado.
Se contém exemplos e aderência para a solução apresentada pelo grupo.

02) Se existem imagens explicativas (até 10 pontos)

Se a dissertação é acompanhada com imagens para ilustrar e melhorar o entendimento da dissertação e solução apresentadas.

A entrega deve ser em um arquivo PDF e entregue via Portal do Aluno, na área especificada para as entregas do Challenge, **sua pontuação irá variar de acordo com a qualidade da entrega.**

DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

Objetivo Principal:

- ❑ Criação de vídeo *pitch* (máximo: 3 minutos) com a apresentação macro da ideia do projeto.

Objetivos Específicos:

- ❑ Apresentar o tema escolhido pelo grupo.
- ❑ Apresentar a explicação do problema.
- ❑ Apresentar as alternativas de solução.
- ❑ Descrever os possíveis frameworks/bibliotecas Python que serão utilizados (seja específico, quais e por quais razões)
- ❑ Descrever de que forma os conceitos de Machine Learning / IA poderão ser utilizados no desenvolvimento do projeto (seja específico neste requisito).

DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

Critérios de Avaliação:

- ❑ Organização e a estrutura do material gerado (até 30 pontos).
- ❑ Apresentação do vídeo de forma clara (até 30 pontos).
- ❑ Abrangência, profundidade e coerência para a aplicação de conceitos de Machine Learning / IA com apresentação das possíveis ferramentas / frameworks a serem aplicados na solução (até 40 pontos)

Entregável:

- ❑ Um arquivo .zip com o link do vídeo carregado no Youtube, conforme objetivos acima.

Condições de entrega:

- ❑ A integridade e o conteúdo do arquivo entregue são de responsabilidade dos integrantes do grupo. Arquivos entregues sem conteúdo ou com arquivos corrompidos não serão considerados.
- ❑ Não serão aceitos arquivos enviados pelo Teams ou fora do prazo.

JAVA ADVANCED



- ❑ Produza pelo menos um dos domínios da sua solução tecnológica proposta utilizando Java e Spring. Esta solução deverá ser capaz de persistir, manter e consultar os dados em um SGBD Relacional (Preferencialmente Oracle). Os dados gerados por meio da solução de TI proposta pela equipe será primordial para a construção da **IA generativa** e do **deep analytics** que foram solicitados pelo cliente para a tomada de decisões estratégicas (baseadas em análise de dados e modelos preditivos).

JAVA ADVANCED



Portanto, você deverá:

Criar aplicação Java com Spring Boot que ajude a resolver um problema (utilize a criatividade). A aplicação deverá respeitar os fundamentos da Programação Orientada a Objetos e as classes de entidade deverão estar relacionadas e mapeadas com JPA e deverá ainda:

- ❑ Conter código que respeite a coesão e o desacoplamento;
- ❑ Respeitar os padrões DTO e utilizar a Bean Validation;
- ❑ Respeitar os conceitos fundamentais do REST (RESTful);
- ❑ Na primeira Sprint, a API deverá estar de acordo com o modelo de maturidade **nível 1**;
- ❑ Utilizar os Design Patterns como Singleton, Repository com Genéricos quando necessário;
- ❑ Poderá utilizar a JPQL e/ou Spring JPA Query Methods ;
- ❑ A Gestão de Configuração dos artefatos de software deverá ser demonstrada pela equipe, portanto todos os artefatos produzidos deverão estar no Github e os professores deverão ter acesso ao repositório.

JAVA ADVANCED



- ❑ Apresentar os testes dos endpoints da API (Utilize o Postman ou Insomnia) exporte as requisições para que o professor possa validar o funcionamento.

Como você será pontuado:

- ❑ a) **(até 5 Pontos)** Apresentar o cronograma de desenvolvimento e respeitar os prazos. Crie um o documento dizendo quem fará o que e quando a atividade deverá ser ou foi realizada;
- ❑ b) **(até 10 Pontos)** Imagens explicativas da arquitetura, definição das classes de domínio da aplicação e o respectivo Diagrama de Classes de Entidade. O **Diagrama de Entidade e Relacionamento** (DER) e o **Diagrama de Classe das Entidades** deverão ser coerentes. Se necessário faça uma breve explicação sobre os relacionamentos e as *constraints* envolvidas;
- ❑ c) **(até 40 Pontos)** Implementação das classes de Entidade necessárias para solução do problema que se propõe resolver. Atenção ao correto encapsulamento, à tipagem dos atributos e ao Mapeamento Objeto Relacional com JPA e Hibernate (A equipe deverá demonstrar a cada Sprint que houve evolução em relação à sprint anterior);

