

1º ENTREGAS

FIAP

AI & CHATBOT (1/2)



Discuta com seu grupo e proponha o design de um sistema que utilize IA e/ou chatbots considerando a solução do **desafio proposto**. O design é um “rascunho conceitual” daquilo que será resolvido. Pensem nas interações que sua solução terá com usuários e se **chatbots** podem ser usados como canais primários ou secundários para a solução do desafio. Também pensem se serão necessários algoritmos de IA de **análise de dados**, **visão computacional**, ou **modelos generativos** para auxiliar nessa solução.

Requisitos - seu grupo deve gerar um relatório em formato **.pdf** contendo:

- Uma descrição geral da solução que o grupo pretende desenvolver como um todo (Qual é o objetivo do projeto final quando todas as disciplinas estiverem integradas? Quais algoritmos de IA vocês utilizarão e com quais objetivos?); **[25 pontos]**
- Um diagrama com o fluxo de utilização do sistema por um usuário considerando a interação via **chatbot**; **[15 pontos]**
- Discuta dois cenários de uso, propondo duas personas que interagem com seu chatbot/sistema. Explique por que o uso do seu sistema pode ser uma vantagem nesses cenários; **[20 pontos]**
- As intenções (pelo menos 6) que seu chatbot deverá identificar. Dê algumas frases de exemplo para cada intenção; **[25 pontos]**
- As entidades (pelo menos 3) que seu chatbot deverá identificar. Dê o nome das entidades e os sinônimos (quando aplicável); **[15 pontos]**
- **Importante:** o relatório deve conter nome e RM de todos os participantes do grupo.

Entregável:

- Arquivo do relatório **.pdf**;

BUILDING RELATIONAL DATABASE (1/3)

Após a análise das classe desenvolvidas na disciplina "**Domain-Driven Design (DDD)**", é crucial que a disciplina "Building Relational Database" atenda às necessidades da maioria dessas classes de negócio (ou domínio). Além disso, é fundamental considerar as funcionalidades criadas na disciplina "**Computational Thinking Using Python**" para garantir uma integração harmoniosa entre os sistemas.

Com base no levantamento de requisitos e nos feedbacks recebidos, propõe-se o esboço do Projeto de Banco de Dados Relacional, que visa atender às demandas identificadas. Este projeto será delineado com foco na estruturação eficiente e na otimização das relações entre os diversos elementos do sistema. Serão considerados aspectos como normalização de dados, integridade referencial e desempenho, visando garantir a robustez e a escalabilidade do banco de dados.

O esboço do Projeto de Banco de Dados Relacional será iterativo, permitindo ajustes e refinamentos com base em novas análises e feedbacks ao longo do desenvolvimento. A colaboração contínua entre as disciplinas envolvidas e os stakeholders será fundamental para garantir o alinhamento dos requisitos e a eficácia do projeto como um todo.

BUILDING RELATIONAL DATABASE (2/3)

- **REQUISITOS**

- Descrever o objetivo da solução definida pelo grupo e quais são as informações necessárias a persistir. **[30 pontos]**
- Desenvolva um esboço do Diagrama de Entidade e Relacionamento (DER) ou Diagrama Logical representando o Modelo Conceitual, contendo no **mínimo** 5 entidades com 4 atributos para cada uma. **[70 pontos]**
 - Utilize o Oracle Data Modeler ou o BrModelo para realizar o Diagrama.

BUILDING RELATIONAL DATABASE (3/3)



- **ENTREGÁVEIS**

- Arquivo **.pdf** sobre o descritivo do seu projeto
- Arquivo **.pdf ou .png** do seu Diagrama
- Arquivo **.txt** com o Nome e RM dos membros do grupo
 - Sugestão para nomeação dos arquivos:
 - 1TDSX_2024_Nome_Grupo
 - X= turma
 - Nome_grupo= Caso não tenha, pode deixar vazio

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

COMPUTATIONAL THINKING USING PYTHON (1/2)

▣ Código Fonte (Total 60 pontos)

