

ARQUITETURA DE UM ASSISTENTE VIRTUAL INTELIGENTE: APLICAÇÃO DE IA COMPORTAMENTAL NO AGENDAMENTO E NA AUTOMAÇÃO ADMINISTRATIVA PARA PROFISSIONAIS AUTÔNOMOS NA ÁREA DE SAÚDE E BEM-ESTAR.

Gustavo Kauan Storch

Engenharia de Software – 7º fase

Data da última revisão: 30/06/2025

RESUMO

Este projeto tem como objetivo desenvolver uma solução tecnológica para profissionalizar autônomos que enfrentam desafios na gestão de suas atividades administrativas. A proposta é criar um assistente virtual inteligente capaz de automatizar tarefas como agendamento de consultas e cobranças de clientes.

Com a implementação desta ferramenta, busca-se aliviar a carga administrativa dos profissionais, lhes permitindo dedicar mais tempo no atendimento aos seus clientes e no aperfeiçoamento de seus serviços.

Também será desenvolvido um sistema web complementar e por meio deste, o profissional poderá configurar parâmetros de funcionamento para a IA (como horário de atendimento, valores, atividades do profissional) e visualizar relatórios analíticos para acompanhar o número de atendimentos, taxas de confirmação e padrões de agendamentos.

A solução será projetada para ser simples, eficiente e intuitiva, otimizando o tempo e os recursos dos profissionais autônomos, ao mesmo tempo em que melhora a experiência do cliente.

ABSTRACT

This project aims to develop a technological solution to support self-employed professionals who face challenges in managing their administrative tasks. The proposal is to create an intelligent virtual assistant capable of automating activities such as appointment scheduling and client billing.

By implementing this tool, the goal is to reduce the administrative burden on professionals, allowing them to dedicate more time to serving their clients and improving their services.

A complementary web system will also be developed, through which professionals will be able to configure the AI's operating parameters (such as working hours, pricing, and services offered) and access analytical reports to monitor the number of appointments, confirmation rates, and scheduling patterns.

The solution will be designed to be simple, efficient, and intuitive, optimizing the time and resources of self-employed professionals while enhancing the overall customer experience.

PALAVRAS-CHAVE: Assistente Virtual Inteligente, chatbot, Telegram, Google Gemini, Google Calendar, Agente de IA, Profissionais Autônomos.

1. INTRODUÇÃO

O Brasil e o mundo passam a todo o tempo por mudanças e transformações que impactam de forma direta a saúde da população (GADELHA, 2022). Constata-se que a atual sociedade passa por uma crescente demanda na procura por serviços de saúde e bem-estar para minimizar os efeitos do rápido crescimento e evolução das gerações. Segundo Gadelha (2022), é possível notar uma mudança significativa no perfil da população e nas doenças, com menos nascimentos, mais gente idosa e doenças que duram um maior período, transformações no jeito de trabalhar, o aquecimento do planeta e a globalização.

Diante disso, práticas como o controle de processos e o monitoramento de indicadores de saúde são vitais para garantir um serviço de melhor qualidade (DA SILVA JÚNIOR et al. 2024). Ao utilizar profissionais para dar assistência aos primeiros atendimentos ao paciente, como recebê-lo na entrada, preencher o formulário e preparar toda a documentação necessária, o profissional permite-se a dedicar seu tempo exclusivamente ao atendimento do paciente. A utilização desses métodos visa não só a melhoria do atendimento ao paciente, mas também o aumento da eficiência organizacional (DA SILVA JÚNIOR et al. 2024).

Com o passar do tempo, podemos observar que os chatbots têm o potencial e estão transformando o atendimento ao cliente, oferecendo suporte

contínuo, 24 horas por dia, além de proporcionar experiências personalizadas que elevam a satisfação dos usuários. Dominar o uso dessas plataformas permite aos profissionais criarem soluções inovadoras, reduzindo custos operacionais e ampliando a capacidade de atendimento das empresas. (SANTOS, 2025).

Nesse mesmo contexto, a utilização de chatbots também se destaca como uma poderosa ferramenta no gerenciamento de clínicas e consultórios médicos, padronizando e facilitando o controle das atividades diárias. Eles contribuem diretamente para a organização do setor administrativo, agilizando serviços como agendamento de consultas, retornos, envio de confirmações e lembretes, além de fortalecer o relacionamento entre paciente e clínica. Esses sistemas ainda possibilitam a manutenção de um histórico detalhado de cada paciente, por meio de prontuários eletrônicos, garantindo mais segurança e eficiência no armazenamento das informações. (BRITO, 2018).

O problema que norteia este trabalho está relacionado ao planejamento e à execução das etapas de implementação de um chatbot em uma clínica, com o propósito de melhorar a comunicação entre paciente e profissional, integrar-se às plataformas de agendamento e ajustar automaticamente os fusos horários. Na prática, espera-se que o chatbot assuma tarefas como marcar consultas, esclarecer dúvidas e fornecer informações, permitindo que o profissional foque exclusivamente no atendimento clínico, além de gerar relatórios semanais e mensais sobre consultas e pacientes.

Com esse sistema, qualquer pessoa poderá agendar e obter suporte a qualquer hora, sem depender de um atendente em horário comercial, e o chatbot identificará automaticamente o fuso horário do cliente com base no DDD e no DDI, garantindo agendamentos corretos para consultas online em diferentes regiões do Brasil. Além disso, o chatbot será comportamental: Analisará o histórico de cada cliente para sugerir horários de consulta de acordo com padrões anteriores e aplicará políticas de cobrança diferenciadas para quem costuma remarcar ou atrasar pagamentos, incentivando a pontualidade e a assiduidade.

2. DESCRIÇÃO DO PROJETO

Esse projeto tem como objetivo o desenvolvimento de um assistente virtual inteligente voltado para profissionais autônomos na área de saúde e bem-estar com foco atual em psicólogos. A ferramenta proposta visa automatizar tarefas administrativas, como agendamento de consultas e cobrança de clientes, utilizando inteligência artificial para otimizar esses processos. O produto será um assistente virtual simples de usar, que permitirá a integração da automação diretamente no fluxo de trabalho diário dos profissionais. As funcionalidades incluirão envio de lembretes automáticos e inteligentes, controle de agenda e interações personalizadas com os clientes.

O assistente será utilizado via Telegram, e terá também um site onde os profissionais possam acessar e configurar o assistente conforme necessidade, a combinação dos dois fará com que os profissionais possam gerenciar seus compromissos de qualquer lugar e a qualquer momento.