JAVA ADVANCED

- ❑ d) **(até 15 Pontos)** A aplicação respeita os conceitos fundamentais do REST criados por Roy Fielding (RESTful) e API está de acordo com o modelo de maturidade nível 1 proposto por Leonard Richardson.
- ❑ e) **(até 10 Pontos)** A Gestão de Configuração dos Artefatos de Software foram demonstrados pela equipe. Ou seja, todos os artefatos produzidos estão no **Github** e os professores têm acesso ao repositório.
- ❑ f) **(até 10 pontos)** Envio do link para o projeto público compartilhado no github contendo entre outros:
 - 1) Toda documentação;
 - 2) Código-fonte;
 - 3) Arquivo README.md com texto contendo:
 - 3.1) Nome da aplicação;

JAVA ADVANCED



- ❑ 3.2) Nome completo e breve apresentação dos integrantes do Grupo (Atividade da qual ficou responsável no projeto);
 - 3.3) instrução de como rodar a aplicação;
 - 3.4) imagem dos diagramas;
 - 3.5) link para vídeo apresentando a Proposta Tecnológica, o público-alvo da aplicação e os problemas que a aplicação se propõe a solucionar;
 - 3.6) Listagem de todos os *endpoints* (Documentação da API);
- ❑ g) **(até 10 pontos)** Se a equipe demonstrou preocupação em testar a aplicação e provou com documentos. Disponibilizou arquivos para que o professor seja capaz de realizar teste dos *endpoints* e houve, após os testes, a perfeita persistência e recuperação dos dados. (exporte do Postman ou do Insomnia as requisições aos *endpoints*). Inclua na pasta documentos do projeto.

MOBILE APP DEVELOPMENT



- ❑ Descrição:
 - Protótipo não funcional wireframe de uma solução mobile com no mínimo 5 telas, acompanhado de explicações detalhadas para cada tela. **(70 pontos)**
 - Incluir cenários possíveis de sucesso ou erro. **(30 pontos)**
- ❑ Formato de Entrega: Documento do Word com imagens e explicações, contendo o nome completo e RM de todos os integrantes do grupo na capa.
- ❑ Dicas de ferramentas: Software de prototipação como Figma, Protopie, Photoshop, Miro e etc;

ADVANCED BUSINESS DEVELOPMENT WITH .NET



Na primeira entrega, estaremos concentrados em desenvolver e aprofundar a ideia central. Por sua vez, a segunda entrega focará na execução prática do produto, seguindo o roteiro estabelecido inicialmente. Assim, a primeira fase engloba a formulação da ideia, acompanhada de estudos e pesquisas necessários para conceber a solução ideal.

▣ **Definição do Problema(até 30 pontos)**

- ▣ É essencial destacar e esclarecer o problema que pretendemos resolver, garantindo que a questão central seja compreendida de maneira inequívoca.

▣ **Público Alvo (até 20 pontos)**

- ▣ Determinar quem são as pessoas afetadas pelo problema em questão, delimitando nosso público-alvo e potenciais clientes.

ADVANCED BUSINESS DEVELOPMENT WITH .NET

■ Proposta de Solução (até 30 pontos)

- Clarificar como nossa solução gera valor para o público-alvo, detalhando o funcionamento e os benefícios que oferece.

■ Análise da Concorrência (até 10 pontos)

- Mapear os principais concorrentes, tanto diretos quanto indiretos, para entender o cenário competitivo.

■ Modelos de receita (até 10 pontos)

- Especificar como o projeto gerará receita, apresentando as diversas fontes de renda previstas.

Apresentação pode ser um pdf/ppt ou outro arquivo similar. Aproveite para estudar a ideia e construir uma versão inicial do material que será apresentado para a banca da plusoft.



2º Entregas

COMPLIANCE & QUALITY ASSURANCE



Faça o desenho da Arquitetura da solução, aplicando TOGAF e usando a ferramenta ARCHIMATE (ARCHI).

