

DacBot

Projetos em Sistema de Computação

Problema

Motivação e propósito

- Há diversos estudos sobre a contribuição do uso de chatbots para alívio de filas de atendimento.
- A DAC já tinha tentado desenvolver e aplicar um chatbot, porém foi descontinuado. Este, descontinuado, inspirou a priorização das requisições do DacBot.

Fontes: E-Commerce Brasil e Época Negócios

Estudo de Requisitos

Pesquisa e Levantamento

- Criamos um Forms e enviamos através do grupo da Unicamp, entre outros canais para mapear as demandas dos estudantes.
- Nos reunimos com a TI e funcionários da DAC para descobrir as demandas dos funcionários, e assim termos uma lista de prioridades.

Reunião DAC

- Chatbot antigo era semelhante a um FAQ, com intenções mapeadas “hard coded”
- Principais dúvidas dos alunos: calendário, aluno especial, problemas com RA, problemas com senha, opção curricular.

Forms Alunos Unicamp

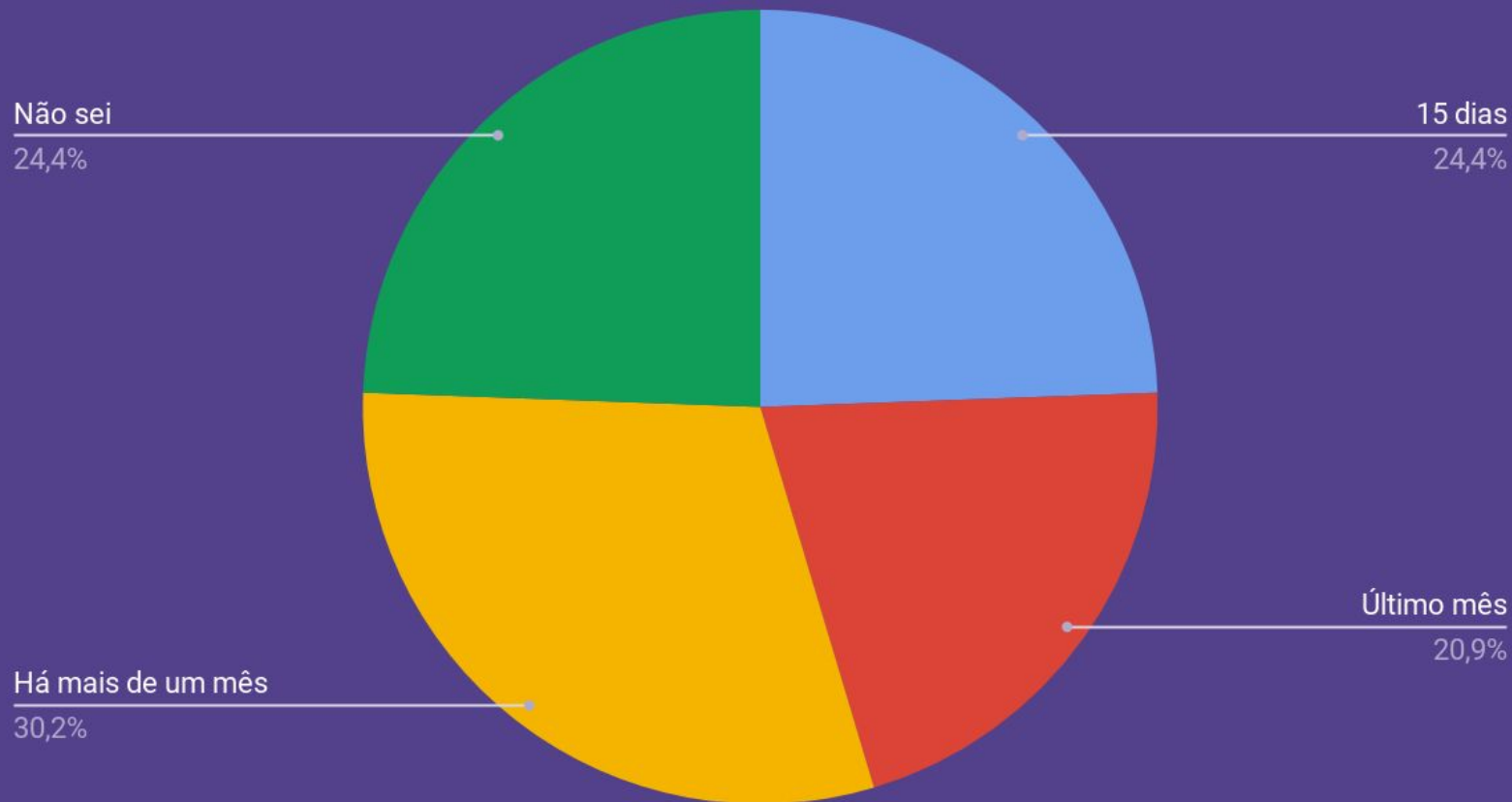
Os tipos de perguntas que fizemos encaixavam dentre os seguintes assuntos:

- Frequência de comparecimento presencial na DAC
- Se o problema foi resolvido ou não
- Motivos de ir na DAC

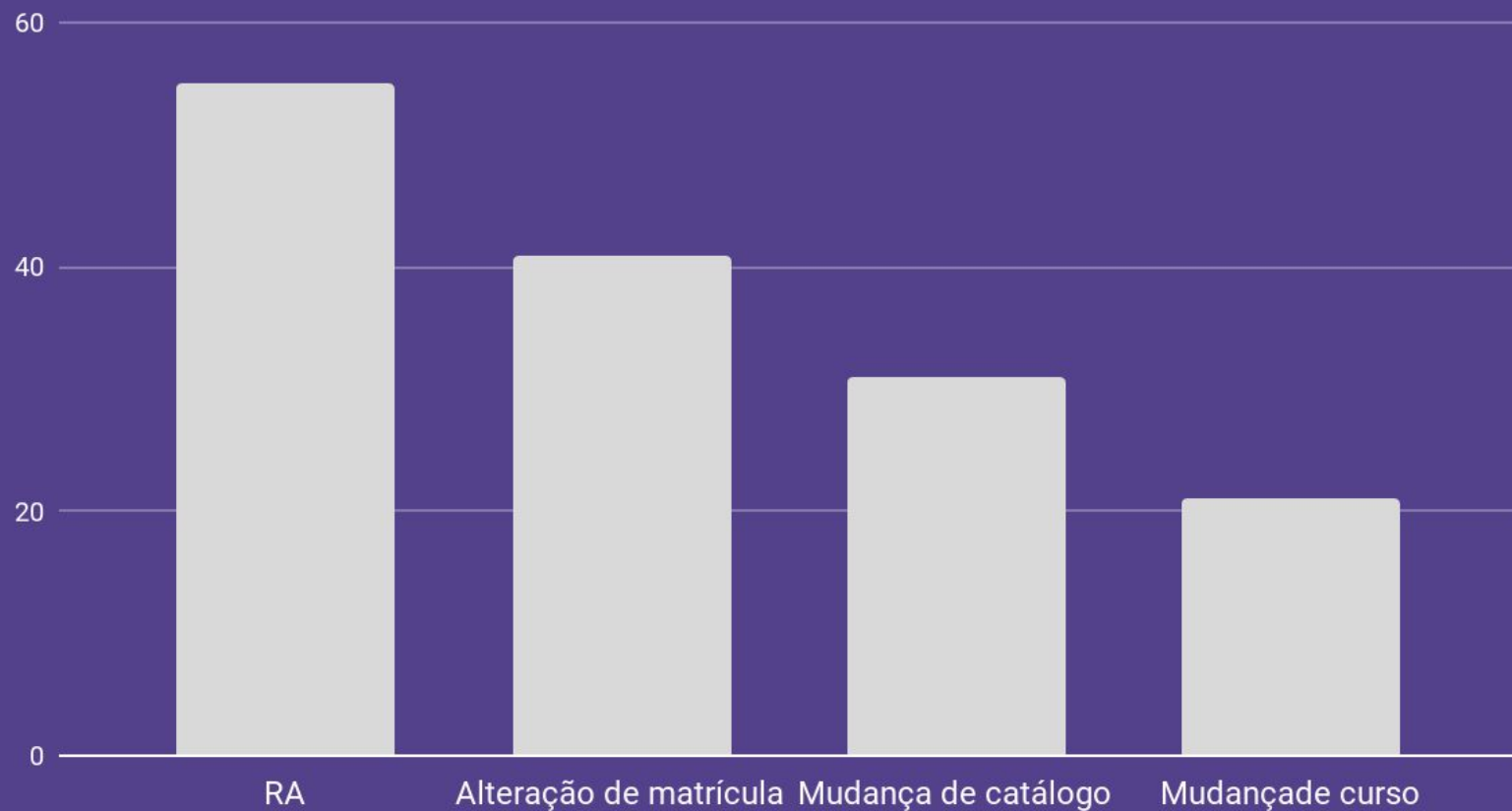
Respostas

- Todos vão no mínimo uma vez na DAC por semestre
- Quase 50% tinham ido dentro de um mês
- 80% dos problemas foram resolvidos

Última vez que precisou ir na DAC



Dúvidas na DAC (%)



Desenvolvimento

Metodologias

- Utilizamos o método Kanban em sprints de 2 semanas, cujas tarefas para a sprint seguinte poderiam ser revistas quando a anterior acabasse caso fosse prioritário.
- O método Kanban era implementado através do GitLab com seus Cards, Issues entre outros recursos.

Tecnologias