Haverá o envio automático e inteligente de lembretes, a ideia principal é usar um agente (IA) responsável por estudar o comportamento de cada cliente. Por exemplo, se o cliente Gustavo costuma agendar suas consultas sempre nas terças-feiras à tarde, entre 15h e 17h, quando ele entrar em contato para um novo agendamento, o assistente virtual sugerirá automaticamente horários semelhantes aos padrões anteriores. Para clientes como João, que frequentemente remarca ou cancela suas consultas, o sistema enviará lembretes antecipados para confirmar a consulta, ajudando a evitar alterações de última hora.

Além disso, o sistema irá considerar o fuso horário de cada cliente e do profissional. Ao realizar um agendamento, o paciente poderá informar seu fuso horário, ou o sistema poderá determinar automaticamente o horário com base no local informado. O assistente virtual irá então sugerir horários compatíveis com a disponibilidade do profissional e dentro do limite de horário do paciente.

Por exemplo, se o paciente Gustavo mora nos Estados Unidos e deseja agendar uma consulta às 14h (horário local dele), o assistente irá sugerir automaticamente o horário equivalente para a psicóloga em Santa Catarina, ajustando a diferença de fuso horário entre os dois. Isso garante que ambos, paciente e profissional, vejam e entendam o horário correto do agendamento, evitando confusões de horário.

2.1. Problemas a resolver

- Gestão de agenda manual: Dificuldade dos profissionais autônomos em gerenciar seus compromissos, resultando em conflitos de horários e perda de consultas.
- Falta de lembretes automáticos: A ausência de notificações pode levar a atrasos e faltas, prejudicando tanto o profissional quanto o cliente.
- Carga administrativa excessiva: Tempo excessivo gasto com tarefas burocráticas reduz a disponibilidade para o atendimento ao cliente.
- Comunicação ineficiente: Agendamentos, cancelamentos e mudanças ainda são feitos de forma manual, aumentando a chance de erros e retrabalho.
- Agendamento entre fusos horários diferentes: Agendamentos serão possíveis serem feitos sem a falha de comunicação por conta de fusos horários, esses tipos de problemas podem ser mais comuns em consultas online.

2.2. O que será feito

- Chatbot via Telegram: Assistente virtual para comunicação e agendamento.
- Integração com calendários: Sincronização com Google Calendar para organização eficiente.
- Banco de dados (PostgreSQL): Armazenamento seguro de informações sobre clientes e agendamentos.
- Notificações automáticas: Lembretes para pacientes e alertas para os profissionais.
- Configuração personalizada: Interface para ajustar horários, valores e métodos de cobrança.
- Tela de login
- Tela para configuração de horários
- Tela para configurações de valores
- Tela para configurações de mensagens e demais permissões de integrações.

- Sugestões inteligentes de horários: O assistente analisará padrões e comportamentos de agendamento dos clientes para facilitar novas marcações, levando em conta as diferenças de fuso horário entre o cliente e o profissional.
- Linguagem de programação: Python (pela versatilidade e suporte a IA) e react para frontend referente às telas de configurações do sistema.
- Hospedagem: Solução em nuvem (Heroku ou AWS).

2.3. Limitações

- Integração com WhatsApp: A API do Whatsapp é paga, como o projeto não possui fim lucrativos ainda, não será realizado esta integração.
- Sistema próprio de pagamentos (integrações com plataformas existentes podem ser avaliadas posteriormente).
- Integração com plataformas de pagamento ainda não implementada na fase inicial.
- Software de gestão completo (ERP): O foco é a automação de tarefas específicas.
- Suporte para grandes empresas ou múltiplos profissionais: voltado para autônomos e pequenos negócios.
- Aplicativo móvel nativo: acesso via Telegram web ou aplicativo Telegram, já a parte de configurações do profissional será por meio do site web.
- Funcionalidades além do escopo administrativo (ex.: controle de estoque ou relatórios financeiros detalhados).
- Escopo focado em saúde e bem-estar: Outras categorias podem ser consideradas no futuro.
- Sem integração com sistemas complexos de gestão: Apenas funções básicas de agendamento, controle e lembretes.
- Sem automação completa de todas as tarefas administrativas: O foco inicial será em agendamento, lembretes e cobrança.

3. ESPECIFICAÇÃO TÉCNICA

A partir dos objetivos e funcionalidades apresentados na seção 2 referentes a descrição do projeto, se faz necessário o detalhamento de forma técnica da solução, como ela será estruturada e implementada. Será

especificado os requisitos de softwares, tecnologias envolvidas, protocolos de comunicação e os algoritmos que serão utilizados para integração da inteligência artificial, os procedimentos operacionais e os formatos de dados adotados. Essa especificação técnica visa garantir clareza no desenvolvimento e a correta integração entre os módulos do sistema.

3.1. Requisitos de software

- **Requisitos funcionais:**

- RF01 – Chatbot via Telegram para interação com pacientes e gerenciamento de agendamentos.
- RF02 – Integração com Google Calendar para sincronização de consultas.
- RF03 – Envios de lembretes automáticos personalizados para pacientes e profissionais.
- RF04 – Suporte a agendamento inteligente com análise de padrões de comportamento dos pacientes.
- RF05 – Gerenciamento de fusos horários entre profissionais e pacientes
- RF06 – Sistema web para configuração:
 - Tela de login.
 - Tela de configuração de horários de atendimento.
 - Tela para definição de valores.
 - Tela de personalização de mensagens.
 - Tela de Cadastro de usuário.
 - Tela de Cadastro de Clínica.
 - Tela de Cadastro de Profissional.
 - Tela de Cadastro de Clientes.
 - Tela de consultas.
 - Tela de agenda.