- ☐ Desenhe em uma mesma página de diagrama no Archi
- ☐ Visão da arquitetura (peso 25%)
- ☐ Arquitetura de negócio (peso 25%)
- ☐ Arquitetura de sistema (peso 25%)
- ☐ Arquitetura de tecnologia (peso 25%)
- ☐ Gere um documento PDF com a imagem do seu desenho e suba a sua resposta da challenge sprint - entregas fora do formato perderão 10%

MASTERING RELATIONAL AND NON-RELATIONAL DATABASE

Caso os grupos opte por **um banco de dados relacional** as entregas são:

- ❑ Aplicar as correções apontados pelo professor ao modelo apresentando na entrega anterior. **(10 pontos)**
- ❑ Criar duas funções para validar entrada de dados **(10 pontos)**
- ❑ Criar procedures de INSERT/UPDATE e DELETE para as tabelas criadas anteriormente **(20 pontos).**
- ❑ Utilizando a extensão **cx_oracle** do **Python** realizar pelo menos 5 chamadas nas procedures criadas para inserção dos dados no modelo proposto na primeira entrega **(20 pontos).**
- ❑ Criar uma procedure utilizando um curso usar Join com pelo menos 2 tabela do projeto. **(20 pontos)**
- ❑ Criar uma procedure que imprima um relatório com pelo menos um regra de negócio, que contenha funções, inner Join, order by, sum ou count. **(20 pontos)**
- ❑ **Entregáveis:** arquivo zipado com documento de descrição do trabalho, prints de execução dos scripts e scripts de criação dos objetos.

MASTERING RELATIONAL AND NON-RELATIONAL DATABASE



Caso os grupos optem por **um banco de dados NoSQL** as entregas são:

- ❑ Deverá ser construído uma rotina em PL/SQL que gere uma saída em JSON. Essa saída deverá alimentar um banco de dados NoSQL MongoDB, que contenha no mínimo 10 documentos. **(60)**
- ❑ Na aplicação, deverá ser construída uma interface que exiba os dados consultados no banco de dados NoSQL. **(20)**
- ❑ Exportar um dataset dos dados inseridos com a procedure. **(20)**
- ❑ **Entregavel:** arquivo zip contendo a procedure criada, print de execução dela em pdf, print de geração do dataset, prints da interface construída com os dados e o dataset gerado.
- ❑ **Observação:** A boa organização do conteúdo dos arquivos é de suma importância para apontamento da nota, caso haja falha nesta organização a correção pode vir a ser comprometida acarretando diminuição da nota.

DEVOPS TOOLS E CLOUD COMPUTING



01) De acordo com a arquitetura e tecnologia da sua solução proposta, qual é o melhor tipo de nuvem (Pública, Privada ou Híbrida) e tipos de serviços (SaaS, PaaS e IaaS) que se encaixam em sua solução? Justifique detalhadamente sua escolha. **(até 35 pontos)**

O que será avaliado:

A) A Organização, dissertação e estrutura do material gerado. **(até 25 pontos)**

Se a dissertação demonstra os conhecimentos adquiridos em aula e se o texto é coerente e organizado.
Se contém exemplos e aderência para a solução apresentada pelo grupo.

B) Se existem imagens explicativas. **(até 10 pontos)**

Se a dissertação é acompanhada com imagens para ilustrar e melhorar o entendimento da dissertação e solução apresentadas.

DEVOPS TOOLS E CLOUD COMPUTING



02) Definir e justificar os possíveis recursos de Cloud Computing que podem ser utilizados em sua solução. São esperados no mínimo três recursos que podem ser utilizados em Cloud. Nesse item o grupo pode utilizar qualquer recurso nos serviços prestados em nuvem: SaaS, PaaS e IaaS. **(até 35 pontos)**

O que será avaliado:

A) A Organização, dissertação e estrutura do material gerado. **(até 30 pontos)**

Se a dissertação demonstra os conhecimentos adquiridos em aula e se o texto é coerente e organizado.
Se contém exemplos e aderência para a solução apresentada pelo grupo.

B) Se existem imagens explicativas. **(até 05 pontos)**

Se a dissertação é acompanhada com imagens para ilustrar e melhorar o entendimento da dissertação e solução apresentadas.

DEVOPS TOOLS E CLOUD COMPUTING



03) Desenhe a arquitetura macro de sua solução com o uso dos recursos em nuvem **(até 30 pontos)**

O que será avaliado:

A) Na visualização do desenho da arquitetura apresentada, será avaliado: Fluxo, imagens dos recursos utilizados e a utilização de legendas ou rótulos. **(até 30 pontos)**

Ferramenta Sugerida para criar o desenho da Arquitetura: Visual Paradigm (apenas sugestão...)