- **(40 pontos)** Implementar um programa em **Python** que aborde o desafio proposto.
 - É esperada uma versão (mesmo que simples) **funcional** do programa, mesmo que seja o protótipo com a expectativa das telas. Estas poderão ser alteradas no decorrer dos sprints.
 - O programa deve incluir um **Menu** com pelo menos **3 funcionalidades** relevantes para o projeto.
 - Será analisada a consistência de dados, ou seja, se digitar valores inválidos - como item do menu existente ou entradas específicas - o programa deve advertir o usuário.
 - Cada funcionalidade deve ter uma implementação inicial que siga as regras de negócio planejadas.
- **(20 pontos)** Será avaliada a aplicação correta dos conceitos de:
 - Entrada, processamento e saída dados.
 - Armazenamento de dados em variáveis.
 - Estruturas condicionais
 - Estruturas de repetição
 - Comentários no código fonte
 - Formatações das saídas de dados
 - Usabilidade, para um melhor entendimento das telas sugeridas

COMPUTATIONAL THINKING USING PYTHON (2/2)

■ Documentação (Total 35 pontos)

- Arquivo **PDE** com a documentação da solução, contendo:
 - **(5 pontos) Capa** contendo o nome do projeto e o nome dos integrantes do grupo.
 - **(5 pontos) Sumário** (como o documento está organizado).
 - **(20 pontos) Descritivo** apresentando os objetivos e justificativa para o projeto, que permita a validação da entrega (se as funcionalidades refletem o contexto do projeto). E a descrição das funcionalidades que serão implementadas no projeto (mínimo de 15 linhas).
 - **(5 pontos) Anexo** contendo o código-fonte e/ou fluxograma da solução.

■ Formato de Entrega (Total 5 pontos):

- Arquivo compactado (**ZIP ou RAR**) contendo:
 - Arquivo **.pdf** com a documentação.
 - Arquivo **.py** com o código fonte do programa.

DOMAIN DRIVEN DESIGN (1/2)



- ▣ **Valor Total da entrega (100 pontos)**

- ▣ **Documentação (30 pontos):**

- **(5 pontos) Capa** contendo o nome da solução e o nome dos integrantes
- **(5 pontos) Sumário** (como o documento está organizado)
- **(20 pontos) Descritivo** explicando o projeto, contendo uma justificativa/objetivos de no mínimo quinze linhas para compreensão do contexto, permitindo a validação da entrega (se as classes refletem o contexto do projeto).

DOMAIN DRIVEN DESIGN (2/2)

- **(70 pontos) Modelagem das classes do projeto**
- (20 pontos) Conter no mínimo 10 classes.
- (25 pontos) Deve constar a representação de encapsulamento (visibilidade) das variáveis.
- (25 pontos) Seguir os padrões da representação de diagrama UML, incluindo a nomenclatura e tipos de dados .
- **Obs:** O Diagrama de Classes deve estar apresentável e com uma boa resolução, caso contrário será descontado na pontuação desse tópico.
 - Apenas o **diagrama de classe**, não é necessário desenvolver/enviar o código Java.
 - Os métodos **getters** e **setters** não precisam ser adicionados na modelagem.
- **Entrega:** Documento/especificação em formato **PDF**. Caso a imagem do diagrama fique sem definição, enviar a imagem do diagrama de classes em um arquivo separado.

FRONT-END DESIGN ENGINEERING (1/12)



- Você e sua equipe foram designados para o desenvolvimento do *front-end* da aplicação *web* do challenge.
- Neste desafio, iremos conceber o protótipo das telas para um novo site. A ferramenta designada para essa tarefa é o **FIGMA**, sendo sua utilização obrigatória.
- O processo inicia-se com a criação das estruturas e visualizações no **Figma**. Posteriormente, será necessário transformar esses *designs* em telas funcionais utilizando **HTML** e não podemos esquecer do versionamento com **Git/Github** ao criar o projeto físico **HTML**.
- O objetivo é garantir uma experiência de usuário coesa e responsiva, adaptada para diferentes dispositivos.