- Optamos pelo **LUIS** como sendo responsável pelo processamento de linguagem natural (**NLP**), juntamente com BLiP para gerenciar e mapear as intenções, entidades, as respectivas API's e a relação com o canal que é o BLiP Chat.
- O Azure foi cogitado porém descartado por motivos de custo inviabilizando seu uso pela DAC.

Design do Fluxo

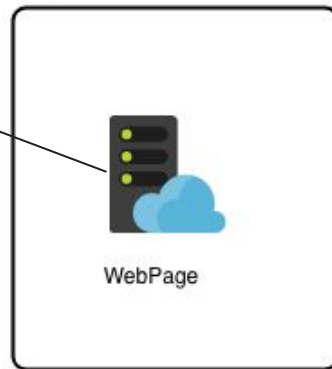
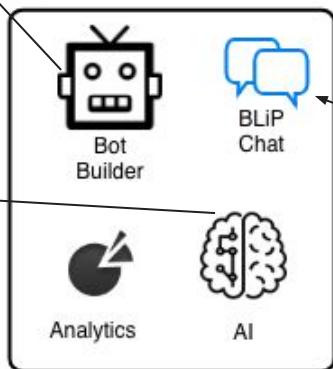
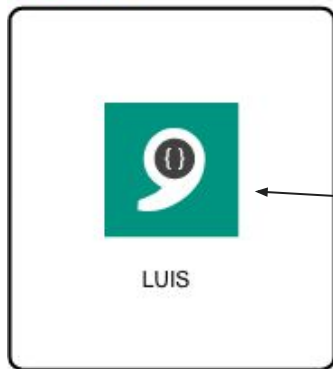
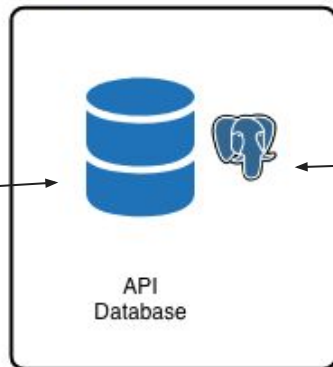
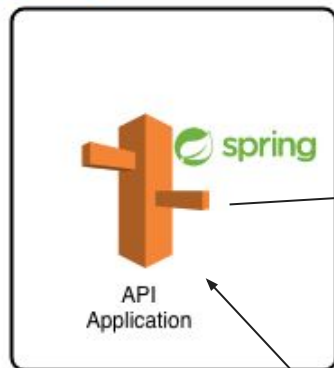
- O fluxo foi feito para que os alunos possam ter suas dúvidas sanadas em menor tempo possível.
- Contando com um "menu" de botões para acesso rápido com as dúvidas mais frequentes, e posterior encaminhamento para as respostas das intenções, indo para um feedback do usuário e voltando para esperar por novas perguntas.