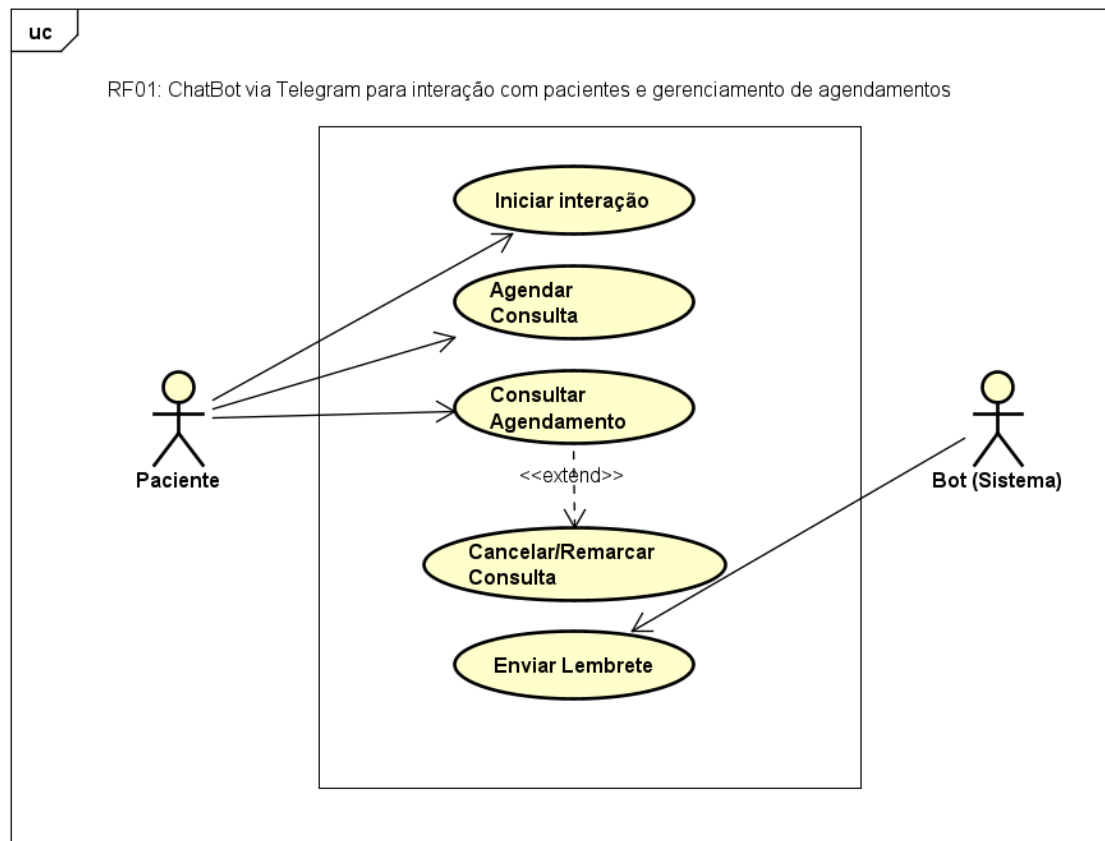
- **Requisitos não funcionais:**

- RNF01 – Interface web responsiva desenvolvida em React.
- RNF02 – Backend desenvolvido em Python utilizando o framework Flask.

- RNF03 – Hospedagem em nuvem (Heroku, AWS).
- RNF04 – Banco de dados relacional PostgreSQL, com acesso seguro com criptografia de pontas.
- RNF05 – Suporte a múltiplos fusos horários.
- RNF06 – Disponibilidade mínima de 99% (considerando hospedagem na nuvem).
- RNF07 – API RESTful para comunicação entre frontend, backend e bot do Telegram.

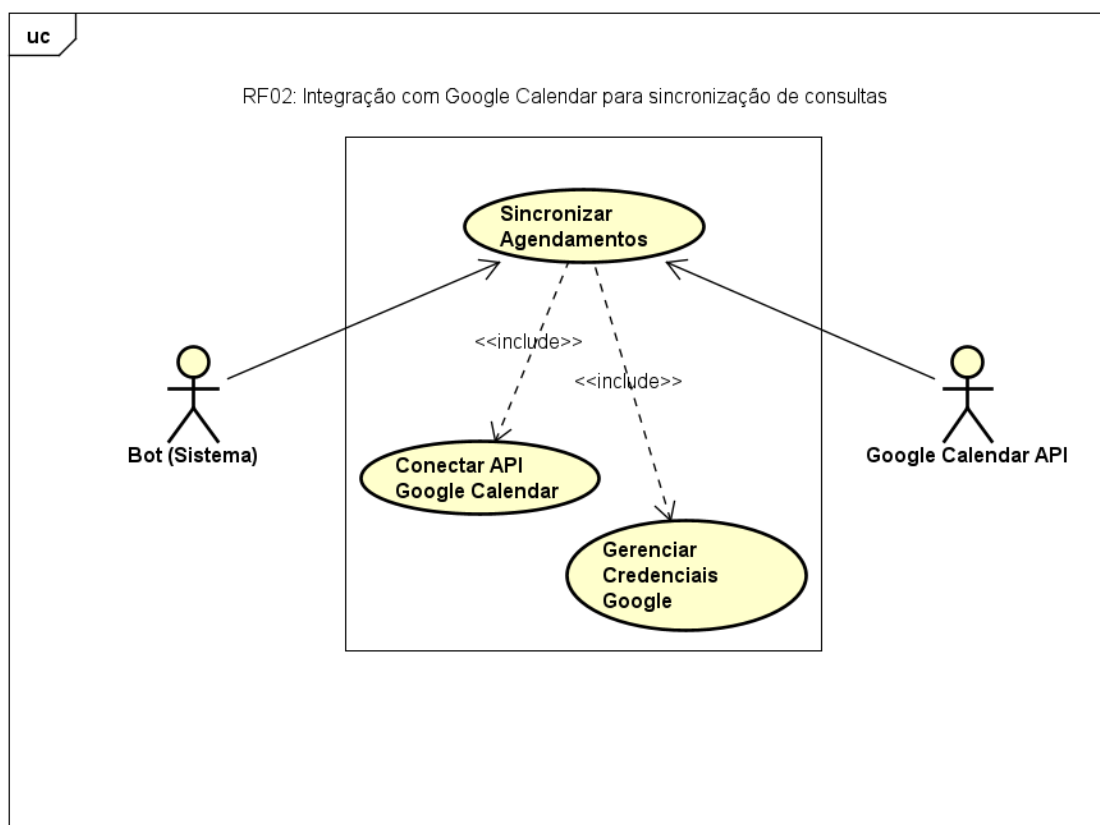
- **Diagramas de Requisitos Funcionais:**

Figura 01 - Requisito Funcional 1.



