



APS

- **Assistente Pessoal de Saúde**

BIC - Business Intelligence Corporation

Pedro Dib, RM84735

Antonio Sassine Mendonça, RM84297

Artur Fonseca Pinheiro, RM86397

Igor Ferreira Calbo, RM81274

INTRODUÇÃO

SUMÁRIO

Este documento explicita o levantamento e análise dos requisitos do APS. APS é um aplicativo responsivo de comunicação que utiliza um Chatbot, cuja principal função é julgar a gravidade dos sintomas do paciente e aconselhá-lo a ou ir ao hospital ou tentar algum tratamento caseiro, evitando assim superlotação de hospitais e contágio desnecessário.

CONVENÇÕES

APS - Assistente pessoal de saúde.

Stakeholders - Partes sem as quais não seria possível a construção do projeto.

DB - Banco de Dados.

RF - Requisito Funcional.

RNF - Requisito Não Funcional.

RN - Regra de Negócio.

UX - Experiência do Usuário.

PÚBLICO ALVO

O APS é um protótipo de um aplicativo de saúde e seu público alvo engloba todos que tem dúvidas sobre procurar um médico ou se tratar em casa.

ESCOPO

Ao acessar, o paciente tem acesso a: uma lista generalizada de sintomas e, para cada sintoma selecionado, uma escala de 1 a 10 da intensidade do sintoma.

Ao selecionar todos os sintomas que sente e seus graus de intensidade, o paciente pode solicitar uma análise feita pelo APS e, tanto antes quanto depois da análise, agendar um atendimento hospitalar - ao julgar que é isso de que necessita.

O APS, já munido desde o começo de seus dados pessoais e localização, decide o hospital mais perto de você com atendimento e/ou leitos disponíveis e marca a consulta para você.

RESPONSABILIDADES

Gestão do projeto	Pedro Dib
Back-end Servidores	Antonio Sassine Mendonça
Front-end	Artur Fonseca Pinheiro
Banco de dados Infraestrutura	Igor Ferreira Calbo

ELICITAÇÃO DE REQUISITOS

LEVANTAMENTO DE REQUISITOS

O projeto foi solicitado pelo professor Allen, da FIAP. Foi pedido que os alunos desenvolvessem alternativas tecnológicas para solucionar alguns dos problemas que estão sendo encontrados durante a pandemia. O grupo BIC - Business Intelligence Corporation decidiu desenvolver uma aplicação que visasse evitar superlotação em hospitais e unidades de saúde, reservando-os assim para aqueles que verdadeiramente precisam.

O levantamento de requisitos do APS foi feito pela equipe através de pesquisa e um *brainstorm* com alguns *stakeholders*. Pesquisamos muito sobre o tema e como aplicações de cunho similar funcionam. Conversamos, também, com alguns dos professores sobre a viabilidade de algumas coisas, e foram consultados, também, profissionais da área de saúde.

ANÁLISE DE REQUISITOS

“Depois do levantamento de requisitos, o projeto passa pela etapa de análise desses mesmos requisitos. A classificação dos requisitos permite a equipe descobrir inconsistências, pontos inconclusivos e conflitos entre os requisitos apresentados.”

(SOMMERVILLE, 2007)

REGRAS DE NEGÓCIO

RN01 - TELA INICIAL

Mostrar na tela inicial o bot de auxílio e algumas opções primordiais, como ajuda e emergência.

RN02 - BUSCA POR UNIDADES DE SAÚDE

Mostrar um campo de pesquisa para que o usuário possa verificar unidades de saúde próximas e suas lotações.

RN03 - DIVISÃO EM CATEGORIAS

A aplicação deve ser dividida em três categorias: Chatbot, Agendamento, Pesquisa.

RN04 - SEGURANÇA DE DADOS

A aplicação deve manter os dados do usuário disponíveis somente ao usuário.

RN05 - BLOQUEIO DE ACESSO

No caso do usuário errar a senha 3 vezes, deve ser bloqueado o seu acesso e um email de desbloqueio será enviado para o seu email vinculado com a aplicação.