Demo

Solução

Pontos de decisão sobre a arquitetura

- Escolha das plataformas cloud visando, além da necessidade de robustez e escalabilidade, a disponibilidade de serviços *free tier*.
- Linguagens de programação/ferramentas escolhidas com base na aptidão da equipe e tendências de mercado.
- Plataformas e fluxo de desenvolvimento buscando estruturar uma solução de fácil evolução.



LUIS

- O LUIS é um serviço de API de compreensão de linguagem oferecido pela Microsoft (Language Understanding) que fica hospedado na nuvem.
- Utiliza aprendizado de máquina no texto do usuário para prever o significado geral e extrair informações relevantes e detalhadas.

Language Understanding

Social Media
Feedback
Monitoring &
Response



Speech enabled
Intelligent desktop
app for order
tracking



Bot servicing product
queries and feedback

Chat bot

I want to call my HR rep.

You

Your HR rep is John Smith johns@contoso.com 212-555-1212

bot at 1:28:00 PM



Type your message...



Request



Response

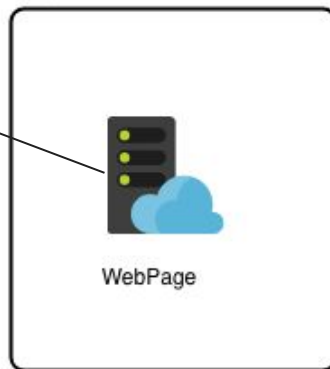
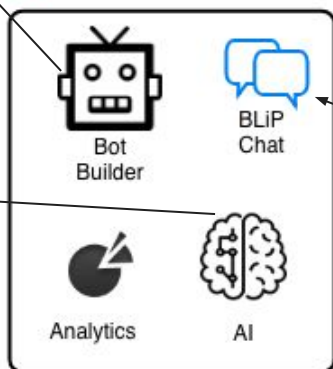
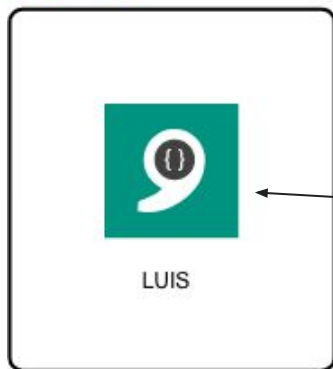
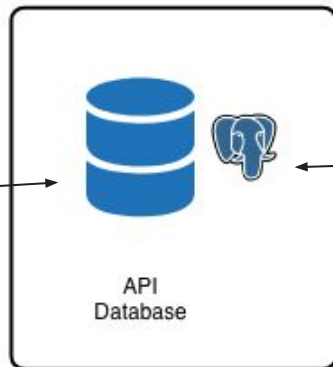
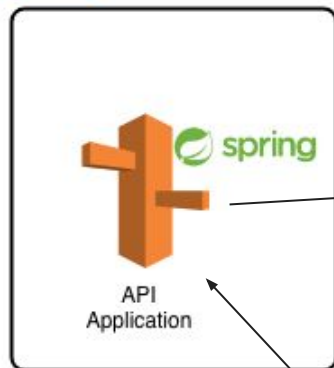
Language Understanding (LUIS)

```
{
  "query": "I want to call my HR rep.",
  "topScoringIntent": {
    "intent": "HRContact",
    "score": 0.921233
  },
  "entities": [
    {
      "entity": "call",
      "type": "Contact Type",
      "startIndex": 10,
      "endIndex": 13,
      "score": 0.7615982
    }
  ]
}
```

LUIS

O treinamento do LUIS é feito por sentenças. Ele analisa a estrutura linguística de cada sentença, de modo a poder extrapolar suas previsões mesmo quando estiver avaliando frases aparentemente diferentes daquelas que “ensinaram” o modelo.