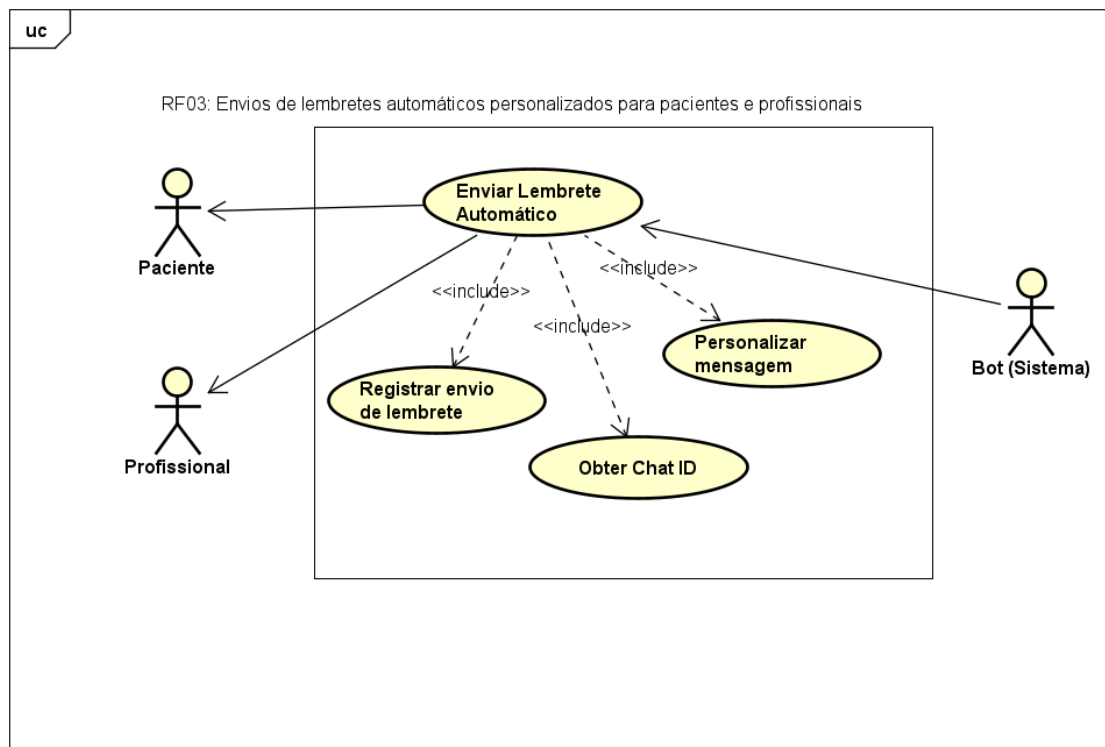
Fonte: Elaborado pelo Autor (2025).

Figura 02 - Requisito Funcional 2.



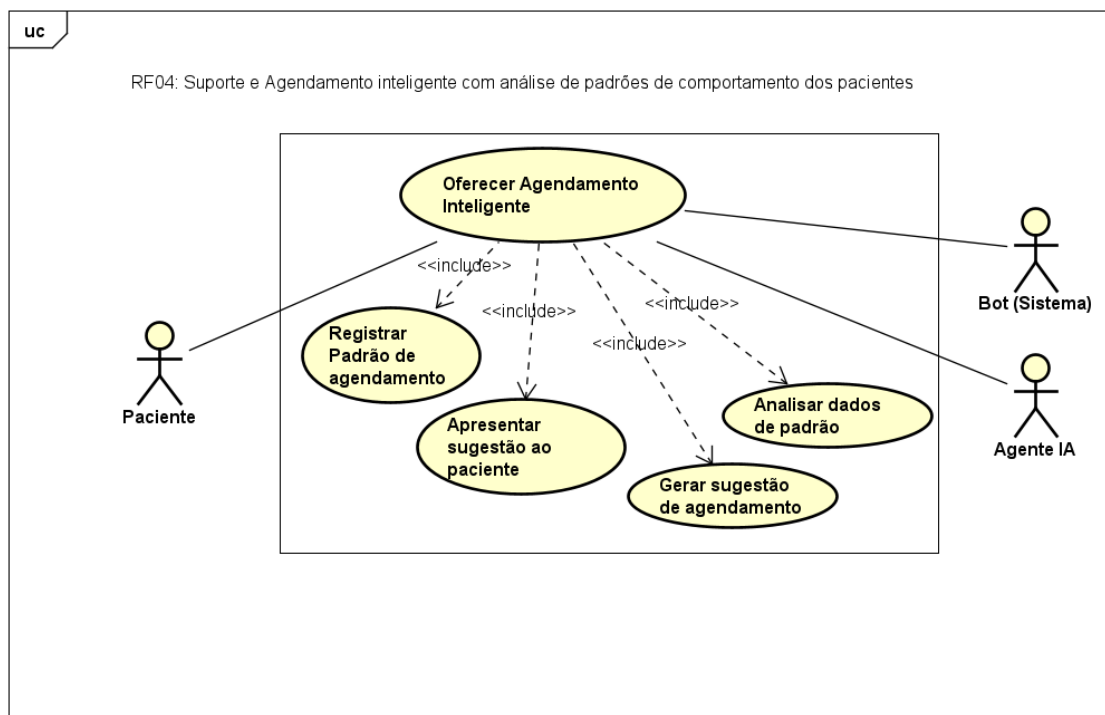
Fonte: Elaborado pelo Autor (2025).

Figura 03 - Requisito Funcional 3.



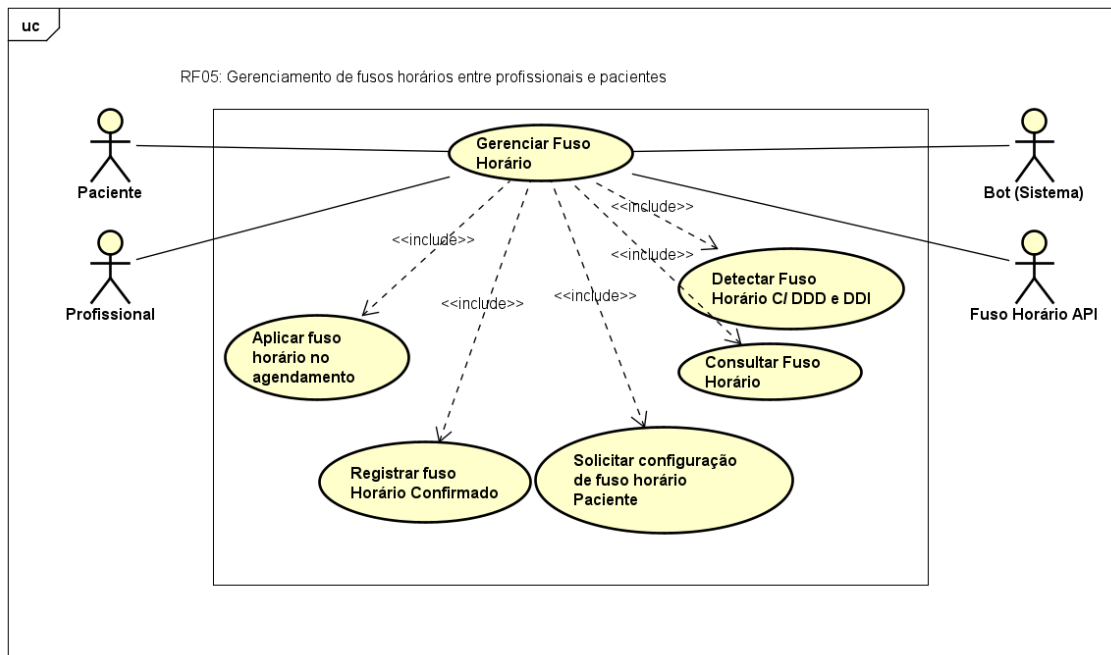
Fonte: Elaborado pelo Autor (2025).

