**SEGUNDA ENTREGA DO CHALLENGE SPRINTS**

* **CHATBOT**
  + 5 intenções com no mínimo 12 exemplos.
  + 5 entidades com sinônimos.
  + Integração com Webchat (só texto).
  + Integrar com o telegrama por texto e áudio, utilizando STT e TTS.
  + Fazer um vídeo mostrando a utilização real do chatbot, tanto no Webchat tanto no telegrama, postar o vídeo o deve estar em um link privado no YouTube.

***ENTREGAR:***

* + Para entregar devemos Arquivo do Dialog Skill **.json**;
  + Fluxo de integração do Telegram em NODE-RED, arquivo **.json**;
  + Arquivo **.txt** com o nome dos membros do grupo, as credencias dos serviços usados e o link do vídeo no YouTube
* **BANCO DE DADOS**
  + É necessário indicar quais os dados/atributos necessários (ou importantes) para persistência E Mencionar as Regras de Negócio apuradas e Relacionadas as Entidades.
  + Entregar o Diagrama de Classes da **UML** desenvolvido na disciplina Domain-Driven Design (DDD).
  + Migre o seu Diagrama de Entidade e Relacionamento (DER) ou Diagrama Logical para a Modelagem Lógica (utilize a "Notação de Engenharia - Informações" no Oracle Data Modeler) devidamente atualizado com a:
  + Definição da Chave Primária para cada Entidade;
  + Definição dos Relacionamentos entre as Entidades;
  + Definição da Chave Estrangeira para cada Entidade Relacionada;
  + Definição dos Tipos dos Atributos [18 pontos

***ENTREGAR:***

* + Arquivo **.pdf** sobre o descritivo do seu projeto
  + Arquivo **.pdf** ou **.png** dos seus Diagramas (Modelagem Lógica e Classes)
  + Arquivo **.txt** com o Nome e **RM** dos membros do grupo
* **PYTHON**
* Implementação/atualização do menu de opções (implementado na Sprint 1), com as principais funcionalidades oferecidas pelo sistema, ao menos um MVP dos itens sugeridos. O programa deve permitir ao usuário escolher uma funcionalidade e fazer a chamada da função correspondente, e após a sua execução, o programa deve retornar para o menu principal.
* O programa deve realizar validações nas entradas de dados do usuário e na manipulação do menu.
* Será avaliada a aplicação correta dos conceitos de:
* Estruturas condicionais e de repetição.
* Funções com passagem de parâmetro e retorno.
* Armazenamento e manipulação de dados em Listas

***DOCUMENTAÇÃO ATUALIZADA:***

* A partir do formato da documentação proposta na Sprint 1, atualizar o Sumário, o Descritivo do Projeto (incluindo as funcionalidades implementadas) e os Anexos que contém o código-fonte e/ou fluxogramas.
* Descrever textualmente as mudanças da Sprint 1 para a Sprint 2, ou seja, o que foi acrescentado, alterado ou retirado do projeto

***ENTREGAR:***

* Arquivo compactado (**ZIP** ou **RAR**) contendo:
* Arquivo **.pdf** com a documentação.
* Arquivo **.py** com o código fonte do programa
* **JAVA**
* Criação do projeto Java com todas as classes, atributos e em seus devidos pacotes, que deve estar de acordo com o Diagrama de Classe e o projeto proposto.
* Conter **métodos construtores, Getters e Setters.**
* Conter no mínimo dois métodos diferentes de funcionalidade do sistema.
* Conter uma classe para executar o programa, instanciar objetos fazendo teste com todas as classes e todas as saídas do projeto

***DOCUMENTAÇÃO:***

* Capa contendo o nome da solução e o nome dos integrantes
* Sumário (como o documento está organizado)
* 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).
* Modelagem das classes do projeto (mínimo de 10 classes)

***ENTREGA:***

* Arquivo **.pdf** para a documentação e um arquivo **.zip** com a pasta do projeto. A entrega deverá ser feita exclusivamente pelo portal, em formato **ZIP.**  Não serão aceitas entregas pelo GitHub.