Um bom treinamento envolve sentenças com diversos formatos e estilos diferentes, até mesmo com erros ortográficos.



API

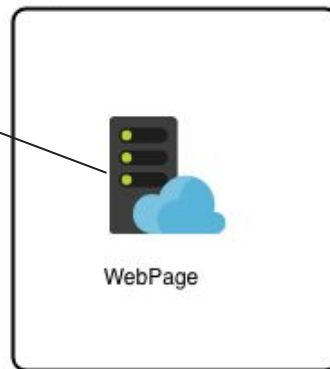
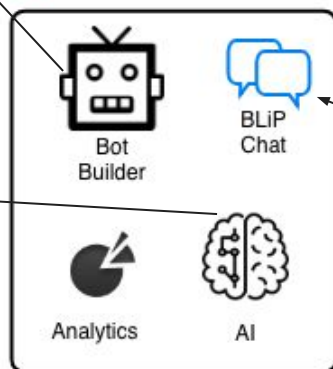
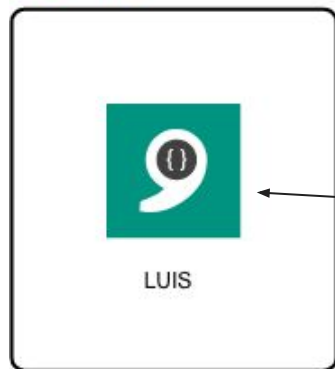
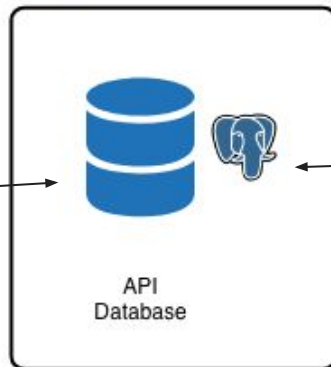
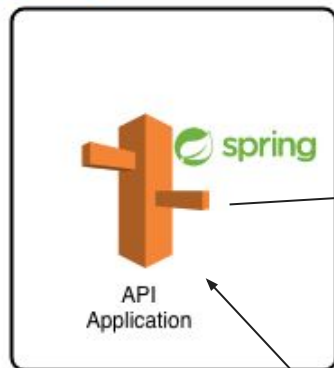
- Hospedada na **AWS** através do serviço **Elastic Beanstalk** (Amazon Linux VM), a API foi desenvolvida em **Java** e **Spring Framework**, um dois mais populares para desenvolvimento de Aplicações Web.

API

- A utilização do serviço da AWS permite alta **escalabilidade** e **disponibilidade** da aplicação. Podendo assim, facilmente se criar novas instâncias desta ou aumentar a capacidade da máquina virtual que a hospeda.
- Além disso, o serviço de cloud utilizado possibilita a configuração de diversas camadas de segurança, ambientes, alertas de funcionamento, análise de dados e muito mais.