Figura 04 - Requisito Funcional 4.



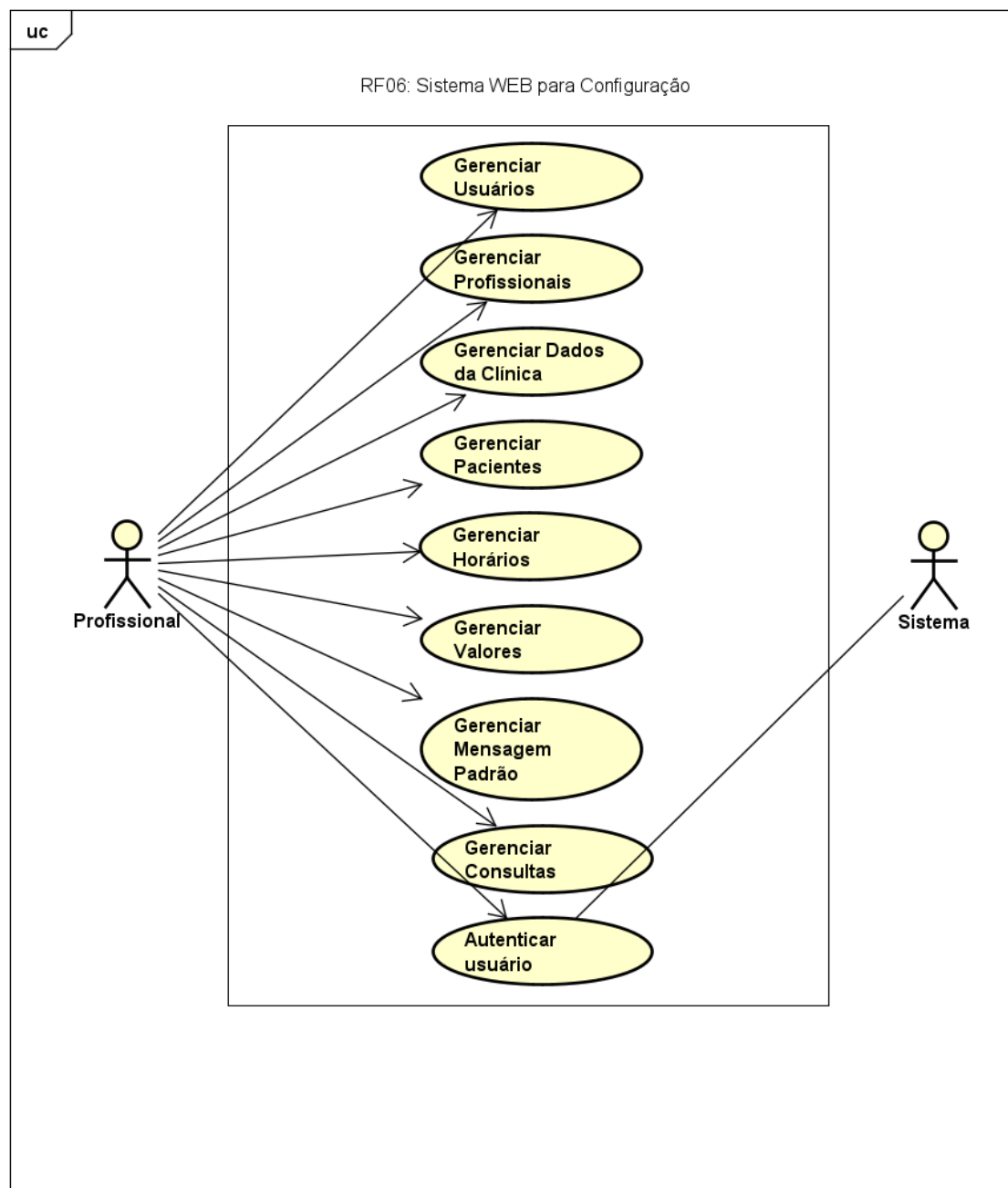
Fonte: Elaborado pelo Autor (2025).

Figura 05 - Requisito Funcional 5.



Fonte: Elaborado pelo Autor (2025).

Figura 06 - Requisito Funcional 6.



Fonte: Elaborado pelo Autor (2025).

3.2. Considerações de design

Nesta seção, são aprofundadas as escolhas de design que orientam o desenvolvimento do assistente virtual inteligente. São abordadas as alternativas consideradas e apresentadas as justificativas para as decisões adotadas, com o objetivo de garantir a robustez, escalabilidade e manutenibilidade da solução. Inicialmente, é apresentada uma Visão Inicial da Arquitetura, na qual são descritos os principais componentes do sistema e suas interconexões. Em

seguida, são discutidos os Padrões de Arquitetura aplicados para estruturar o projeto de forma eficiente. Por fim, a arquitetura é detalhada por meio dos Modelos C4, proporcionando uma compreensão clara e hierárquica do sistema em diferentes níveis de abstração: Contexto, Contêineres, Componentes e Código.

3.2.1 Visão Inicial da Arquitetura

A arquitetura do assistente virtual será composta por quatro principais pilares: Chatbot integrado com o Telegram, a API para leitura das regras, o agente de IA para orquestrar esses tópicos e o sistema web de configurações e visualização de relatórios sobre os clientes e agendamentos.