Ferramenta de diagrama do Azure online

<https://online.visual-paradigm.com/pt/diagrams/features/azure-architecture-diagram-tool/>

A entrega deve ser em um arquivo PDF e entregue via Portal do Aluno, na área especificada para as entregas do Challenge, **sua pontuação irá variar de acordo com a qualidade da entrega.**

DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

Objetivo Principal:

- Apresentação de uma versão Beta (simples) envolvendo as principais ideias e os recursos que serão utilizados no projeto.

Objetivos Específicos:

- Apresentar as diferenças entre a etapa atual do projeto e a proposta apresentada na primeira entrega.
- Apresentar, com mais detalhes em relação à primeira entrega, os frameworks/bibliotecas Python, APIs e demais recursos/ferramentas que serão utilizados no desenvolvimento.
- Explicar, de forma breve, o funcionamento dos recursos/ferramentas (descritos acima) dentro da aplicação que será desenvolvida.
- Descrever resumidamente de que forma os conceitos de Machine Learning / IA estão sendo utilizados no desenvolvimento do projeto.

DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

Critérios de Avaliação:

- ❑ Organização e a estrutura do material gerado **(até 30 pontos)**.
- ❑ Apresentação do vídeo de forma clara **(até 30 pontos)**.
- ❑ Abrangência, profundidade e coerência para a aplicação de conceitos de Machine Learning / IA com apresentação das possíveis ferramentas / frameworks a serem aplicados na solução **(até 40 pontos)**.

Entregável:

- ❑ Um arquivo .zip com o link do repositório do projeto, contendo o link do novo vídeo e a documentação organizada, conforme descrito nos objetivos.

Condições de entrega:

- ❑ A integridade e o conteúdo do arquivo entregue são de responsabilidade dos integrantes do grupo. Arquivos entregues sem conteúdo ou com arquivos corrompidos não serão considerados.
- ❑ Não serão aceitos arquivos enviados pelo Teams ou fora do prazo.

JAVA ADVANCED



- ❑ A equipe deverá demonstrar evolução entre a primeira e a entrega que está sendo realizada nesta segunda Sprint.
- ❑ A equipe deverá portanto aperfeiçoar a solução tecnológica fazendo as devidas refatorações de código visando a evolução da qualidade da aplicação ou a inclusão de novos módulos ou domínios da solução tecnológica proposta utilizando Java e Spring.
- ❑ Esta solução deverá se manter capaz de persistir, alterar e consultar os dados em um SGBD Relacional (Preferencialmente Oracle). Os dados gerados por meio da solução de TI proposta pela equipe será primordial para a construção da **IA generativa** e do **deep analytics** que foram solicitados pelo cliente para a tomada de decisões estratégicas (baseadas em análise de dados e modelos preditivos).

JAVA ADVANCED



Portanto, você deverá:

Aperfeiçoar a aplicação Java com Spring Boot em desenvolvimento. A aplicação deverá respeitar os fundamentos da Programação Orientada a Objetos e as classes de entidade deverão estar relacionadas e mapeadas com JPA e deverá ainda:

- ❑ Conter código que respeite a coesão e o desacoplamento ;
- ❑ Respeitar os padrões DTO e utilizar a Bean Validation;
- ❑ Respeitar os conceitos fundamentais do REST (RESTful);
- ❑ A API deverá estar de acordo com o modelo de maturidade **nível 2** proposto por Leonard Richardson Utilizar os Design Patterns como Singleton, Repository com Genérics quando necessário;
- ❑ Poderá utilizar a JPQL e/ou Spring JPA Query Methods ;
- ❑ A Gestão de Configuração dos artefatos de software deverá ser demonstrada pela equipe, portanto todos os artefatos produzidos deverão estar no Github e os professores deverão ter acesso ao repositório.