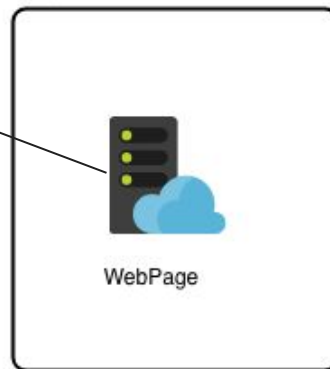
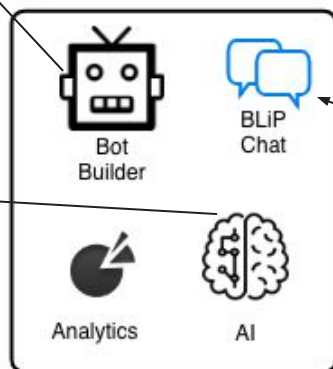
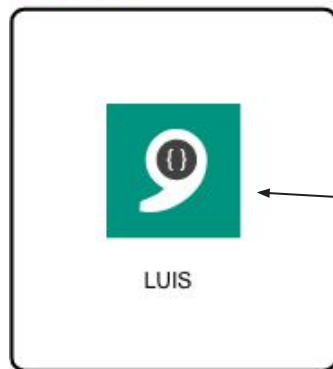
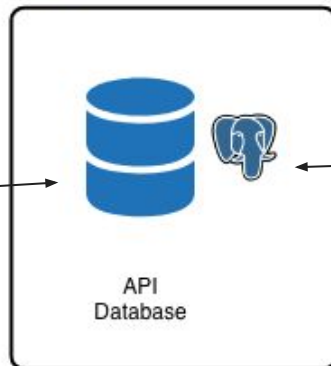
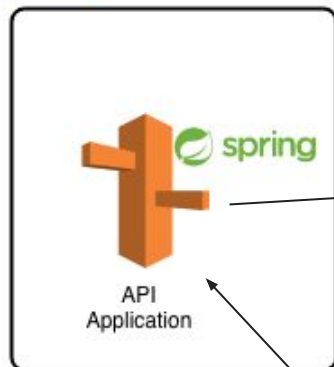
API

- A função da API dentro do sistema distribuído do Chatbot é expor serviços para buscar informações do banco de dados suportado pela cloud Heroku.
- Possibilitando assim, que o retorno das consultas tenha um formato mais amigável para se utilizar no chatbot, maior facilidade para se construir novas queries e segurança para se acessar os dados.



Scrapper

- Feitos em python, usando a biblioteca BeautifulSoup.
- Automatiza processo de pegar informações do site da DAC.
- Formulários, cadernos de horários, calendário.
- Arquivo de saída CSV.

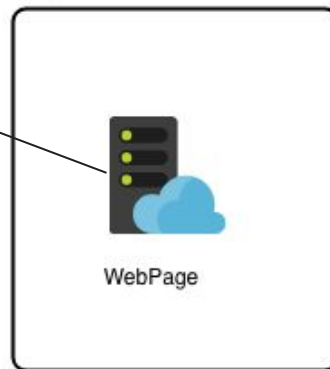
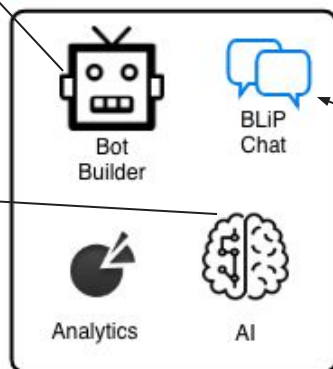
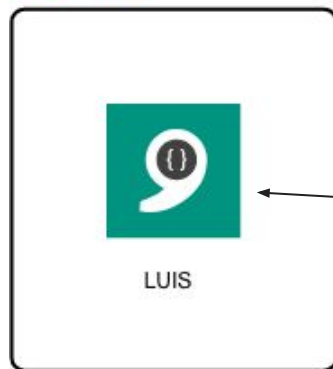
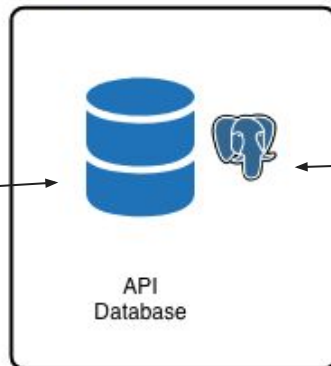
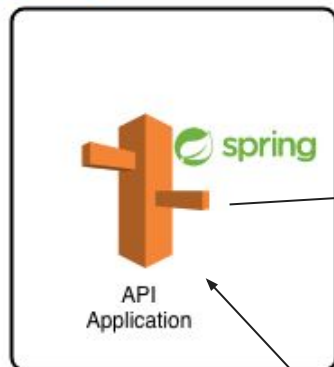