O chatbot integrado com o Telegram será a porta de entrada dos pacientes, por meio dele os pacientes poderão enviar mensagens solicitando informações, retirando dúvidas, e realizando seus agendamentos e cancelamentos, além disso o chatbot poderá enviar lembretes aos pacientes referentes as próximas consultas, ou cobrança de valores a serem pagos.

A API será responsável por conter as regras de negócio do sistema e realizar a ligação com o banco de dados. Na API será tratado a gravação em base de informações como os agendamentos realizados, log de mensagens com o paciente, além de permitir a consulta de informações como horário de atendimentos, valores, localização e a conversão de fuso horários.

O Agente de IA será o núcleo de toda a operação, processando as requisições e aplicando a inteligência comportamental para sugerir horários, além de realizar a integração com o Google Calendar. Também será utilizado o Google Gemini para tratamento das mensagens por parte do paciente como por parte da API. Além de ter um outro agente secundário que será responsável por apreender o comportamento de cada usuário com base em suas consultas, retornando então para a IA principal os melhores horários para oferecer a esse paciente, tal como enviar lembretes assim que chegar perto da data da consulta levando em consideração o histórico de cancelamento de cada cliente.

O sistema web servirá como a base de alimentação do nosso agente de IA, ou seja, teremos o cadastro e configuração de cliente, profissionais, horários de atendimento, valores, tempo de consulta, localidade das clínicas e mensagem

de boas-vindas disponíveis no qual o agente fará a leitura para que tenha conhecimento sobre como o profissional trabalha e assim poder responder de forma coerente a cada paciente, além de realizar os agendamentos somente dentro dos dias e horários pré-definidos.

Como mencionamos anteriormente, o chatbot se comunicará diretamente com o Agenda de IA que atua como o núcleo/orquestrador de tudo, a partir de então ele irá armazenar em memória as informações que estão sendo processadas no chat ativo, realizando as interações necessárias conforme a solicitação do cliente, ou seja, o agente de IA poderá se comunicar com a API para buscar informações como horários e valores das consultas, com essa resposta o agente de IA poderá enviar essa informação ao Gemini que irá construir uma mensagem coerente para que o agente de IA possa retornar ao paciente. Caso o paciente queira agendar uma consulta, a integração do agente de IA se dará diretamente com o Google Calendar, registrando a consulta na agenda da psicóloga e com a API para gravar na base de dados esse agendamento, tendo a possibilidade de ter um histórico de consultas para geração dos indicadores.

3.2.2 Padrões Arquiteturais usados

O sistema web será desenvolvido seguindo o modelo MVC (Model-View-Controller), garantindo a separação clara entre interface, regra de negócio e banco de dados. A arquitetura vai seguir o modelo em camadas e será composto por:

- Camada de apresentação (Chatbot e Website);
- Camada da regra do negócio (API RESTful e Agente de IA);
- Camada do banco de dados. E a comunicação ocorrerá via API RESTful utilizando JSON como formato de dados.

3.2.2.1 Modelos C4

A arquitetura do sistema foi detalhada seguindo o modelo C4, com os seguintes níveis:

- **Contexto**

- Paciente: Interage exclusivamente com o chatbot;
- Profissional: Acessa o sistema web para configurar e consultar dados;
- Sistema externo: Integração com o Google Calendar;
- Agente de IA centraliza todas as interações, intermediando a comunicação entre os elementos.

- **Contêineres**

- Chatbot (Telegram): Será o canal de comunicação com o paciente;
- Agente de IA: Orquestrador do sistema e responsável pelas decisões inteligentes;
- API: Camada de negócio e integração com o banco;
- Sistema web (react): Painel de controle pra o profissional.
- Banco de dados (PostgreSQL): Persistência de dados estruturados
- Serviços externos: Google Calendar e Gemini (para NLP e geração de mensagens).

- **Componentes**

Dentro da API teremos alguns componentes principais tais como:

- AgendamentosController;
- UsuariosController;
- ConsultasController;
- ConversaoFusoHorarioController.

Já dentro do agente de IA teremos os principais componentes:

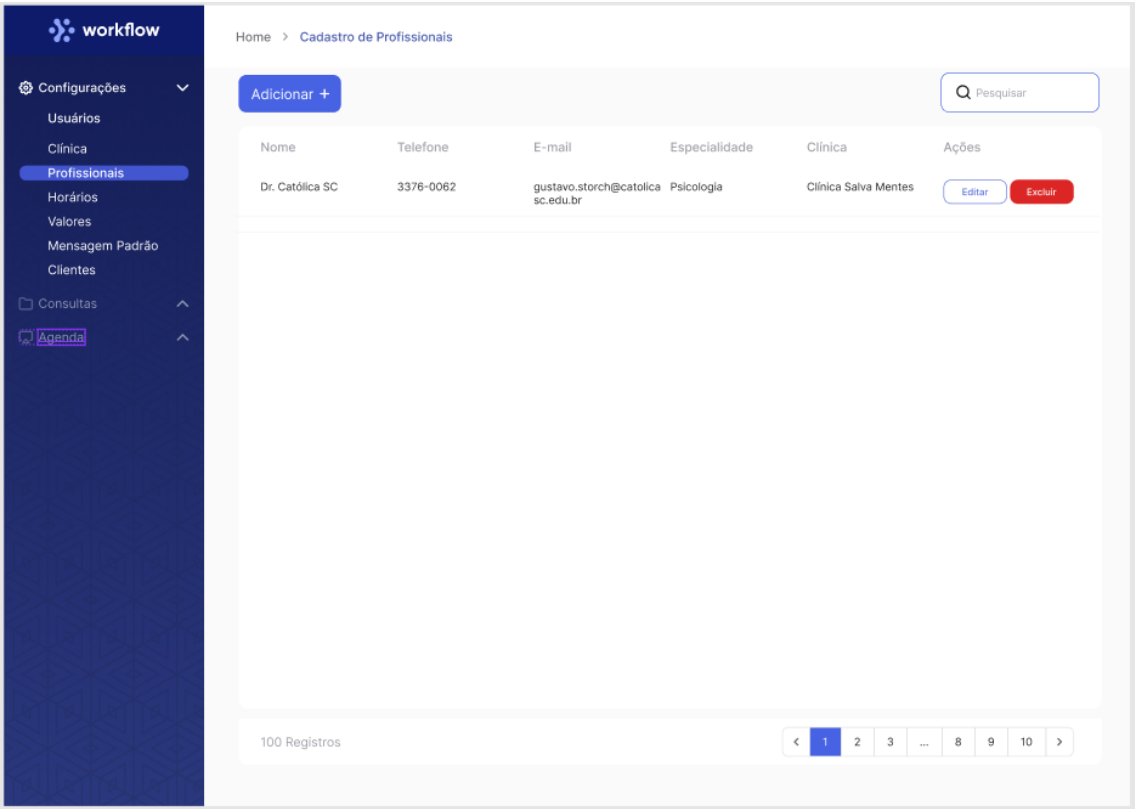
- IAAgendador;
- IAComportamental;
- GeradorMensagem.