JAVA ADVANCED



- ❑ Apresentar os testes dos endpoints da API (Utilize o Postman ou Insomnia) exporte as requisições para que o professor possa validar o funcionamento.
- ❑ Como você será pontuado:
- ❑ a) **(até 5 Pontos)** Apresentar o cronograma de desenvolvimento e respeitar os prazos. Crie um o documento dizendo quem fará o que e quando a atividade deverá ser ou foi realizada;
- ❑ b) **(até 10 Pontos)** Imagens explicativas da arquitetura, definição das classes de domínio da aplicação e o respectivo Diagrama de Classes de Entidade. O **Diagrama de Entidade e Relacionamento** (DER) e o **Diagrama de Classe das Entidades** deverão ser coerentes. Se necessário faça uma breve explicação sobre os relacionamentos e as *constraints* envolvidas;
- ❑ c) **(até 40 Pontos)** Implementação das classes de Entidade necessárias para solução do problema que se propõe a resolver. Atenção ao correto encapsulamento, à tipagem dos atributos e ao Mapeamento Objeto Relacional com JPA e Hibernate (A equipe deverá demonstrar a cada Sprint que houve evolução em relação à sprint anterior);

JAVA ADVANCED

- ❑ d) **(até 15 Pontos)** A aplicação respeita os conceitos fundamentais do REST criados por Roy Fielding (RESTful) e API está de acordo com o modelo de maturidade de proposto por Leonard Richardson
- ❑ e) **(até 10 Pontos)** A Gestão de Configuração dos Artefatos de Software foram demonstrados pela equipe. Ou seja, todos os artefatos produzidos estão no **Github** e os professores têm acesso ao repositório.
- ❑ f) **(até 10 pontos)** Envio do link para o projeto público compartilhado no github contendo entre outros:
 - 1) Toda documentação;
 - 2) Código-fonte;
 - 3) Arquivo README.md com texto contendo:
 - 3.1) Nome da aplicação;

JAVA ADVANCED



- ❑ 3.2) Nome completo e breve apresentação dos integrantes do Grupo (Atividade da qual ficou responsável no projeto);
 - 3.3) instrução de como rodar a aplicação;
 - 3.4) imagem dos diagramas;
 - 3.5) link para vídeo apresentando a Proposta Tecnológica, o público-alvo da aplicação e os problemas que a aplicação se propõe a solucionar;
 - 3.6) Listagem de todos os *endpoints* (Documentação da API);
- ❑ g) **(até 10 pontos)** Se a equipe demonstrou preocupação em testar a aplicação e provou com documentos. Disponibilizou arquivos para que o professor seja capaz de realizar teste dos *endpoints* e houve, após os testes, a perfeita persistência e recuperação dos dados. (exporte do Postman ou do Insomnia as requisições aos *endpoints*). Inclua na pasta documentos do projeto.
- ❑ **ATENÇÃO: A equipe deverá deixar BEM CLARO que houve evolução em comparação a entrega da Sprint anterior.**

MOBILE APP DEVELOPMENT

❑ **Descrição:**

- Diagrama demonstrando a integração/comunicação do aplicativo com API/banco de dados e sistemas externos. **(30 pontos)**
- Programação visual de 5 telas do aplicativo funcional. **(40 pontos)**
- Realizar chamadas de uma API (login, esqueceu senha e ao menos 3 telas fazendo requisições). **(30 pontos)**

- ❑ **Formato de Entrega:** Documento do Word com imagens, explicações de cada integração, e programação visual das telas, contendo o nome completo e RM de todos os integrantes do grupo na capa.

- ❑ **Ferramentas:** Activity e Views no Android, elementos do JetPack compose, ou Componentes do React Native

ADVANCED BUSINESS DEVELOPMENT WITH .NET



Atualização da primeira apresentação. O grupo pode atualizar toda apresentação, se for necessário. **Na apresentação, deverá ser acrescentada os seguintes tópicos:**

- ❑ **Arquitetura da Solução (até 20 pontos)**

- ❑ Visão macro dos sistemas e aplicativos, destacando como se integram através de tecnologias como API Rest, aplicativos móveis, IoT, chatbots, inteligência artificial, aplicativos web e bancos de dados. Este esquema facilitará o entendimento das funcionalidades e o fluxo de dados entre os componentes da solução.

- ❑ **Tecnologias Utilizadas (até 20 pontos)**

- ❑ Tecnologias selecionadas para o projeto como linguagens de programação e frameworks modernos, servidores, sistemas de banco de dados e APIs, etc.

ADVANCED BUSINESS DEVELOPMENT WITH .NET



- ❑ **Persistência de Dados (até 60 pontos)**
 - ❑ Implementar pelo menos 3 CRUDs
 - ❑ Descrever e implementar a funcionalidade principal (core) da aplicação backend

DÚVIDAS?

Procure o Scrum Master mais perto!

Obs. Dúvidas sobre as entregas das disciplinas, procure o professor responsável pela disciplina.