TEM MAIS COISA LÁ PRA BAIXO DESCE UM POUCO MAIS

* **BUSINESS MODEL**

Estamos elaborando a documentação do seu projeto, portanto, vamos agregar todos os entregáveis abaixo no documento anterior. O grupo deverá complementar o documento entregue na Sprint anterior- PDF os seguintes itens:

1. CONTEXTO DO PROBLEMA: o
   * Business Model Canvas da sua solução para a Porto Seguro.
   * Mapa de Empatia do público-alvo identificado na solução do grupo.
2. CONTEXTO DA SOLUÇÃO:
   * Solução: A partir do feedback recebido pelos professores na Sprint 1 e do resultado dos itens anteriores: -
     + ***Matriz CSD*** - Inclua no documento (PDF anterior) uma Matriz CSD para visibilidade do entendimento atual do grupo
     + ***Sua solução***: Refine e descreva com mais detalhes a solução do grupo para resolver o desafio através das tecnologias que serão vistas durante o curso. A solução proposta será considerada para validar as histórias do Backlog da Sprint 2 – Deve conter as correções solicitadas da Sprint anterior.

* + - ***Product Backlog da Sprint 2***: Mantenha o link do (Trello/Jira) com acesso liberado para o backlog da Sprint 2. As histórias neste momento precisarão conter os requisitos funcionais, não funcionais e regras de negócio especificados.
    - ***Diagrama de Caso de Uso*** – Elaborar os Diagramas completos de Casos de Uso que explicam as funcionalidades do sistema: Represente as funcionalidades da sua solução nos seus diagramas no ASTAH (Mínimo de três diagramas de caso de uso)
    - Utilize todas as ferramentas que foram ensinadas em sala, os documentos descritivos devem acompanhar cada caso de uso.
    - Não será aceito uso de outras ferramentas.
    - Não será aceito diagrama sem os documentos descritivos.
    - O uso correto dos conceitos deve ser praticado, caso contrário não será aceito
    - ***Pitch:*** Inclua no documento um link com acesso liberado para a gravação em vídeo do PITCH da solução atual.
    - O pitch deverá possuir **no máximo 3 minutos**, ✓
    - O Uso de IA para geração automática do Pitch não será aceito. ✓
    - O pitch deve ser produzido com dedicação. ✓
    - Vídeo com apenas os alunos/aluno explicando verbalmente não é permitido. ✓
    - Deve ter imagens da solução (reais ou protótipo). ✓
    - O Link deve ser enviado no arquivo no **PDF**. ✓
    - **Caso o link esteja corrompido, seja zerado a nota**

* **FRONT-END**
* Para este desafio, vamos implementar o que foi prototipado na primeira Sprint.
* O objetivo é garantir uma experiência de usuário coesa e responsiva, adaptada para diferentes dispositivos.
* Garantindo a qualidade e seguindo os padrões de desenvolvimento técnicos, por exemplo utilizando o GitHub como ferramenta de versionamento.
* **Estrutura básica HTML**
  + Criação de Arquivos:

Desenvolver arquivos HTML separados para cada página do protótipo, dentro dos padrões e convenções de nomenclatura já apresentado

* + Estrutura essencial:

Implementar estruturas básicas HTML com tags <html>, <head> e <body> para cada página.

Incluir metatags essenciais, como aquela relacionada à viewport e outros requisitos específicos.

* **Semântica e estruturação**
  + Tags Semânticas:

Utilizar tags semânticas HTML5 para estruturar o conteúdo, como <nav>, <article>, <section>, <header> e <footer>.

* + Hierarquia Clara:

Garantir uma hierarquia de elementos HTML clara e lógica, refletindo a estrutura do wireframe.

* + Atributos Semânticos:

Utilizar atributos semânticos, como role e aria-\*, para aprimorar a acessibilidade

* **Estilo Responsivo com CSS**
  + Implementação de CSS:

Desenvolver estilos CSS para cada página, mantendo a consistência com o wireframe inicial.

* + Media Queries para Desktop:

Integrar Media Queries para garantir uma apresentação otimizada em telas de desktop.

* + Media Queries para Tablets:

Adaptação do design para telas de tablets usando Media Queries

* **Estilo Responsivo para dispositivos móveis**
  + Media Queries para Dispositivos Móveis

Implementar Media Queries para garantir uma experiência de usuário apropriada em dispositivos móveis.

* + Aprimoramento Visual para Móveis:

Ajustar o estilo para otimizar a apresentação em dispositivos móveis.

* + Navegação Intuitiva para Móveis:

Garantir uma navegação intuitiva e fácil de usar em dispositivos móveis

* **Incorporação de elementos de formulários**
  + Incorporação de Elementos de Input:

Integração de elementos de input conforme documentação e entregas anteriores

* + Estilo dos Elementos de Input:

Aplicação de estilos apropriados aos elementos de input para uma apresentação coesa

* **Versionamento do projeto**
  + 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**
  + 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.***
* **ENTREGA:**
  + 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.**