3.2.3 Protótipos e interface do usuário

Para ilustrar o funcionamento e o fluxo de uso do sistema, foram desenvolvidos protótipos no Figma com o objetivo de representar as principais

telas da interface web acessada pelos profissionais de saúde. O Figma é uma ferramenta de interfaces baseado na nuvem, voltada para a criação colaborativa de protótipos interativos e responsivos (FIGMA, 2025). As telas foram projetadas com foco na simplicidade, usabilidade e padronização visual.

Figura 07 – Cadastro de Profissional



Fonte: Elaborado pelo autor (2025)

Conforme ilustrado na Figura 07, a tela de cadastro de profissionais possui um layout simples e intuitivo. É apresentada uma lista com todos os profissionais registrados no sistema, acompanhada de um botão para adicionar novos profissionais, que ao ser clicado, exibe um modal com os campos necessários para preenchimento. Cada registro também oferece a opção de edição, abrindo o mesmo modal com os dados já preenchidos, além da opção de exclusão direta.

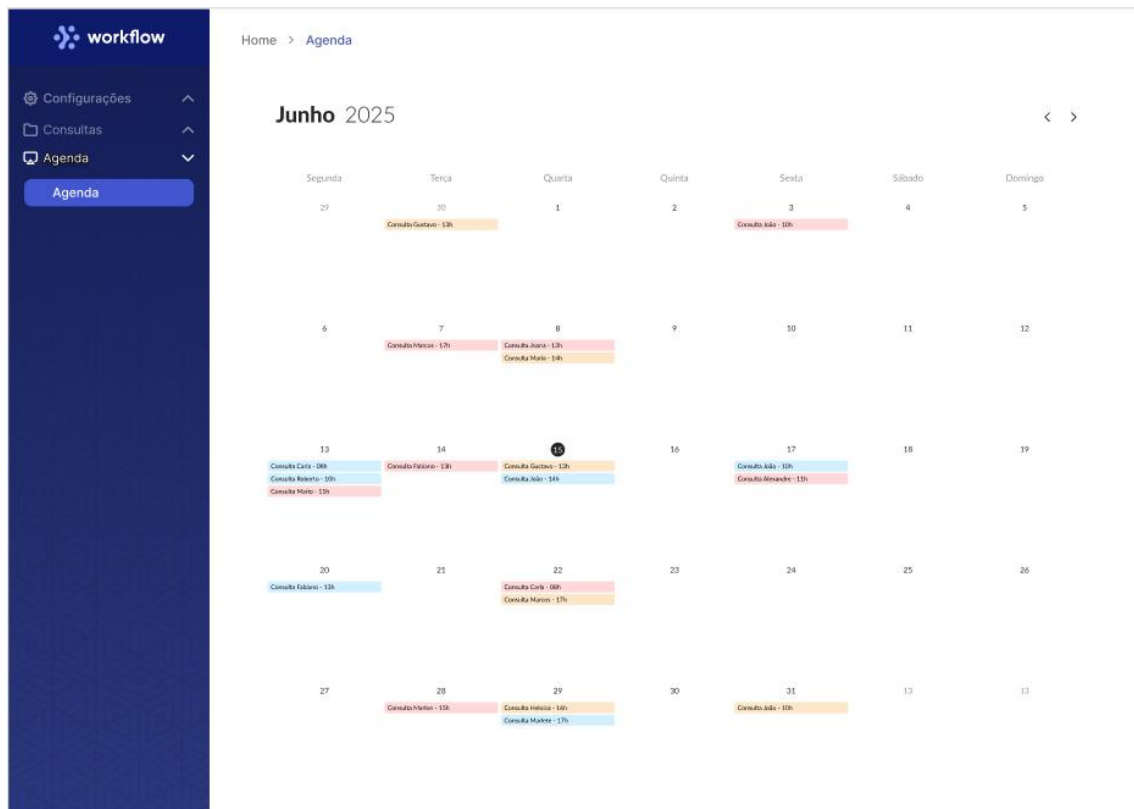
Figura 08 – Cadastro de Horário

Profissional	Dias da Semana	Horários Manhã	Horário Tarde	Horário Noite	Ações
Dr. Católica SC	Seg, Ter, Qar, Qui, Sex	08:00 - 12:00	13:00 - 17:00	20:00 - 22:00	Editar Excluir

Fonte: Elaborado pelo autor (2025)

A Figura 08 apresenta a tela destinada ao cadastro de horários disponíveis para atendimento. A estrutura visual segue o mesmo padrão da tela anterior, garantindo consistência na experiência do usuário. São listados os horários cadastrados no banco de dados, com opções para adicionar, editar ou remover horários conforme a necessidade do profissional.

Figura 09 – Consulta de Agenda



Fonte: Elaborado pelo autor (2025)

A Figura 09 mostra a tela de consulta da agenda de atendimentos. Essa funcionalidade será integrada com o Google Calendar, permitindo ao profissional visualizar os agendamentos em tempo real, de forma prática e organizada. A integração facilita o acompanhamento da rotina de atendimentos, otimizando o planejamento diário.

3.3. STACK DE TECNOLOGIAS

Para definição da Stack tecnológica do projeto foi considerado critérios como a robustez, integração entre as tecnologias, curva de aprendizado, compatibilidade com as soluções modernas de automação, inteligência artificial e desenvolvimento web.

3.3.1. Linguagem de programação

-C# (.NET 8):

Foi escolhido o C# para o desenvolvimento da API por sua robustez, segurança e seu bom desempenho em aplicações. O framework .net 8, oferece suporte moderno a aplicações web, APIs RESTful, autenticação, injeção de dependências e middlewares. A escolha se deu também pela maturidade da plataforma e pela facilidade de manutenção e escalabilidade.