FRONT-END DESIGN ENGINEERING (2/12)

- Mas atenção, este protótipo precisa ser atraente e moderno.
- Todas as resoluções de tela deverão ser atendidas, pois na próxima entrega, deveremos ter nosso site responsivo.
- Algumas regras:
 - A quantidade de páginas, fica a critério de cada equipe, porém cuidado, pois na próxima etapa, essas telas deverão ser desenvolvidas.
 - Em uma das páginas(**Página de Integrantes**), deve conter obrigatoriamente, o nome, foto, RM, turma – independentemente se os integrantes forem de turmas diferentes, links para o **Github** de todos os integrantes da equipe.
 - Como mencionado anteriormente, a equipe deve utilizar como ferramenta de desenvolvimento, o Figma, obrigatoriamente.
 - **Deverão criar uma conta para a equipe, esta conta deverá ser disponibilizada ao professor responsável para disciplina, no período da avaliação.**

FRONT-END DESIGN ENGINEERING (3/12)



- A equipe deverá criar o protótipo no FIGMA para os seguintes breakpoints:
 - Extra Pequeno (XS) - Telefones Pequenos:
 - Largura: Menos de 576 pixels.
 - Pequeno (SM) - Telefones Médios e Tablets Pequenos:
 - Largura: De 576 pixels a 767 pixels.
 - Médio (MD) - Tablets e Laptops:
 - Largura: De 768 pixels a 991 pixels.
 - Grande (LG) - Desktops Médios:
 - Largura: De 992 pixels a 1199 pixels.
 - Extra Grande (XL) - Desktops Grandes:
 - Largura: 1200 pixels ou mais.

- **Wireframes Iniciais (10 pontos)**

- *Wireframes* básicos (2,5 pontos):
 - Esboce estruturas simples para cada tela do aplicativo, indicando a disposição dos principais elementos.
- Inclusão de elementos essenciais (2,5 pontos):
 - Certifique-se de que todos os elementos cruciais, como barras de navegação, botões de ação e campos de entrada, estejam presentes nos *wireframes*.
- Lógica na disposição (2,5 pontos):
 - Garanta uma disposição lógica dos elementos, seguindo a jornada do usuário de forma intuitiva.
- Documentação clara (2,5 pontos):
 - Adicione comentários ou notas explicativas nos *wireframes* para garantir clareza na compreensão do desenvolvedor.

- **Design Visual Inicial (10 pontos)**

- Paleta de Cores (2,5 pontos):
 - Desenvolva uma paleta de cores consistente e alinhada com a identidade visual da marca ou o tema do aplicativo.
- Estilo de Fonte (2,0 pontos):
 - Aplique estilos de fonte que sejam legíveis e adequados ao design, mantendo consistência em todo o projeto.
- Elementos Visuais (3,0 pontos):
 - Adicione elementos visuais atraentes, como ícones ou ilustrações, para enriquecer a experiência visual.
- Acessibilidade (2,5 pontos):
 - Considere a acessibilidade visual, garantindo contraste adequado e tamanhos de fonte legíveis.

- **Versão para dispositivos móveis (15 pontos)**

- Adaptação para Smartphones (5 pontos):
 - Modifique o design para telas de smartphones, ajustando a disposição e o tamanho dos elementos para tamanhos menores.
- Usabilidade em Dispositivos Móveis (5 pontos):
 - Certifique-se de que os elementos essenciais sejam facilmente acessíveis e utilizáveis em dispositivos móveis.
- Navegação Intuitiva (5 pontos):
 - Garanta uma navegação fluida e intuitiva, considerando gestos comuns em dispositivos móveis.

- **Versão para Tablets (10 pontos)**

- Consistência Visual (5 pontos):
 - Mantenha a consistência visual com a versão móvel, adaptando o design para telas de tablets.
- Layout Otimizado (5 pontos):
 - Otimize o layout para tamanhos intermediários de tela, garantindo uma experiência agradável.

- **Versão para Desktops (20 pontos)**

- Estrutura HTML Básica (5 pontos):
 - Crie a estrutura básica de HTML para cada tela do aplicativo, incluindo tags como <header>, <main>, <footer>, e outros elementos essenciais.
- Aparência Geral (5 pontos):
 - Adapte o design visual para telas de desktop, mantendo a consistência visual e ajustando a disposição dos elementos conforme necessário, mesmo que não fidedigna à prototipação.
- Responsividade Inicial (5 pontos):
 - Inclua os primeiros passos para tornar as telas responsivas em desktop, considerando a adaptação a diferentes tamanhos de tela.
- Eficiência de Carregamento (5 pontos):
 - Otimize o código para garantir um carregamento eficiente das páginas desktop.