Banco de Dados

- O BD é preenchido através do scrapper que cria um script SQL com os dados vindos do portal da DAC.
- O PostgreSQL foi escolhido por ser um banco de dados relacional e Open Source. Além disso, é um dos BD mais difundidos e utilizados por ser altamente confiável, robusto e de alta performance.
- Várias Clouds possuem deployment direto usando o PostgreSQL, o que facilita também a manutenção e atualização do BD.

Banco de Dados

- Criamos o banco de dados PostgreSQL no Heroku Cloud contendo informações sobre atividades do calendário que foram extraídas pelo scrapper. Informações tais como: a descrição, semestre, intervalo de data e a URL de onde foi extraída a informação.
- A API acessa o BD pelo Heroku Cloud através do JDBC.



BLiP

- O BLiP é uma conceituada plataforma de construção e gerência de chatbots. Sendo também um **hub** de tecnologias condizentes ao segmento.

BLiP

- O BLiP já possui nativamente integração com diversos canais de mensageria, provedores de IA, ferramentas de pagamento online, ferramenta de atendimento humano.
- Além disso, a plataforma possui ferramentas e SDKs para possibilitar novas integrações.

Testes A/B com usuários

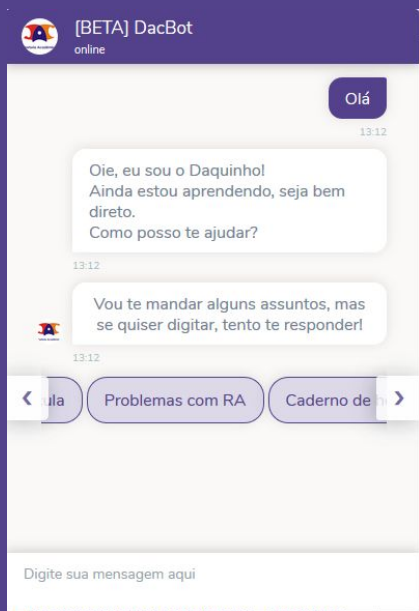
Testes A/B consistem em testes realizados em duas versões de um produto contendo variações entre elas, a fim de avaliar o impacto de inovações de interface ou funcionalidades para o usuário. O objetivo é identificar ou descartar alterações que podem aumentar ou não o engajamento dos usuários com o sistema.

Testes A/B com usuários

Realizamos testes com usuários finais (10 alunos) utilizando dois modelos de chatbot:

- Com botões de acesso no início da conversa com o chatbot.
- Sem botões, apenas texto por parte do usuário.

Testes A/B com usuários



- Os usuários preferem a versão com botões no início, afirmando que é um acesso mais rápido às dúvidas principais
- Em relação a interação por texto em ambos modelos, usuários avaliaram como bom o nível de compreensão das frases, apenas apontando como melhorias a interpretação das frases quando gírias são usadas.

Próximos passos

Manutenção

- Adicionar novas intenções e conteúdos para respostas.
- Treinar e refinar mais exemplos, partindo das interações do bot com os usuários.
- Analisar os dados de respostas e fluxos dos usuários, através de relatórios de *analytics*.
- Executar os scripts de scraping para manutenção dos dados do BD.

Aprendizado e Conhecimentos aplicados

- Funcionamento de um chatbot
- Conceitos de NLP
- Web Scraping
- Clouding
- Provedores de IA
- Soft Skills
- BD
- Engenharia de Software
- Web Services
- Testes A/B

Perguntas?

Obrigado!