- Python:

Será utilizado o python para o desenvolvimento do agente de inteligência comportamental, que é responsável por analisar o histórico dos pacientes e sugerir horários mais adequados com base em seus comportamentos anteriores. Ele também foi escolhido por sua versatilidade, e por sua grande adoção para projetos que envolvem IA e ciências de dados, além das bibliotecas essenciais disponíveis como o pandas, scikit-learn e fastapi.

3.3.2. Frameworks e bibliotecas

- .Net Web API:

Será utilizado o framework .net web api para a construção de APIs RESTful em C#, desenvolvendo uma estrutura clara de controllers e serviços com integrações eficientes com o banco de dados postgresql.

- REact:

O frontend do sistema que ficará disponível ao profissional será construído utilizando o react, por ter sua base em Javascript e ser amplamente utilizada para o desenvolvimento de interfaces modernas e responsivas. Utilizando o react, será possível criar componentes reutilizáveis, otimizando o código.

-N8N:

Esta ferramenta de automação de fluxo será o orquestrador entre a API, a IA e os demais serviços externos, como o Google Calendar e o Google Gemini. Com os fluxos visuais e integrações flexíveis, o n8n permite criar conexões entre os módulos sem necessidade de codificação excessiva.

- PostgreSQL:

Será utilizado o banco de dados relacional PostgreSQL, muito por conta da sua compatibilidade com o .net, possibilidade de subir um serviço em nuvem do banco de dados, sua escalabilidade, e seus recursos avançados como procedures, JSONB e triggers.

- Google Gemini:

Será utilizado o Gemini para interpretar mensagens dos pacientes e gerar as respostas com a linguagem natural. Integrado via API, o Gemini permite que o sistema ofereça interações mais naturais e humanas com o paciente, trazendo ao paciente a sensação de estar conversando com um atendente verdadeira.

3.3.3. Ferramentas de desenvolvimento e gestão de projetos

- Visual studio / Visual studio code

Ambas as ferramentas serão utilizadas ao decorrer do desenvolvimento, o Visual Studio será utilizado para o desenvolvimento da API em C# .Net, enquanto o Visual Studio Code será utilizado para o frontend em React.

- Postman:

Será utilizado o Postman para validação dos endpoints da API.

-Figma:

Foi utilizado para a criação dos protótipos das interfaces das telas do sistema web que o profissional terá acesso.

- Git e GitHub:

Será utilizado para o controle de versão do sistema web, como dos agentes.

- Trello:

Será a ferramenta de gerenciamento de tarefas, e organização do projeto durante todo o desenvolvimento do projeto.

- N8N (self-hosted):

É a plataforma escolhida para integração dos agentes com o restante do projeto. Ele é uma plataforma Low-Code de automação, configurada para realizar integrações entre os módulos do sistema com o agente de inteligência.

3.4. Considerações de segurança

Este projeto é voltado ao recebimento de dados sensíveis de pacientes, como informações pessoais, dados de saúde e endereços. Por esse motivo, é fundamental que sejam adotadas boas práticas de segurança, a fim de garantir a confidencialidade, integridade e o uso adequado das informações. A seguir, são apresentadas algumas das principais medidas de segurança que serão implementadas no sistema.

3.4.1. Proteção de dados e LGPD

O sistema será desenvolvido seguindo todas as diretrizes presentes na lei geral de proteção de dados, sendo adotadas medidas técnicas para proteger os dados pessoais, garantindo a segurança de todos. Será adotado o princípio do mínimo necessário, assim o sistema salvará em sua base de dados somente os dados que realmente farão uso dentro do sistema.

3.4.2. Autenticação e controle de acesso

O login do sistema web de configurações que os profissionais terão acesso contará com um sistema de login protegido com autenticação de senha criptografada com hash e salt. Já o acesso as funcionalidades serão liberadas de acordo com o grupo de permissões que o usuário estiver configurado.

3.4.3. Comunicação segura

Toda a comunicação do sistema, seja ela entre website e api, ou até mesmo o Telegram, será feita usando os protocolos HTTPS, garantindo que todos os dados sempre sejam criptografados. A API contará com a proteção de

chaves de autenticação, TOKENS JWT, assim impedindo acessos não autorizados aos endpoints da API.

3.4.4. Boas práticas de desenvolvimento seguro

Serão testadas todas as entradas de dados, para que assim possamos prevenir ataques como SQL Injection, XSS e Command Injection. Também será adotado o método de secure coding, incluindo o uso de bibliotecas confiáveis e frequentemente atualizadas.

3.4.5. Dependências externas

A integração com ferramentas como o Google Calendar, N8N e Google Gemini, será feita por meio de APIs Oficiais e com a autenticação segura por meio do OAuth. O N8N também será configurado em ambiente protegido e com restrição de acesso ao seu painel de configuração.

4. PRÓXIMOS PASSOS:

A próxima fase do projeto será voltada para o desenvolvimento, iniciando com a criação do banco de dados PostgreSQL, logo após realizando o desenvolvimento da API que fará parte do website e da integração com o Agente de IA. Com esses dois tópicos prontos, é possível iniciar a integração das partes principais do sistema, Telegram, N8N, Google Gemini, Google Calendar, API e banco de dados, seguido da criação da IA e treinamento dela para prever os melhores horários de agendamentos de cada paciente. Com tudo isso finalizado será criado o website onde será possível os profissionais estarem configurando todas as informações que a inteligência artificial precisará para trabalhar perfeitamente.

5. REFERÊNCIAS

DA SILVA JÚNIOR, Daniel Melo et al. GESTÃO EM SAÚDE: PRÁTICAS GERENCIAIS PARA A OTIMIZAÇÃO DOS SERVIÇOS. LUMEN ET VIRTUS, v. 15, n. 43, p. 9531-9541, 2024.

GADELHA, Carlos A. Graboís. A Saúde como opção estratégica para o desenvolvimento do Brasil. SAÚDE É DESENVOLVIMENTO, p. 12, 2022.

SANTOS, Isabela. AUTOMAÇÃO COM INTELIGÊNCIA ARTIFICIAL: ANÁLISE DE UMA EMPRESA PROVEDORA DE PLATAFORMA DE CHATBOT COM IA. 2025.

BRITO, Ariadne Elesbão et al. MEDCLINIC: GERENCIAMENTO DE CLÍNICAS MÉDICAS. 2018.

FIGMA. Design, prototype, and collaborate all in the browser – with Figma. 2025. Disponível em: <https://www.figma.com>. Acesso em: 27 jun. 2025.