- **Versionamento do projeto (15 pontos)**

- Criar um repositório privado para o projeto **e no dia da entrega dar permissão ao professor** no repositório da disciplina através de seu username do github. (5 pontos):
- Mínimo de 10 commits (5 pontos):

ATENÇÃO

- A participação de todos os integrantes do grupo é obrigatória. Caso um ou mais integrantes não apareçam nas métricas do repositório, estes não receberam a pontuação de versionamento.
- **O link do repositório deve ser inserido na página de integrantes devidamente sinalizada.**

FRONT-END DESIGN ENGINEERING (10/12)

- **Entrega final (20 pontos)**
 - Protótipo funcional (15 pontos):
 - **TODAS** as telas criadas no **Figma**, para versão desktop, devem ser implementadas, mesmo que seja de maneira não fidedigna aquela projetada no **Figma**.
 - Compatibilidade de Navegadores (5 pontos):
 - Teste o protótipo em diferentes navegadores para garantir compatibilidade.

- **Entrega do projeto**

- O grupo é encarregado de compactar e submeter o **repositório** do projeto no formato .ZIP, garantindo que o tamanho do arquivo não ultrapasse 50 MB, conforme o limite estabelecido pelo portal da FIAP. O não cumprimento dessa norma resultará em uma penalidade significativa, causando a **perda de cinquenta pontos** para todos os membros do grupo. Certifique-se de ajustar o tamanho do arquivo para atender a esta exigência e evitar quaisquer repercussões negativas.
- O líder eleito pela equipe é responsável por revisar o documento antes da entrega, garantindo a ausência de falhas ou equívocos. Recomenda-se realizar testes em múltiplas máquinas, se necessário.
- A entrega deve ser feita por apenas um aluno do grupo. Caso ocorra a **entrega por mais de um aluno, será descontado UM ponto do grupo para cada entrega adicional**.
- Alunos pertencentes a outras turmas devem ser claramente identificados com seus respectivos RM e turma em uma das páginas do projeto (página de integrantes).

- **Entrega do projeto**
 - Como mencionado anteriormente, a equipe deve utilizar como ferramenta de desenvolvimento, o Figma, obrigatoriamente.
 - **Deverão criar uma conta para a equipe, esta conta deverá ser disponibilizada ao professor responsável para disciplina, no período da avaliação.**

SOFTWARE DESIGN & TX (1/2)



- **PRODUCT BACKLOG** : Cada grupo deverá construir e entregar uma versão preliminar do Backlog do Produto no Trello ou Jira. Esse Backlog deve ter as funcionalidades, atividades e histórias dos usuários da solução. (30 pontos) . O link deve estar no PDF de entrega. Deve ser organizado, definido responsáveis para cada card ou subitem e datas reais.
- **DESCRIÇÃO DA SOLUÇÃO**: Para facilitar a compreensão do que o Backlog abrange, elaborar a descrição da solução destacando os aspectos diferenciais da ideia focados na "dor" da Porto. Como vocês iriam expor o seu challenge para alguém que não tem familiaridade com o assunto? (20 pontos)

SOFTWARE DESIGN & TX (2/2)



- PITCH : Exponha a sua solução em formato PITCH gravado, o link deverá ser aberto para o professor acessar.

Utilizando ferramentas, como por exemplo, o youtube, filmora, zoom, entre outras, contendo um vídeo de no máximo 3 minutos, caso o professor não tenha permissão para acessar este vídeo, isto é, o link esteja bloqueado, essa pontuação não será atribuída (35 pontos). O link deve estar no PDF de entrega. Proibido uso de 100% de IA para geração do Pitch.

- FORMATO DE ENTREGA: Em PDF, modelo exposição da solução para o cliente, com identidade visual da sua empresa com o nome da solução, com capa, sumário, nome dos integrantes e nome do seu grupo. (15 pontos)

Lembrem-se: Os 4 entregáveis são complementares, ou seja, a gestão do projeto deve estar de acordo com a descrição da solução e exposta no seu pitch. A forma de entrega deve ser organizada e interessante ao leitor.