RN06 - ALTERAÇÃO DE CADASTRO

A aplicação deve permitir alterações no cadastro.

RN07 - PERMISSÕES DO CHATBOT

O chatbot deve ter permissão de acesso ao banco de dados para interagir com o usuário.

RN08 - INTENÇÃO DE DELETAR

Caso o usuário queira deletar o aplicativo, deve ser informada sua senha para prosseguimento da operação.

RN09 - PÁGINA DE AJUDA

Deve existir, na tela inicial, uma página de ajuda com algumas FAQs (perguntas frequentes) e um formulário de Fale Conosco, formulário este que necessita de email e nome completo do usuário para envio de uma mensagem.

RN10 - ENCERRAMENTO DA SESSÃO

Caso o usuário fique sem interagir com a aplicação por 30 minutos, sua sessão será encerrada - por motivos de segurança de dados - e ele precisará fazer o login novamente.

REQUISITOS FUNCIONAIS

RF01 - CADASTRAR USUÁRIOS

O sistema deve permitir que uma pessoa física se cadastre como usuário.

RF02 - ALTERAR CADASTRO

O sistema deve permitir alteração dos dados providos no cadastro.

RF03 - EXCLUIR CADASTRO

O sistema deve permitir exclusão do cadastro de um usuário.

RF04 - CADASTRAR SINTOMAS

O usuário deve poder cadastrar seus sintomas no aplicativo, através do Chatbot.

RF05 - INFORMAR SOBRE COVID-19

O usuário pode se informar sobre a pandemia do Covid-19 através do Chatbot.

RF06 - MARCAR CONSULTA

O usuário deve poder marcar uma consulta com uma unidade de saúde, através do Chatbot.

RF07 - DESMARCAR CONSULTA

O usuário deve poder desmarcar uma consulta com uma unidade de saúde, através do Chatbot.

REQUISITOS NÃO FUNCIONAIS

USABILIDADE

RNF01 - INTERFACE GRÁFICA

A interface gráfica deve ser consistente, intuitiva e de fácil compreensão dos usuários.

RNF02 - RESPONSIVIDADE

O sistema deve ser responsivo para atender a necessidade de seus usuários tanto em desktops quanto em aparelhos móveis.

CONFIABILIDADE

RNF03 - TEMPO DE RETORNO

O sistema deve ter um tempo de retorno de, no máximo, 30 minutos após alguma falha.

RNF04 - DISPONIBILIDADE

O sistema deve estar disponível para os usuários 24 horas por dia, 7 dias por semana. Updates devem ser feitos em versões testes, e depois aplicados na versão disponível para o usuário.

DESEMPENHO

RNF05 - TEMPO DE RESPOSTA

O tempo de resposta dos questionamentos simples deve ser de, no máximo, 15 segundos. Em caso de agendamento de uma consulta, 30 segundos.

SEGURANÇA

RNF06 - BLOQUEIO DE CONTA

No caso de erro de senha na terceira tentativa, sua conta será bloqueada e será enviado um email para que possa desbloqueá-la.

HARDWARE E SOFTWARE

RNF07 - LINGUAGENS

O sistema deve ser implementado utilizando a linguagem Java, utilizando os conceitos do paradigma de orientação a objeto. Além disso, a interface gráfica deve ser implementada em HTML e CSS, contendo códigos em JavaScript.

RNF08 - BANCO DE DADOS

O banco de dados utilizado será o Oracle MySQL, devendo estar disponível o tempo todo para alimentar o sistema.

RNF09 - SERVIDOR

O servidor utilizado será o Node.js, que irá hospedar o banco de dados e eventuais necessidades do sistema.

BIBLIOGRAFIA

Looking to Technology to Avoid Doctors' Offices and Emergency Rooms

<https://www.nytimes.com/2019/02/21/business/medical-technology-ai-tests.html/>

SOMMERVILLE, Ian. Engenharia de Software. Tradução: Selma Shin Shimizu Melnikoff, Reginaldo Arakaki, Edilson de Andrade Barbosa. 8ª Ed. São Paulo: Pearson Addison-Wesley, 2007.